UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Telecomunicació
i Aeroespacial de Castelldefels

TREBALL DE FI DE GRAU

TFG TITLE: A trajectory computation algorithm for adaptive aircraft descents considering altitude and speed restrictions

DEGREE: Enginyeria Tècnica Aeronàutica, especialitat Aeronavegació

AUTHOR: Èric Soler Arjona

ADVISORS: Xavier Prats Menéndez
David de la Torre Sangrà

DATE: July 6, 2023

Títol: Algorisme de càlcul de trajectòries per a descensos adaptatius d'avions tenint en compte restriccions d'altitud i velocitat
Autor: Èric Soler Arjona

Directors: Xavier Prats Menéndez
David de la Torre Sangrà

Data: 6 de juliol de 2023

Resum

L'objectiu principal d'aquest treball de fi de grau és millorar el nivell de maduresa de l'eina de simulació de trajectòries desenvolupada pel grup de recerca Icarus UPC. En conseqüència, els avenços realitzats en aquest projecte tenen com a objectiu adaptar el programari per oferir una interpretació més realista dels escenaris de simulació i, per tant, obtenir resultats més precisos i fidels a les operacions aèries reals.
Per assolir aquest objectiu, s'ha dut a terme la planificació i disseny d'una nova funcionalitat. Aquest informe comença descrivint el concepte de la implementació en un context aeronàutic, establint així les bases per assolir els requisits de l'actualització del programari. A més, es presenta un esquema complet de l'arquitectura de la implementació, juntament amb una explicació detallada del funcionament dels diferents blocs lògics. Aquesta secció inicial de l'informe també descriu la integració d'aquesta implementació dins l'arquitectura del programari principal.
Posteriorment, aquest informe ofereix detalls sobre el procés de verificació dut a terme per garantir el bon funcionament de la lògica de la funcionalitat. Aquest procés engloba la planificació d'un escenari particularment interessant, que inclou diversos subcasos en els quals s'avalua a fons la implementació. Addicionalment, s'analitzen meticulosament els resultats produïts pel programari i se n'extreuen conclusions sobre la lògica aplicada.
Més enllà de la verificació anteriorment esmentada, també es descriu un procés de validació que planteja un entorn més fidel a la realitat. Aquest procediment permet estudiar el rendiment i eficàcia del sistema en un escenari d'aplicació plausible. Finalment, els resultats són sotmesos a una anàlisi en profunditat. Aquest estudi detallat proporciona una comprensió més profunda dels resultats, facilitant coneixements valuosos i possibles millores potencials per a futures etapes de desenvolupament.

Title : A trajectory computation algorithm for adaptive aircraft descents considering altitude and speed restrictions
Author: Èric Soler Arjona

Advisors: Xavier Prats Menéndez
David de la Torre Sangrà

Date: July 6, 2023

Overview

The main objective of this final degree project is to enhance the maturity level of the trajectory simulation software suite developed by the Icarus UPC research group. Consequently, the advancements made in this project aim to adapt the software to provide a more realistic interpretation of the simulation scenarios and, therefore, yield more accurate and faithful results, compliant with real flight operations.
To accomplish this objective, the planning and design of a novel functionality are undertaken. This report begins by describing the conception of the implementation within an aeronautical context, thereby establishing the guidelines to fulfill the requirements of the software upgrade. Additionally, the report provides a comprehensive outline of the implementation's architecture, along with a detailed explanation of the operational aspects of the various logical blocks that conform the functionality. This initial section of the report further describes the integration of this implementation within the main software architecture.
Afterwards, this report provides details regarding the verification process conducted to ensure the proper operation of the logic. This process encompasses the planning of a particularly interesting scenario, comprising various sub-cases in which the implementation is thoroughly tested. Moreover, the results produced by the software are meticulously analyzed and conclusions about the applied logic are drawn.
In addition to the previous verification, another validation process of a more realistic situation is presented. This procedure allows for a meticulous examination of the system's performance and effectiveness within a real-world application scenario. Finally, the results are subjected to an in-depth analysis. This detailed study provides a deeper understanding of the outcomes, yielding valuable insights and potential improvements for the future development of the software.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

One of the main safety measures implemented in air traffic management is the maintenance of separation between aircraft throughout all phases of flight. In order to achieve this objective, two types of separation strategies are employed: strategic measures and tactical measures. The responsibility for managing air traffic falls upon the Air Navigation Service Provider (ANSP), an entity that has a team of Air Traffic Controllers (ATC) who are entrusted with the task of ensuring real-time tactical separation between aircraft. However, in congested airspaces, such as some Terminal Maneuvering Areas (TMA), separation must be maintained not only between arrivals and departures but also among converging arrival flows. Therefore, additional measures, known as strategic measures, are needed to decrease the ATC workload. These strategic separation measures are typically given in terms of altitude constraints in certain waypoints (or legs) of the depart or arrival/approach procedures, along with potential speed limitations. In this context, the ANSP publishes navigation charts detailing the flight procedures and where these operational constraints are depicted.

Flight efficiency has a great importance for airlines, as they try to fly the optimal trajectory throughout every phase of flight, according to their business needs. The primary focus of this project lies in descent operations, where a trade-off between an efficient vertical descent profile and the traffic capacity of the concerned airspace emerges. In these situations, a Continuous Descent Operation (CDO) is the most desirable flight procedure in order to maximise flight efficiency. CDO involves the aircraft descending continuously from cruise altitude to the runway threshold, without leveling off, using idle or near-idle thrust thorough the whole descent. This procedure reduces fuel consumption and noise emissions, thereby enhancing the overall efficiency. However, CDO operations suffer from a well-known drawback, as Dr. Raúl Sáez highlights in his PhD thesis [1]: "*the loss of predictability from the air traffic control (ATC) point of view in terms of overfly times at the different waypoints of the route*". This unpredictability can compromise the capacity of the airspace, and thus measures need to be put in place. A balance must be achieved, as the increase in operational constraints leads to a less efficient trajectory but improves capacity.

The Icarus UPC research group has developed a suite of software tools called Dynamo, capable of predicting lateral and vertical aircraft trajectory profiles based on a known aircraft intent [2, 3] and using realistic and accurate weather and aircraft performance data. Dynamo can also optimize these lateral and/or vertical profiles. It is worthwhile mentioning that a complete and detailed explanation of Dynamo's purpose and functioning can be found in [4]. Furthermore, as stated in Dynamo's users manual: [5] "*Dynamo can be used for dispatching purposes (i.e., computing flight plans) for on-board optimisation computations, for trajectory prediction or simulation*". Nevertheless, one of the main limitations of this software is that it is not able to consider the operational limitations (speed and/or altitude constraints) set in approach or arrival charts. Although Dynamo can compute optimal descent profiles, these may infringe one or several constraints and not be fully compliant with the published procedure, since they follow a CDO. This project aims to address this problem by providing Dynamo with a new set of features aimed at incorporating operational constraints published in arrival and approach charts into the vertical trajectory prediction process.

Hence, the main objective of this project is to design, implement and validate this new functionality, called Intent Procedure Adapter (IPA), and integrate it seamlessly within the rest of the Dynamo software. In addition to implementing the algorithm, this project has more specific goals, such as integrating the new function robustly and in a modular way with other functionalities of Dynamo, and to anticipate future development needs. Additionally, the project aims to research and identify application cases to validate the implementation proposed, and to apply it to a real-world operational scenario to verify its proper operation.

Overall, this project seeks to make descent phase profile predictions computed by Dynamo compliant with real flight operations, ultimately contributing to the continued improvement of the product developed by Icarus UPC research group. Consequently, the IPA aims to enhance the maturity level of the software by making it closer to state-of-the-art flight management systems (FMS). The FMS is an on-board computer system used by pilots, which assists in planning vertical profiles, optimizing fuel consumption, and ensuring safe and efficient flight operations by taking into consideration operational regulations, among others.

CHAPTER 1. BACKGROUND

In order to give context to the application of the IPA, it is firstly needed to identify the context in which this new feature is going to be used. To begin, this chapter gives an overview on navigation charts and published flight procedures. Then, some information on flight management systems (FMS) is given. Finally, an overview on Dynamo, its architecture and methodological aspects relevant to this TFG are presented.

1.1. Published flight procedures

Operational flight charts are the main source of information for pilots or dispatchers, as it is in these charts where the air navigation service provider (ANSP) sets the limitations that the aircraft must follow when flying by a certain waypoint of the chart. For instance, ENAIRE is the main ANSP in the Kingdom of Spain, and is the source from where the charts assessed in this project are taken. After an analysis of many different operational flight charts, such as the one depicted in Figure 1.1, and after also checking the ICAO regulations for procedure design [6], it has been concluded that the ANSP might only impose restrictions on speed and/or altitude.

As it can be seen in Figures 1.1 and 1.2, where ZMR4A RNAV1 STAR (standard terminal arrival route) is shown, all the waypoints of the arrival chart have different limitations in altitude and velocity [7]. For example, waypoint PODOG is limited with a minimum altitude of FL200 and a maximum IAS (indicated airspeed) of 250 kt, or waypoint RILKO, which is the IAF (initial approach fix), has a maximum altitude of 11,000 ft and a maximum IAS of 220 kt. This means that the aircraft must not exceed the limitation at the waypoint. As commented before, these constraints are set to ensure some strategic separation between departures and arrivals, and also to facilitate the sequencing and merging procedures of arrival traffic.

It is worth noting that speeds in the operational charts are designated as IAS (indicated air speed) because they are the speeds that the pilot reads from the instruments in the cockpit. However, in Dynamo, as it is dealing with the speed of the aircraft in itself, it is used the CAS (calibrated air speed), which corresponds to the IAS corrected for instrumentation errors. Hence, in this project IAS shown in the charts are treated as CAS.

To summarise, there are three main application cases relevant to this project: when the waypoint has a maximum altitude constraint, when it has a minimum altitude constraint, and when it has a maximum IAS constraint. Upon analysing various charts from multiple airports, it has been observed that IAS is never subject to a minimum limitation. Consequently, this case is not important and is thereby disregarded.

In addition to these three cases, there is a fourth one which is not essentially a limitation given in the chart but needs to be taken into account for the logic of the IPA implementation. This case is when the aircraft reaches a waypoint in itself, where it needs to update the potential constraints to the new leg. A leg is the portion of trajectory delimited by two adjacent waypoints and these two waypoints set the limitations of the leg.

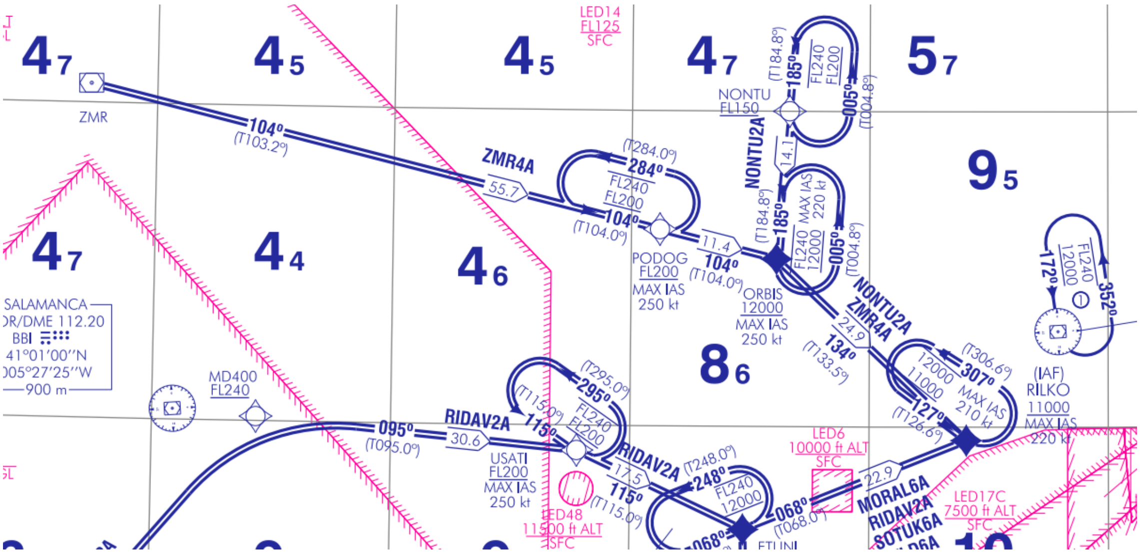


Figure 1.1: LEMD STAR 1 ZMR4A (Source: Spanish AIP [7])

ZMR4A RNAV1											
001	IF	ZMR	–	–	+0.4	–	–	–	–	–	RNAV1
002	TF	PODOG	–	104 (103.2)	+0.4	55.7	–	+F200	-250	–	RNAV1
003	TF	ORBIS	–	104 (104.0)	+0.4	11.4	–	+12000	-250	–	RNAV1
004	TF	RILKO	–	134 (133.5)	+0.4	24.9	–	+11000	-220	–	RNAV1

Figure 1.2: LEMD STAR 1 ZMR4A waypoint constraints (Source: Spanish AIP [7])

1.2. Typical FMS implementations

As explained in the introduction, there is a system inside the aircraft capable of computing flight trajectories, as well as implementing guidance commands, called Flight Management System (FMS). It is a sophisticated computer system that automates and manages various aspects of a flight. The FMS integrates with other onboard systems such as the autopilot and navigation equipment to provide a fully integrated solution for managing the aircraft trajectory and flight mission.

The FMS works by using a combination of data inputs from various sensors, including GNSS (global navigation satellite systems), air data computers, and other avionics systems, to determine the aircraft's position, speed, and altitude. What is important for this project is that it then uses this information to calculate the most efficient flight path, taking into account factors such as wind speed and direction, fuel consumption, and other flight-related parameters. In fact, it allows the pilots to program a flight plan into the system, specifying the desired altitude, speed, and route for the aircraft. The system then uses this information to provide the pilots with real-time guidance and alerts to keep the aircraft on course and on schedule.

One of the main objectives of this project is for Dynamo to be able to replicate the FMS functionality when planing a descent trajectory. That is why, it is important to clearly understand its functioning when designing and building the IPA. This implementation uses the FMS as a base for its logic of solving the different cases found when introducing constraints in waypoints.

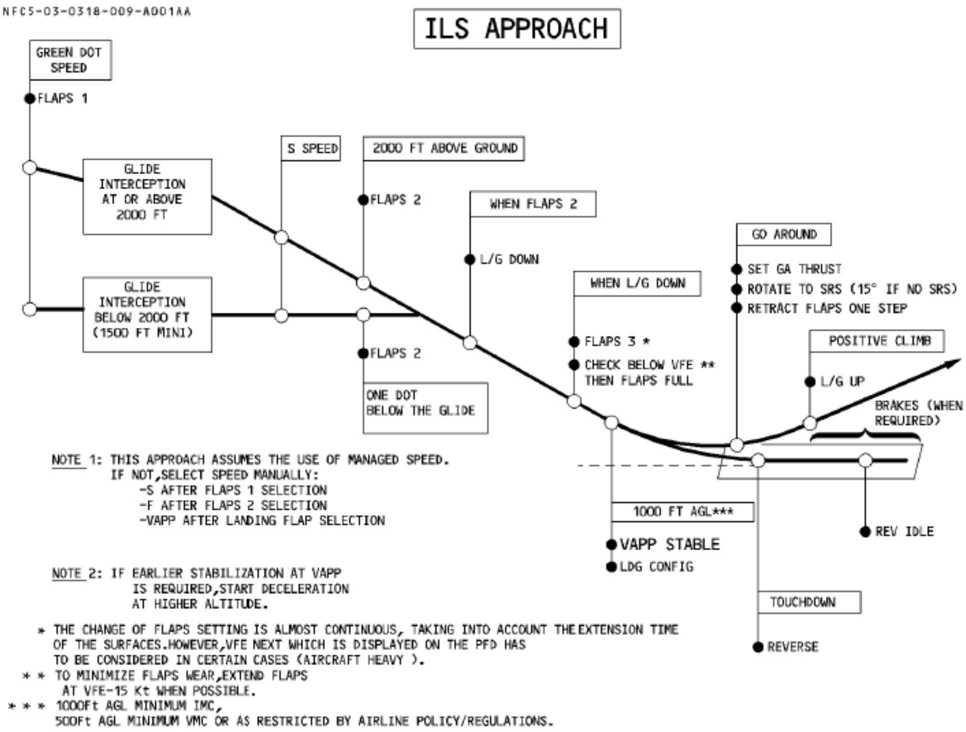


Figure 1.3: ILS approach guidance profile (Source: Airbus A320 FCOM [8])

Yet, FMS are commercial devices subject to intellectual property rights and the exact models used and implementation details are not publicly disclosed. Depending on the FMS manufacturer, different aircraft motion models, aircraft performance models, and/or weather models might be implemented. Hence, conducting a detailed analysis of its functioning is nearly impossible. Nevertheless, pilots and aircraft operators use a document called FCOM (flight crew operating manual), which describes the minimum indispensable information for them to be able to use the FMS properly.

The FCOM is produced by aircraft manufacturers and provides comprehensive information and guidance to flight crew members on the operation of an aircraft. It does not provide, however, detailed information on how different FMS features are implemented from an engineering point of view. As base of the IPA's operating methodology, the FCOM of the Airbus A320 is used in this project, as it is the main aircraft model used for validation purposes [8].

For instance, in Figure 1.3 it can be seen can how the FCOM explains the execution of an A320 ILS approach. This aircraft model is later used as a base to explain the final part of the approach profile used to do the simulations and the configuration of the aircraft during each phase of the approach.

With the limited engineering information that is provided by the FCOM and the profile shown in Figure 1.4, the FMS behaviour to comply with the operational constraints can be estimated. First of all, for maximum altitude constraints, it computes a geometric path taking the constraint's waypoint position and altitude. Then, when it encounters a speed limitation, it overrides the speed being used for the computation with that of the limitation. As stated in [8], "*the descent profile is usually the ECON speed profile, amended by any speed constraints and speed limit contained in the flight plan*". Finally, the last point of

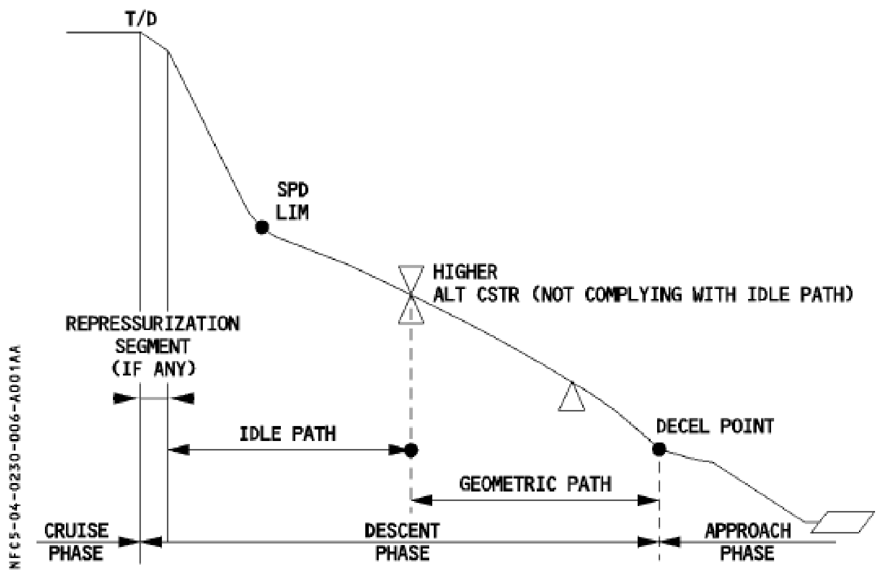
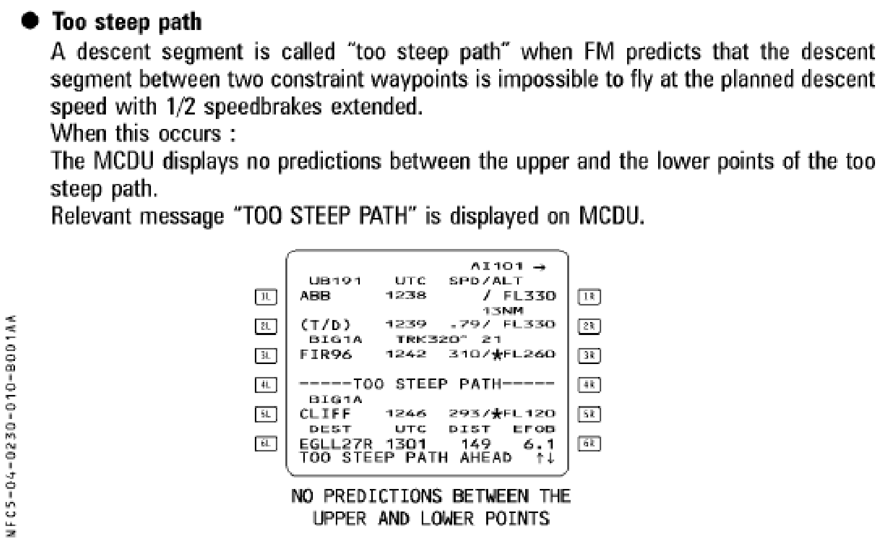


Figure 1.4: FMS descent trajectory (Source: Airbus A320 FCOM [8])



● **Too steep path**
A descent segment is called "too steep path" when FM predicts that the descent segment between two constraint waypoints is impossible to fly at the planned descent speed with 1/2 speedbrakes extended.
When this occurs :
The MCDU displays no predictions between the upper and the lower points of the too steep path.
Relevant message "TOO STEEP PATH" is displayed on MCDU.

Figure 1.5: FMS error: "Too steep path" (Source: Airbus A320 FCOM [[8]])

interest in the FCOM is when a minimum altitude constraint is not met. As it can be seen in Figure 1.5, the FMS displays an error message, *"TOO STEEP PATH"*, when an unfeasible flight path is found.

To conclude, current Dynamo implementation does a comparable job to the FMS when calculating the most efficient unimpeded vertical path, but does not consider potential constraints in waypoints. Therefore, in order to reproduce the best way possible the operation of an FMS, Dynamo needs a new feature which takes into consideration these limitations found in navigation charts. The IPA is intended to give a solution when a path does not comply with these constraints, using the same methodology found in the Airbus FMS, as well as using its own additional features.

1.3. Dynamo

Dynamo is a software used to computing aircraft trajectories, allowing for trajectory prediction, optimisation, estimation and simulation [4]. Taking a closer look, Dynamo is able to compute four-dimensional (4D) trajectories taking into account the lateral and vertical profiles along time.

Even though Dynamo takes as inputs a wide variety of parameters defined by the user, the main inputs are: the lateral route, with the destination and origin airports and all the procedures (SID, STAR, AWY) the aircraft must follow with their corresponding waypoints; the vertical profile, with the aircraft performance characteristics requirements for every flight phase; and the flight configuration, with the masses of the aircraft, the performance model, etc.

In this project, however, the objective is to implement a functionality which only concerns the vertical profile, assuming that the lateral trajectory (i. e., the so called route) is known and fixed beforehand. Thus, from this point on, the project is focused on the vertical profile of the flight plan, and all aspects of Dynamo related to the lateral profile are not taken into consideration.

1.3.1. Dynamo layers

As it can be seen in Figure 1.6, the main logic of Dynamo when computing the vertical profile is divided in two different layers, ordered from top (i. e. highest level in the software architecture) to bottom: the trajectory optimiser (TO) and the trajectory predictor (TP).

The main purpose of the TO is to select the best trajectory according to some arbitrary user criteria. It achieves this by calling the TP multiple times with different profiles and receiving the integrated trajectory for each profile. The TO receives as inputs: the template vertical profile, scenario configuration, and the route followed by the aircraft, and returns as final output: the optimal trajectory.

The TP is responsible for taking all the constraints and guidelines input by the user or the TO, and create a feasible profile using a given integration method. The profile is divided in phases in order to tweak all its parts as the user wants (i. e. each phase has some flight

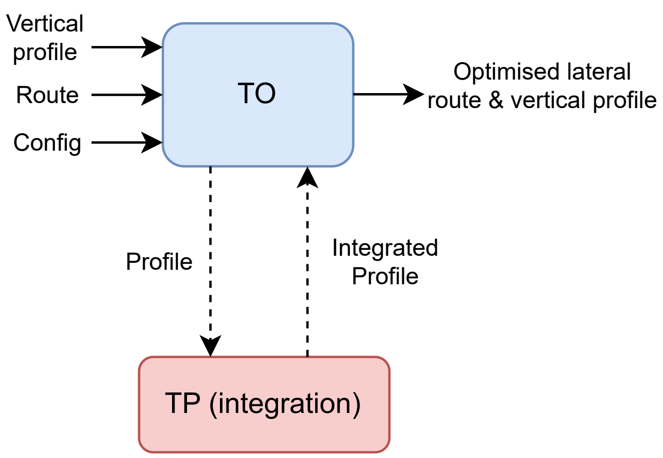


Figure 1.6: Dynamo layers

parameters intended to be flown and some constraints not to be crossed), which will be further explained in section 1.3.2..

The IPA is mainly implemented and tested using the TP layer, as it is essentially a modification of the profile being integrated in the prediction layer. That is the reason why a deeper explanation of the complexity of the TO layer will not be provided in this report, as it is out of the scope of this project. In addition, as stated before, the route is fixed by the user and it is essentially a sequence of waypoints, some of which have defined constraints that serve as the primary input for the IPA's operation.

1.3.2. Intents to controls

As it is mentioned before, the TP runs the integration of a certain phase. These phases are grouped in what is called a block, which constitutes a part of the vertical profile (i. e. a vertical profile is a group of blocks, which in turn is a group of phases). For example, in the descent part of the flight, a block from the TOD (top of descent, when the aircraft starts the descent) until it reaches FL100 can be defined. This block will have different phases which will characterise the performance of the aircraft throughout the application period.

A phase has two main characteristics: aircraft intent instructions and end conditions. As explained in [9] "*a sequence of path constraints, known as aircraft intent instructions, are required to generate a trajectory in the planning phase. The path constraints are operational instructions that specify how the aircraft should intend to meet the preferences defined in the planned trajectory. [...] The (auto)pilot uses a sequence of guidance modes to steer the aircraft in such phase. A guidance mode is a combination of commands that specify how the aircraft should behave to perform the desired trajectory. In the vertical plane, two path constraints are required. For instance, at the beginning of the climb phase, setting the throttle at the maximum rate and accelerating the aircraft to meet the planned rate of climb.*"

This pair of intent instructions given as a guidance mode to compute the trajectory of the aircraft need then to be converted to what are called controls. The motion model of the aircraft is a two degree of freedom aircraft point-mass model, as vertical equilibrium is assumed, where these two degrees of freedom are controlled by variables γ (aerodynamic flight path angle) and π (throttle), which describe the motion of two aircraft actuators: elevator and throttle, respectively. The aircraft motion is modelled as follows[9, 4]:

$$
\begin{aligned}
\frac{\mathrm{d}h}{\mathrm{d}t} &= v\sin\gamma \\
\frac{\mathrm{d}s}{\mathrm{d}t} &= \sqrt{v^2\cos^2\gamma - W_x^2} + W_s \\
\frac{\mathrm{d}v}{\mathrm{d}t} &= \frac{1}{m}[T(\pi, v, h) - D(v, h, m, \xi)] - g\sin\gamma \\
\frac{\mathrm{d}m}{\mathrm{d}t} &= -q(T, v, h)
\end{aligned}
\tag{1.1}
$$

where h is the geometric altitude, s is the along path distance, v is the true airspeed and m is the mass of the aircraft. These variables compose the state vector $x = [h, s, v, m]$, which describes the motion characteristics of the aircraft at each time instant in the vertical plane (recall the lateral trajectory is fixed). In addition, T is the thrust delivered by the aircraft

Command 1 $(Elevator)$	**Command 2** $(Throttle)$	**Parameters vector**
MACH CAS ACC/DEC	THR	$p = [M, \pi]$ $p = [v_{CAS}, \pi]$ $p = [k, \pi]$
VS	MACH CAS ACC/DEC	$p = [v_h, M]$ $p = [v_h, v_{CAS}]$ $p = [v_h, k]$
FPA	MACH CAS ACC/DEC	$p = [\gamma_g, M]$ $p = [\gamma_g, v_{CAS}]$ $p = [\gamma_g, k]$
VS FPA ALT	THR	$p = [v_h, \pi]$ $p = [\gamma_g, \pi]$ $p = [v_h = 0, \pi]$

Table 1.1: Guidance modes in vertical plane (Source: [[9]])

engines, D is the aerodynamic drag, q is the fuel flow, W_x is the cross-wind component, W_s is the along-path wind component, g is the gravitational acceleration, and ξ is the setting of aircraft configuration high lift devices and landing gear.

As stated in [9] *A set of* $N = 12$ *possible guidance modes are considered to cover the whole aircraft trajectory in the vertical plane. All pairs of guidance commands (e.g., CAS-THR refers to flying at a constant calibrated airspeed with a minimum/maximum throttle rate) are summarized in Table 1.1. The first and second columns show the guidance commands that steer, respectively, the two independent aircraft actuators (i. e., elevator and throttle). The third column gives the two guidance command parameters that are needed to obtain the mathematical model of the system dynamics, for each guidance mode.*

In Table 1.1, M depicts the Mach number, v_{CAS} the calibrated airspeed (CAS), v_h is the vertical speed (i. e. rate of climb/descent), γ_g is the geometric flight path angle, and k is the acceleration/deceleration rate expressed in terms of energy share factor (ESF)[1]. Each pair of intents define the two independent variables, which set the performance of the actuators and so the motion of the aircraft throughout the corresponding phase.

As mentioned before, each phase must have at least one end condition. The purpose of the end conditions is to stop the integration process. These conditions can be set by various sources (profile, optimiser, etc.), but they are essentially a variable value which, once reached, triggers the integration of the following phase. For example: if a certain phase needs to be stopped when aircraft reaches a CAS of 200 kt, this value is set as an end condition for that particular phase.

Some of these intents and end conditions values are given by the user, but there are some of them which are computed taking into account the aircraft performance model (APM) and the atmosphere model (e. g. v_{gd} green dot speed of the aircraft). These models are explained in [10] and will not be further explained in this report, as it is not among its goals.

[1]Defined as the rate of how much of the available power is allocated to gain the potential energy in climb or descent: $k = (1 + \frac{v}{g}\frac{dv}{dh})^{-1}$

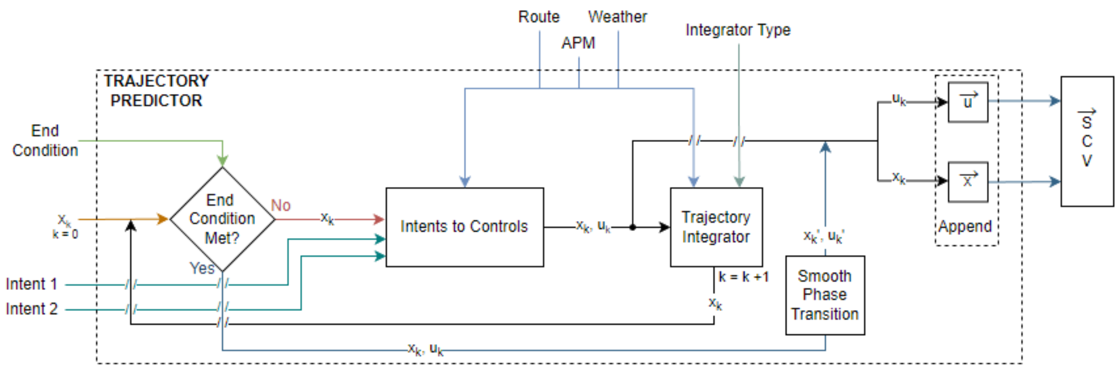


Figure 1.7: Trajectory Predictor internal functioning (Source: Dynamo wiki [5])

One of the main goals of this project is to provide a solution trajectory when the profile so far computed by Dynamo does not comply with the restrictions set by the operational charts. In order to achieve that, the IPA will modify some phases of user's vertical profile, by adjusting their intents and/or end conditions. Therefore, if the constraints are not met, the final result profile will be slightly different from the user's input.

1.3.3. TP operation

With all concepts stated, in order to completely understand how TP works, Figure 1.7 shows its internal operation when integrating an entire phase.

Citing Dynamo's documentation [5]: "*Given an initial state (x_k at k = 0), the TP will check whether the end condition is met. If this condition is not met, the pair of intents and the state vector at a given timestamp (k) will be used for computing the required controls (u_k) for that timestamp according to the given intents. The current state (x_k) and current controls (u_k) then perform two tasks: they are firstly stored into the respective vectors ($\vec{x}$ and $\vec{u}$), and then fed into the Trajectory Integrator, which shall return the state at the next timestamp (x_{k+1}), which will become an input to the control acquisition function, becoming the current state. This loop shall be repeated until the end condition is met or if a maximum number of iterations is reached; when the TP will stop the loop and indicate whether the phase has concluded correctly or the end condition was never reached. If the end condition has indeed been reached, a smooth phase transition will take place, where the final value of the state control vector will be finely tuned so it meets the exact intended value, obtaining the final state-control vector (x_k' and u_k'), which are then added to the resulting vector of state-control vectors.*"

CHAPTER 2. IPA DESIGN

The main objective of this final degree project is to enhance Dynamo so it is able to take into consideration the constraints set by the air navigation service providers (ANSP) in the charts describing flight procedures. In order to achieve that, a new functionality is proposed, the Intent Procedure Adapter (IPA), which gathers information set by the user in the route planning (i.e., sequence of waypoints) and unimpeded profile and modifies this profile (if needed) in order to take into account the constraints set in the route.

One of the first ideas when this project started was to make this implementation as generic as possible, trying to make it involve all the flight phases (climb, cruise, descent). However, upon realising the complexity of the solution for this approach, it was decided to shift the focus onto the descent phase, which is the one that is most constrained by the waypoint limitations, as well as the most challenging one. Thus, although from now on this project is mainly aimed to descents, a future development, out of the scope of this project, will be to adapt this implementation to all flight phases, as waypoint constraints are found all over the route. In order to enable this future work, the logic and code proposed in this TFG is designed sufficiently generic and modularised so it can be easily adapted for climb and cruise in the future.

2.1. IPA high-level architecture

In the current Dynamo implementation, as explained in section 1.3.1., the vertical trajectory computation is divided in two layers: the trajectory optimiser (TO) and the trajectory predictor (TP). Essentially, the objective of the IPA is to take into account the constraints set in the route waypoints when integrating each phase, and apply the corresponding solution depending on the nature of the constraint that triggers the IPA. Therefore, the IPA forms a new layer by itself, which is placed in the middle of the TO and the TP.

First of all, as shown in the diagram of Figure 2.1, the TO gives a profile to the IPA, which then gives one phase to the TP. Next, the TP integrates the phase and gives the resulting trajectory back to the IPA. At this point, the IPA decides if it needs to take action applying the necessary correction, according to whether a constraint from the waypoints has stopped the integration or not. Afterwards, IPA layer gives the following phase to the TP and the process keeps repeating until all phases of the profile are integrated. Finally, the IPA gives the now fully integrated profile back to the TO.

To have a better understanding of how the IPA operates, a more detailed diagram can be seen in Figure 2.2. Firstly, the IPA takes a phase from the profile given by the optimiser and using the state vector of the aircraft at the current point of integration, locates it inside the route and fetches the waypoint constraints which apply to that leg. Then, it joins these constraints to the phase as end conditions and gives it to the predictor. The TP then integrates this phase until an end condition is triggered, and at that point it stops.

The integration stop can trigger due to various types of end conditions. Some of these conditions can be defined in the unimpeded profile (given by the user), in the optimisation layer by the TO, or in the IPA layer as a consequence of constraints found in the waypoints of the route. The latter are the constraints that the IPA layer is concerned about, so if the predictor is stopped by the trigger of another type of constraint, the IPA logic is not

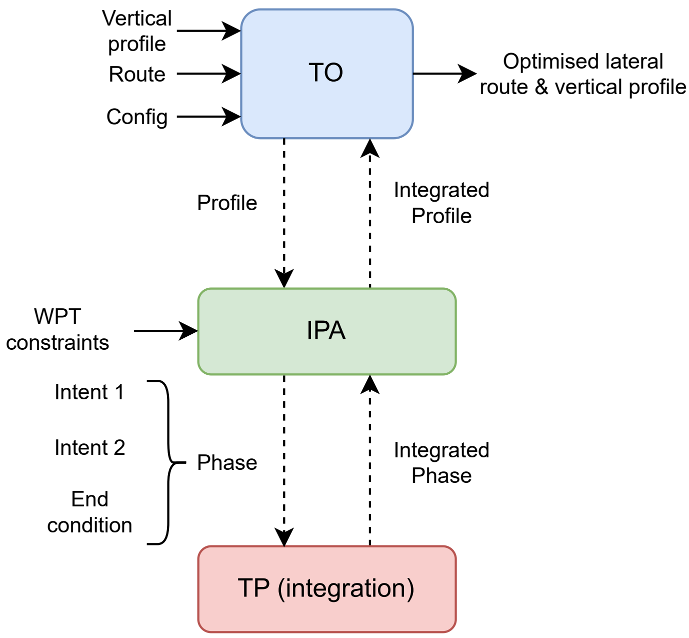


Figure 2.1: Integration of IPA layer in Dynamo's architecture

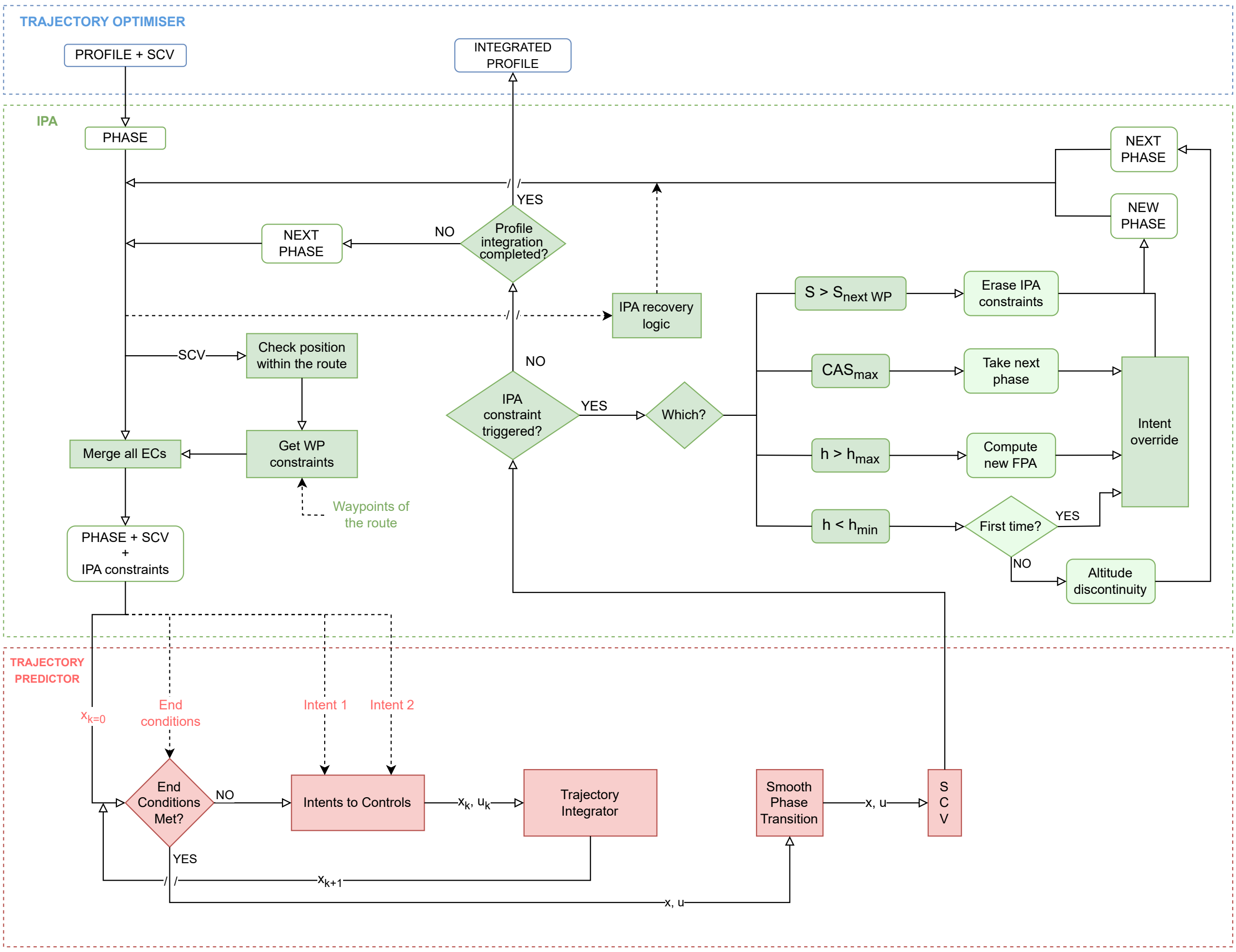


Figure 2.2: Dynamo layers and IPA internal operation

engaged. Then, once all profile is integrated, the optimisation layer will be the responsible for taking the next action, otherwise the next phase of the profile is integrated.

Then, if the end condition is an IPA constraint, the IPA applies a different solution method for each of the four cases. These are explained further below in section 2.3.. Finally, each case gives a new phase to be integrated, which is passed to the first part of the IPA and does all the process again, or it just jumps to integrate the next phase.

The approach of considering the IPA as a distinct layer enables the possibility of arbitrarily deactivating it when necessary. It works as an independent module of Dynamo, thus to optimize a profile without considering the chart constraints, the deactivation of this layer allows the software to bypass it. This is one of the most important perks of the IPA implementation and one of the main goals in the development, because as a result, numerous code problems are avoided, facilitating future development without encountering any additional difficulties.

Finally, the implementation can be divided in three big blocks:

- **Constraints propagation**: the IPA fetches the constraints and adds them to the original phase (see section 2.2.).
- **Constraint fulfillment logic**: the IPA identifies the type of constraint that is being violated by the unimpeded profile and applies a certain logic to resolve the issue (see section 2.3.).
- **Intent overrides**: the IPA overrides the original intents of the phase found in the unimpeded profile to adjust the trajectory in order to fulfil the constraints (this logic applies to each sub-case described within section 2.3.).

These three blocks are described in detail in the following sections of this chapter. It is worth noting that all the details below are explained from a high-level point of view, with the intention that there is no need to know the programming language used or the low-level details of the software.

2.2. Constraints propagation

Constraints set at the waypoints are to be met by the aircraft when it flies by, but in order to let the integrator know when should it stop the integration, these constraints must be evaluated throughout a certain period of integration and not just in the point of the waypoint. That is why a specific logic of propagation of these constraints must be applied. In other words, the program uses this logic to decide in which direction a specific constraint of a waypoint must be expanded, towards the next waypoint to be flown or towards the previous waypoint.

The three types of constraints, previously defined in Section 1.1., are divided in two types of propagation. First, constraints that set a maximum value (i.e., maximum altitude and maximum CAS) are propagated forwards, from the waypoint setting it towards the next waypoint in the route. If the next waypoint does not have an own maximum constraint of the same type, this constraint is propagated until the next waypoint, and so on and so

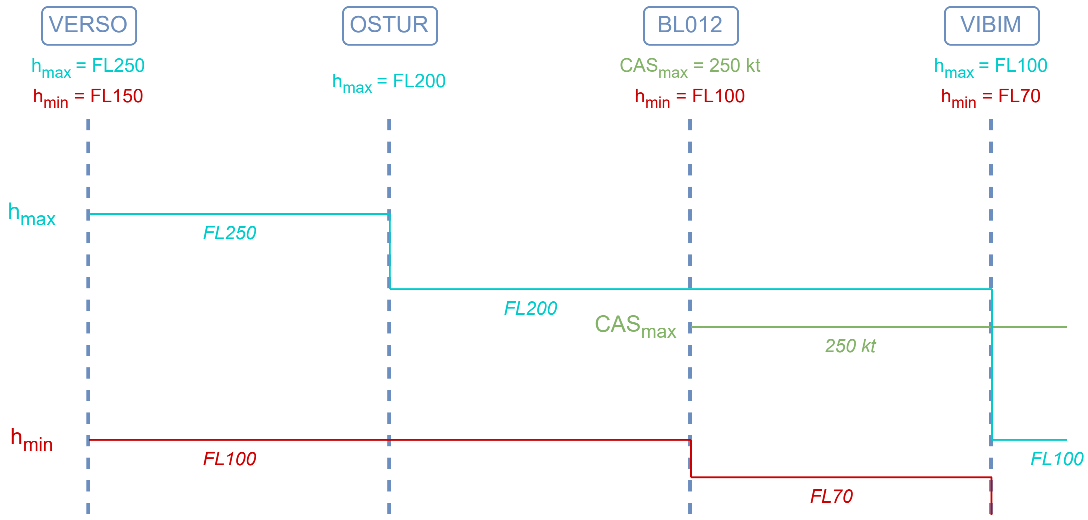


Figure 2.3: STAR LEBL RWY06L/R VERSO2E vertical profile limitations

forth until it reaches a waypoint that has a constraint of the same type, or the end of the route. These constraints are propagated forwards because as the aircraft is descending and reducing its velocity, it is able to surpass the maximum before reaching the waypoint, but after passing it by, aircraft has no need to go back up or increase the velocity. Note that only descent performances are considered; this logic would be different in other flight phases.

On the other hand, the minimum altitude constraint type is propagated backwards, from the waypoint setting the constraint towards the previous waypoint in the route, until it reaches another waypoint that sets a constraint of the same type, or it reaches the first waypoint of the STAR.

Figure 2.3 shows how the different constraints of a real navigation chart get propagated. For this example constraints are obtained from VERSO2E STAR of Barcelona-El Prat airport [11], which is the chart used for the verification and validation processes of Chapters 3 and 4 (see Figures 3.1 and 3.2). Firstly, maximum altitude constraints are propagated forwards, such as the h_{max} = FL200 set at OSTUR waypoint and propagated until VIBIM, where there is h_{max} = FL100. Then, taking a look at minimum altitude constraints, we can see that they are propagated backwards, such as h_{min} = FL100 set at BL012, which is expanded until VERSO, which has another minimum altitude constraint h_{min} = FL180. Finally, in case of ex. maximum CAS constraint, we can see waypoint BL012 set CAS_{max} = 250 kt onward.

2.3. Constraint fulfilment logic

From the trigger of one of the constraint types defined above, four different cases may arise. Depending on them, a particular solution will need to be applied. Thus, it is important to clearly identify which case is triggering the IPA logic. From now on these constraints are going to be called IPA constraints or IPA end conditions, in order to differentiate them from other constraints that have other origins, like the unimpeded profile set by the user.

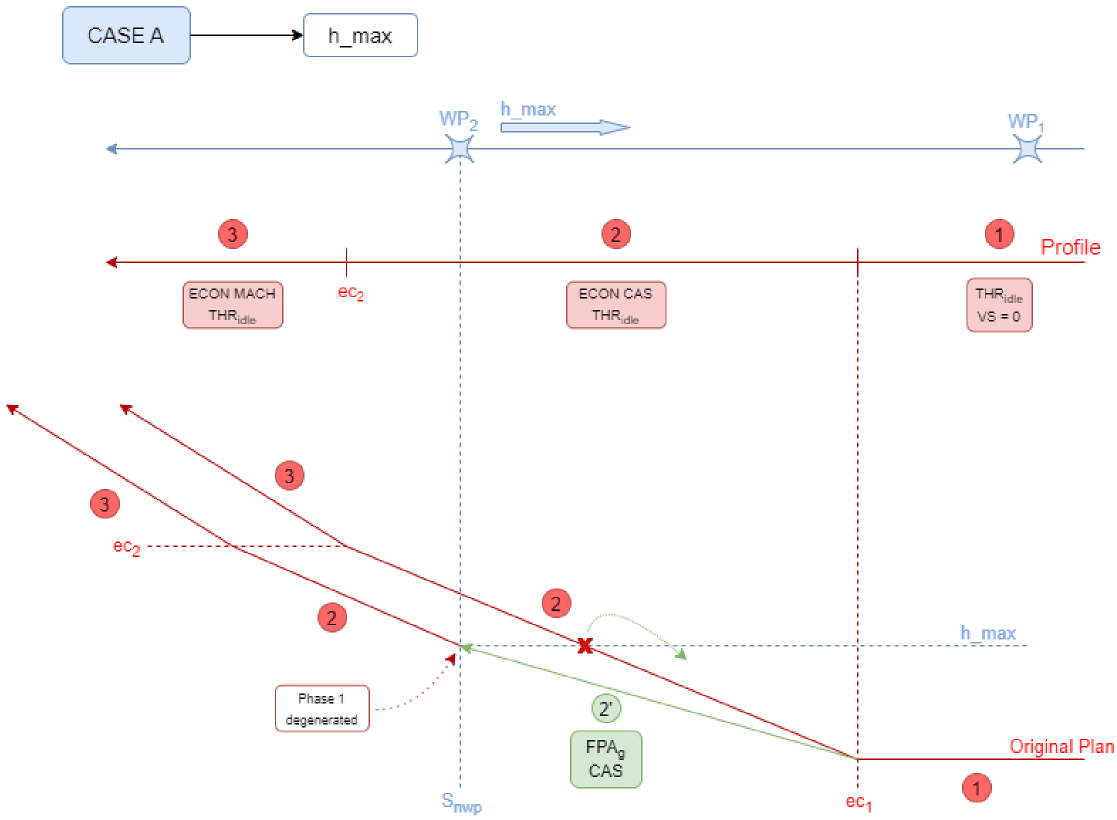


Figure 2.4: Integration profile solution for the first sub-case with an h_{max} constraint

Note that the integration of the descent profile is done backwards (from the runway threshold to the TOD). Hence, the schemes and the reasoning explained below are considered with the aircraft flying backwards.

2.3.1. Case A: Maximum altitude constraint

In this first case, the condition which makes the integration stop is a condition of type h_{max}. Two sub-cases are analysed, the first when the waypoint constraint is "deactivated" before reaching the end condition set by the unimpeded profile, and the second when the end condition is reached before the waypoint setting the h_{max} constraint.

2.3.1.1. IPA logic

In the first sub-case, called *Case A.I: Maximum altitude* and shown in Figure 2.4, the predictor integrates until it reaches the maximum altitude, when it resets the integration to the beginning of the phase and computes a new FPA_g. Then a new phase is created with the pair of intents set using intent override logic explained below in section 2.3.1.2.. This new phase is called 2' because it is the same as 2 but with different intents. Finally, when the integration reaches the waypoint, as the h_{max} condition is no longer active, the IPA phase recovery logic is engaged (see section 2.5.).

In the second sub-case, called *Case A.II: Maximum altitude with intermediate end condition* and shown in Figure 2.5, the end condition of phase 2 triggers before it reaches

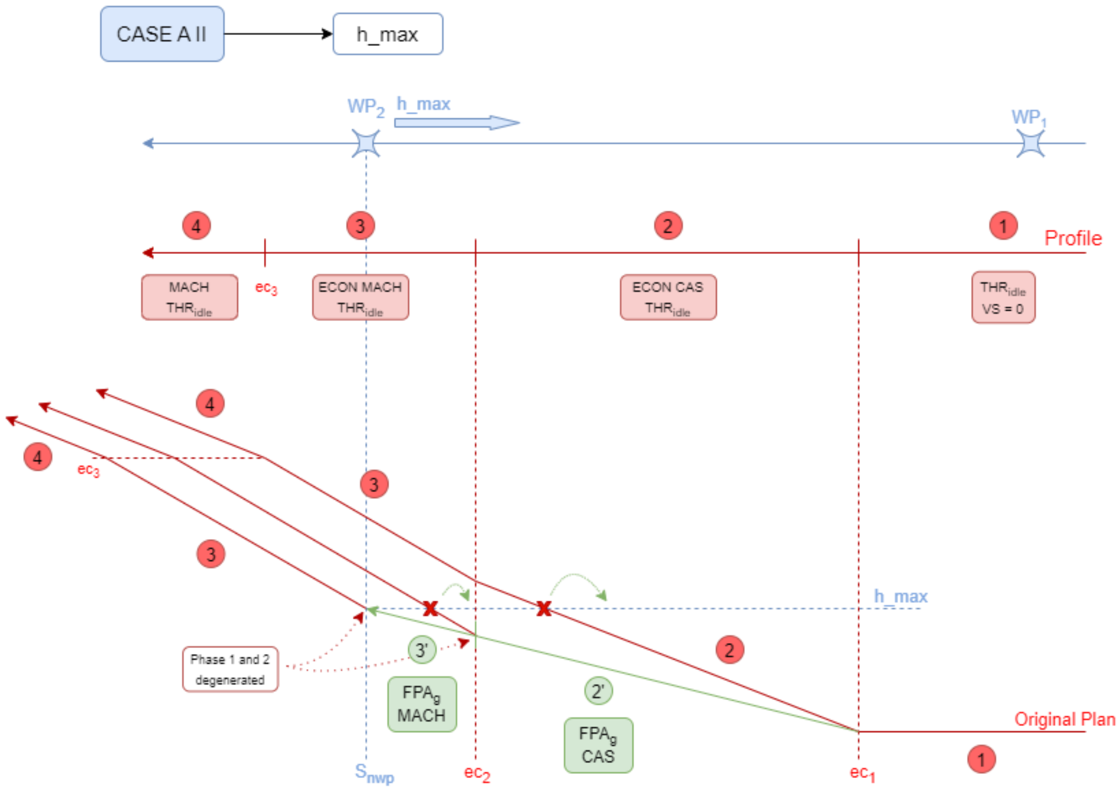


Figure 2.5: Integration profile solution for the second sub-case with an h_{max} constraint

the waypoint, and after the point where h_{max} is triggered. So it is the same as the previous sub-case up until it reaches ec_2, where the IPA deactivates and it tries to go back to the normal profile integrating phase 1 and 2 (both degenerated) and phase 3, due to IPA phase recovery logic. However, h_{max} constraint triggers again, so phase is reset again to the point of ec_2, and a new phase with new intents (see section 2.3.1.2.) is integrated until the waypoint is reached. At this point, IPA phase recovery logic is engaged again, placing phase 2 (usually degenerated, but not always) and phase 3 as the next phases to be integrated.

Note that a degenerated phase is one that when it begins to be integrated, the end condition is already met. This immediately stops the integration, resulting in an ”empty” phase.

2.3.1.2. Intents override

Concerning the intent override, for this case of maximum altitude the main intention of the aircraft is to fly with the new computed flight path angle with respect to the ground. Therefore, as shown in Table 2.1, the logic applied is to change the intent defining the movement of the elevator, and keep the velocity (Mach, CAS or ESF) constant. This way the aircraft flies through the phase at constant speed and with the maximum FPA not to surpass the h_{max} of the waypoint at the end of this phase (i. e. where the aircraft flies by the waypoint).

Profile phase intents		IPA h_{max} phase intents	
Elevator	**Throttle**	**Elevator**	**Throttle**
MACH CAS ESF	THR	FPA	MACH CAS ESF
VS	MACH CAS ESF	FPA	MACH CAS ESF
FPA	MACH CAS ESF	FPA	MACH CAS ESF
VS FPA ALT	THR	FPA $\leftarrow N/A \rightarrow$	VS THR

Table 2.1: Intent override for maximum altitude case

2.3.2. Case B: Minimum altitude case

In this case, where the integration is stopped by a condition of type h_{min}, there are two different sub-cases as well. One in which the aircraft is not able to arrive at the minimum allowed altitude to fly by that waypoint, and another one in which the aircraft is able to correct its flight path angle enough to have the sufficient altitude at the waypoint.

2.3.2.1. IPA logic

Shown in Figure 2.6, in this first sub-case, called *Case B.I: Minimum altitude with discontinuity*, the integration reaches the waypoint at an altitude lower than the minimum set by that waypoint, so it would need to ascend faster. In order to do so, the phase intent corresponding to the computation of throttle control is changed to idle Thrust (see section 2.3.2.2.), and this new phase is integrated again. In spite of this change, the new phase is not able to reach the minimum altitude, so an altitude jump at this value is done and the integration continues. Finally, the optimiser layer is informed of this discontinuity in the path. Likewise, as seen in section 1.2., when the FMS encounters this same situation it displays the message: ”TOO STEEP PATH”.

In the second sub-case, called *Case B.II: Minimum altitude* and seen in Figure 2.7, everything remains equal to the first sub-case, but when the new phase is being integrated, it reaches the waypoint at an altitude higher than the minimum set. In this situation, the IPA is able to correct the trajectory enough to be compliant with the h_{min} constraint. Hence, at the waypoint, IPA recovery phase logic is carried out, placing phase 1 and 2 as the next to integrate.

2.3.2.2. Intents override

For this case, the main intention of the aircraft is to fly a steep enough vertical profile in order to be at a higher altitude at the waypoint than the h_{min} constraint. Therefore, as

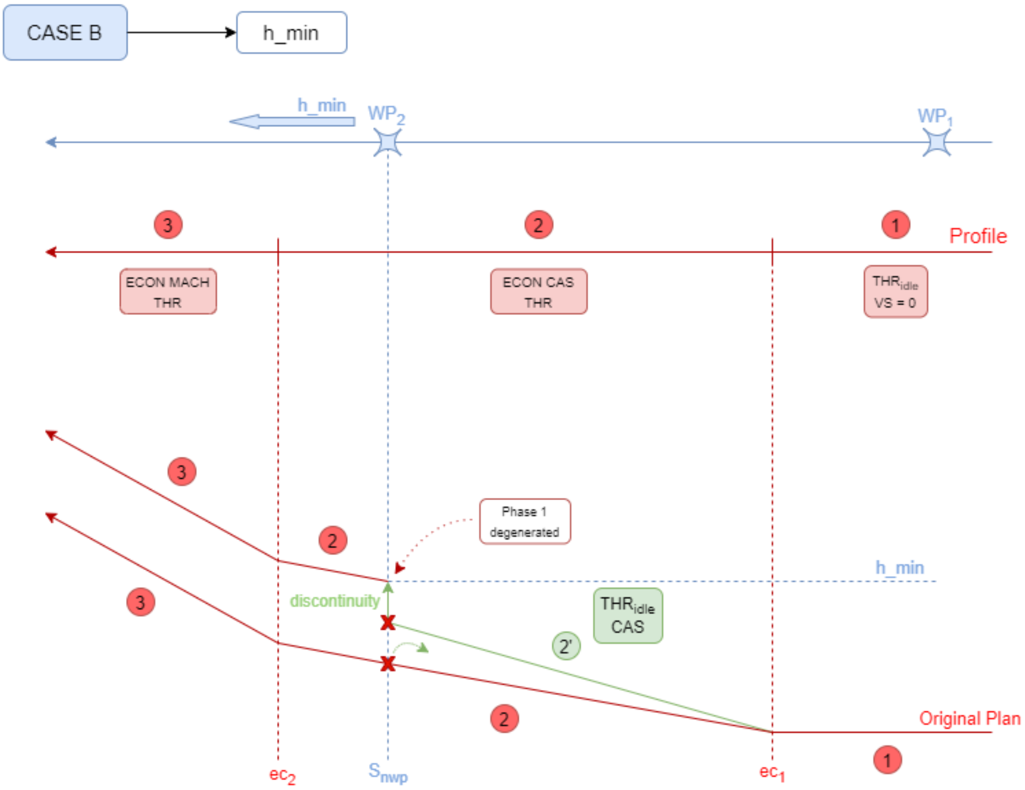


Figure 2.6: Integration profile solution for the first sub-case with an h_{min} constraint

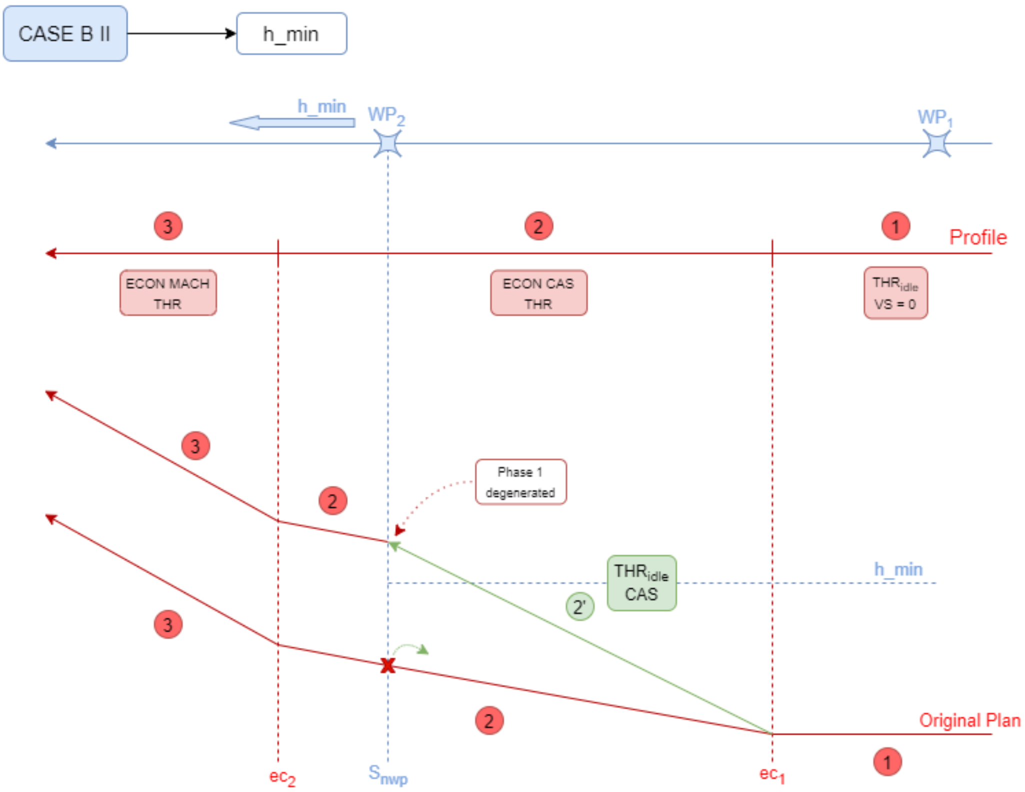


Figure 2.7: Integration profile solution for the second sub-case with an h_{min} constraint

Profile phase intents		IPA h_{min} phase intents	
Elevator	**Throttle**	**Elevator**	**Throttle**
MACH		MACH	
CAS	THR	CAS	THR_{idle}
ESF		CAS	
	MACH	MACH	
VS	CAS	CAS	THR_{idle}
	ESF	CAS	
	MACH	MACH	
FPA	CAS	CAS	THR_{idle}
	ESF	CAS	
VS		CAS	
FPA	THR	CAS	THR_{idle}
ALT		CAS	

Table 2.2: Intent override for minimum altitude case

shown in Table 2.2, the IPA enforces that the intent controlling the throttle actuator is $T = T_{idle}$ for any intent pair. It can happen that the aircraft already flies at T_{idle}, then when the IPA carries out the intent override, intents of the new phase will be the same as the ones of the old phase, and so the new integrated profile will be the same.

The ESF (energy share factor) is a value that relates the loss of altitude with the loss of speed throughout the period of integration of the phase. Hence, in this case, as we want to maximise the loss of altitude, the ESF intent variable is replaced for a constant CAS.

2.3.3. Case C: Maximum CAS case

In this case, the constraint triggered at the end of the integration is a constraint of type CAS_{max}. There are two possible sub-cases in this case too, one that the waypoint setting the CAS_{max} is encountered before the current phase is ended by the profile original end conditions, and another one in which the end condition is reached before arriving at the waypoint.

2.3.3.1. IPA logic

As it can be seen in Figure 2.8, in this first sub-case, called *Case C.I: Maximum CAS*, the integration goes on along phase 2, which is a level-off acceleration phase, until it reaches the CAS_{max}. At this point a new phase is added (3'), which is essentially the same as the next phase of the profile (3) with the only difference that the CAS of the aircraft is the CAS_{max} set by the waypoint, instead of the CAS that the aircraft would reach at the end of phase 2 (see section 2.3.3.2.). Then, the integration reaches the waypoint, the constraint disappears and the IPA phase recovery logic (see section 2.5.) is carried out, placing the phase previous to the one just finished (2) as the next phase to integrate. Here it is not necessary to add the original phase of the one just finished (3), because as it is not reached in IPA mode, it is already there.

In the second sub-case, called *Case C.II: Maximum CAS with intermediate end condition*

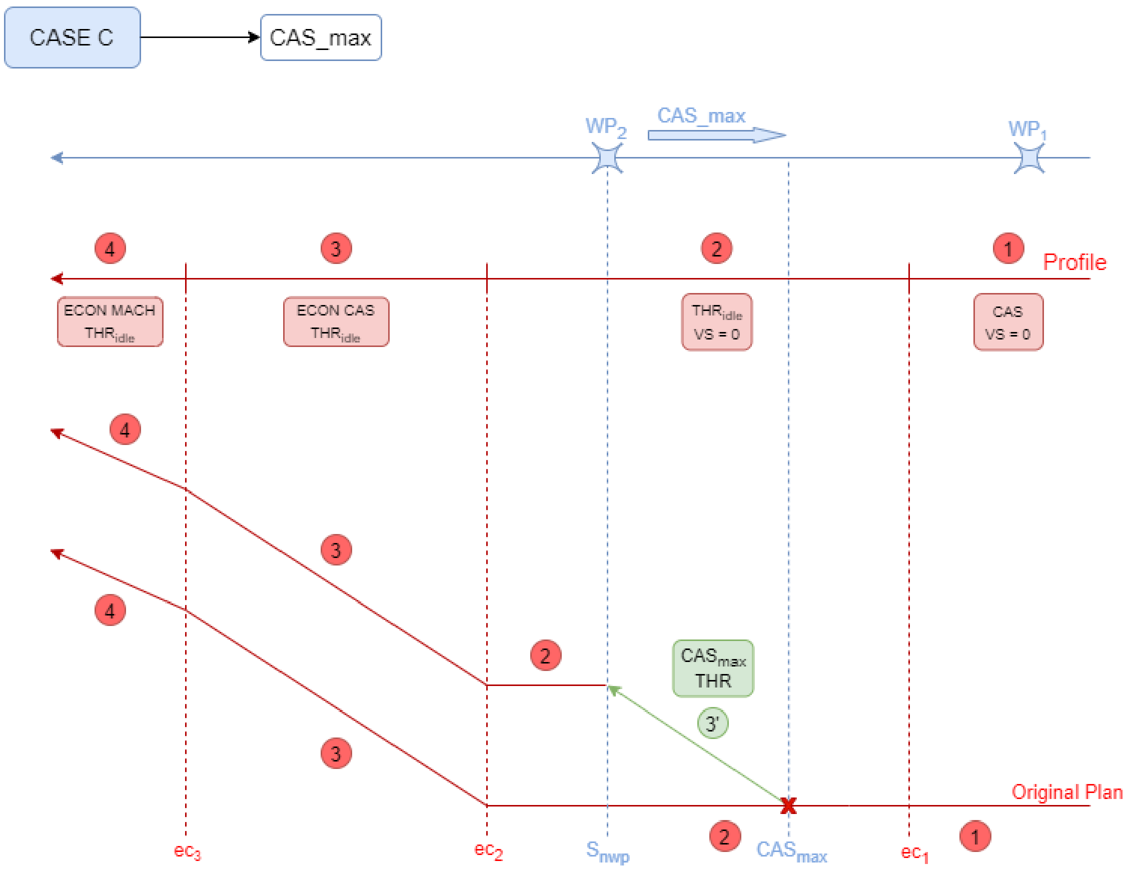


Figure 2.8: Integration profile solution for the first sub-case with a CAS_{max} constraint

and shown in Figure 2.9, after the trigger of CAS_{max} constraint a new phase (3') is added, as in the previous sub-case. Nevertheless, during the integration of phase 3', its end condition (ec_3) is reached before the waypoint. Therefore, there is an additional new phase, which is essentially the same as phase 4, treated as 4'. Finally, integration reaches the waypoint, CAS_{max} constraint deactivates and the IPA phase recovery logic is run. In this case, the last one added is phase 4', so it adds phase 3 and 4. However, phase 3 is degenerated because its end condition (ec_3) is already met, so it jumps straight for the next phase (4).

For the sake of better understanding, in Figure 2.10 the velocity profile is shown. In purple we have the maximum velocities, in red the original profile that would be followed if there were no IPA constraints triggered, and in green the profile integrated due to CAS_{max} constraint. The curved part of the lines are at constant CAS and the straight constant Mach.

Focusing on the green line, the new phase (3') makes the aircraft ascend until it reaches M_{ECON}, which is its end condition (ec_3). Then, we can confirm that phase 3 is degenerated after the waypoint, because it is already at M_{ECON}. We also note that the crossover altitude in the new profile is higher than in the original one, because it is ascending at a lower CAS.

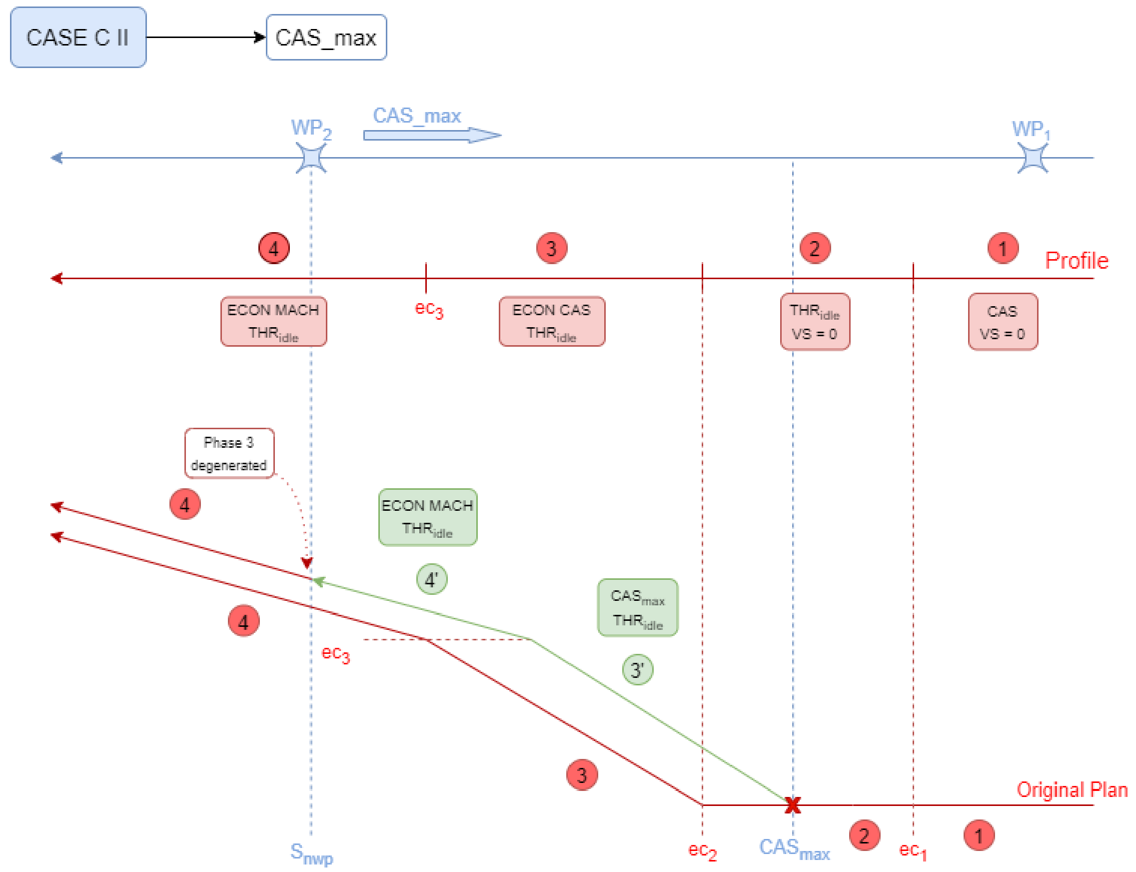


Figure 2.9: Integration profile solution for the second sub-case with a CAS_{max} constraint

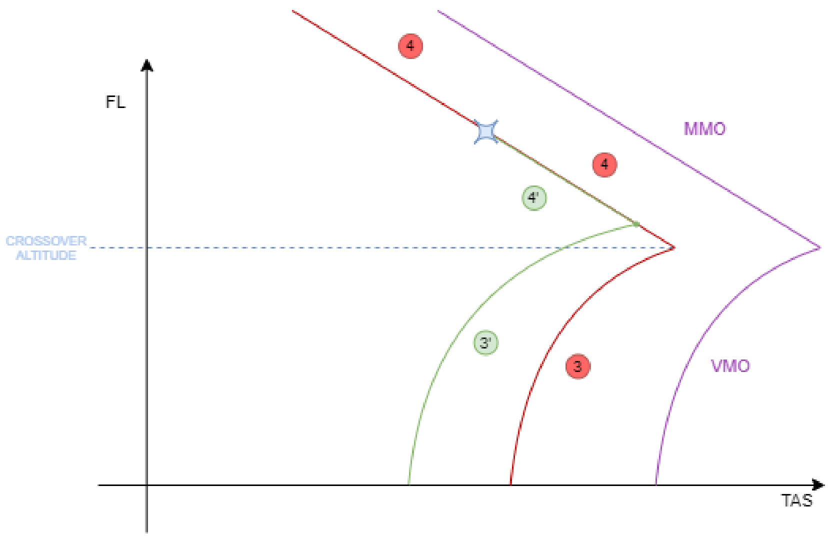


Figure 2.10: Velocity profile (TAS) as a function of altitude (FL)

2.3.3.2. Intents override

Finally, for the case of maximum CAS, as it can be seen in Table 2.3 the new phase intents variables do not change. However, the IPA makes sure that the intent has an equal velocity (CAS or Mach) which is equal to the maximum set by the constraint. Therefore, the

Profile phase intents		IPA CAS_{max} phase intents	
Elevator	**Throttle**	**Elevator**	**Throttle**
MACH CAS ESF	THR	MACH CAS CAS	THR
VS	MACH CAS ESF	VS	MACH CAS CAS
FPA	MACH CAS ESF	FPA	MACH CAS CAS
VS FPA ALT	THR	VS FPA ALT	CAS

Table 2.3: Intent override for maximum CAS case

intention of the aircraft is to fly at a CAS = CAS_{max} until the waypoint setting the limit. Here, ESF is also changed to CAS in order to maintain a constant CAS equal to the maximum.

2.4. Waypoint leg logic (Case D)

As it has been explained in section 2.2., IPA constraints apply to legs of the route. Therefore, the software needs to split the profile at every waypoint location to match the profile phases with the legs of the route.

As shown in Figure 2.11, in this case no phase is added: when the integration reaches the next waypoint, it fetches the constraints of the new leg and updates the phase which is being integrated. The phase update operation consists in erasing the previous IPA end conditions and adding the new constraints from the next leg. Some of these constraints are set by the initial waypoint and some others by the final waypoint of the leg. When the end conditions of the phase are updated, integration resumes at the same point where it had been interrupted.

It is important to mention that, even though this is not a proper IPA case, for readability purposes this case will be referred to as ”Case D” wherever appropriate throughout the remainder of this report.

2.5. Phase recovery logic

When the integration of a certain phase is interrupted due to the trigger of an IPA constraint, a new phase is added and integrated. Nevertheless, this original profile phase that is stopped has its purpose inside the vertical profile of the descent and can not be disregarded. Therefore, the IPA needs to recover this phase interrupted in order to continue with the original objective of the profile.

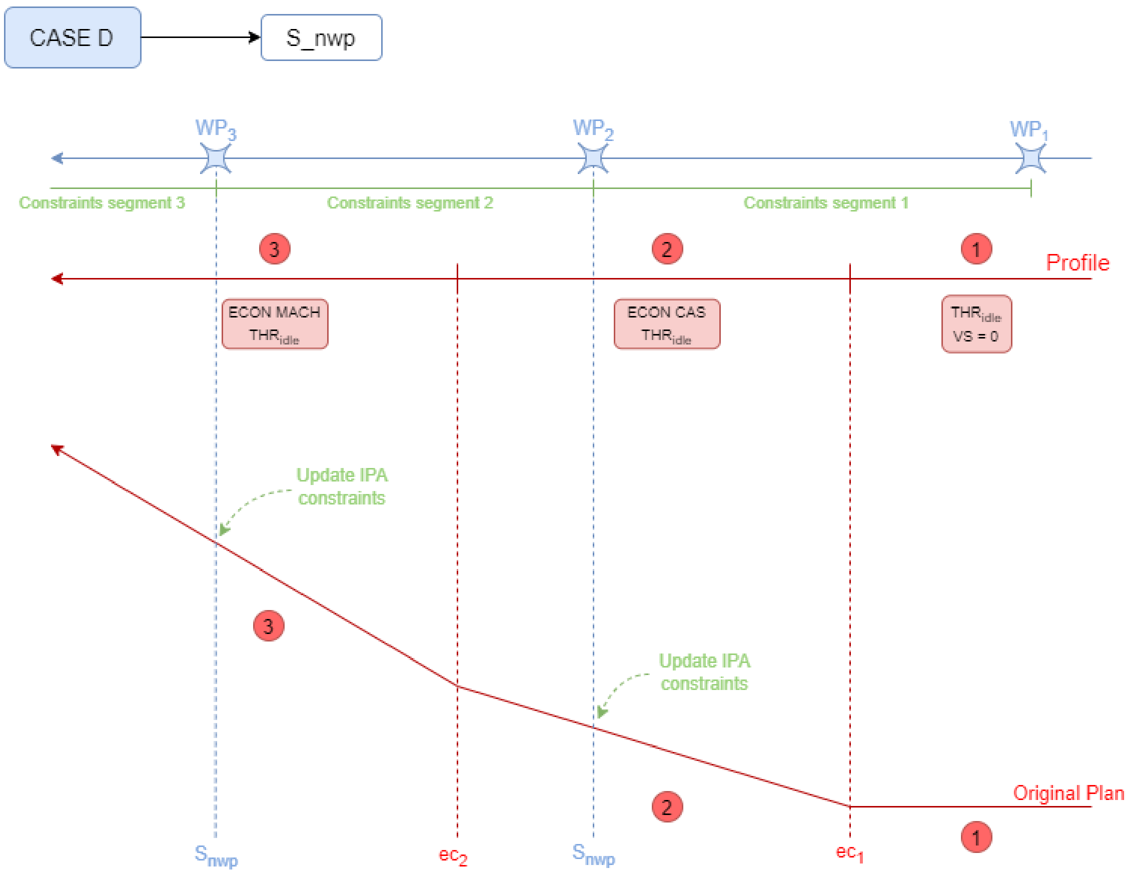


Figure 2.11: Integration profile solution for the S_{nwp} constraint

This feature consists of recovering the phase before the new one spawned by the IPA and the original from the one copied, and placing them respectively as the next phases to integrate. For instance, in sub-case of Figure 2.4, this logic recovers phase 1 and 2, and places them after 2'. However, phase 1, as the end condition 1 has already been reached, is degenerated (empty), and so the logic jumps right to phase 2 integrating it with the new IPA constraints of the next waypoint.

CHAPTER 3. IPA LOGIC VERIFICATION

Having provided a comprehensive explanation of the design logic of the IPA, this chapter aims at providing verification evidences of the proposed design. Hence, each of the seven (sub)cases explained in section 2.3. are individually tested running the IPA in a particular scenario, specifically designed to trigger the (sub)case under test. Thus, each (sub)case is treated individually with its own constraints, and the testing scenarios are tried to be as similar as possible in all (sub)cases. Yet, in some test there might be required to introduce slight changes to the unimpeded input profile and/or the route flown, in order to trigger the desired IPA (sub)case.

Ultimately, the objective of this chapter is to demonstrate that the IPA works properly for each different isolated case, where each specific constraint is tested. Later on in Chapter 4, a real case is tested with all constraints operative at the same time (i.e., validation of the IPA functionality).

3.1. Verification scenario

In order to run Dynamo it must be set: a scenario which comprises the route followed by the plane from the origin to the destination airports, the model of the aircraft used in the verification and the unimpeded vertical profile plan with all the phases. As the IPA implementation is designed for descent procedures, the verification is focused on this phase of the flight.

In all verification tests no wind is considered. Therefore, some variables which differ due to wind velocity and direction, such as GS (ground speed) and TAS (true airspeed), exhibit the same value. Considering wind conditions and further verifying the proper implementation of the IPA is left for future work.

3.1.1. Operational chart

Barcelona - El Prat airport is chosen because there is usually a high volume of traffic in its TMA. Hence, the ANSP typically applies many strategic constraints to the SIDs, STARs and approach procedures, which makes it an interesting scenario where to apply the IPA. Furthermore, the specific arrival and approach procedures to serve as a base to carry out the validation chosen are:

- Arrival: STAR2 RWY06L/R VERSO2E. Lateral route in Figure 3.1 and vertical waypoint constraints in Figure 3.2
- Approach: TRAN2 RWY06L VIBIM1E. Lateral route in Figure 3.3 and vertical waypoint constraints in Figure 3.4

The route followed by the aircraft is the one depicted in the charts, starting the arrival with the STAR VERSO2E and following with the approach VIBIM1E. An FMS would plan the descent taking into account that the aircraft follows these procedures, so for this IPA verification no other procedures, such as a holding, are considered. At the end of the approach, as this is a tromboning procedure, the aircraft has different waypoints to start the base leg

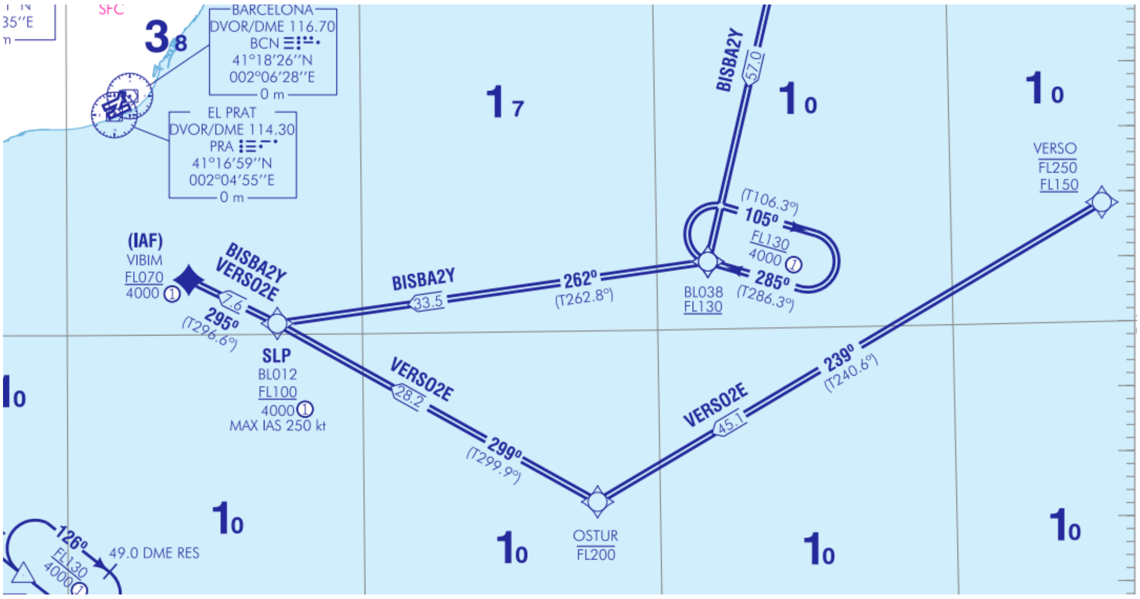


Figure 3.1: STAR2 RWY06L/R VERSO2E (Source: Spanish AIP [11])

VERSO2E											
001	IF	VERSO	–	–	-1.2	–	–	+FL150 -FL250	–	–	RNAV1
002	TF	OSTUR	–	239 (240.6)	-1.2	45.1	–	-FL200	–	–	RNAV1
003	TF	BL012	–	299 (299.9)	-1.2	28.2	–	+FL100	-250	–	RNAV1
004	TF	VIBIM	–	295 (296.6)	-1.2	7.6	–	+FL070	–	–	RNAV1

Figure 3.2: STAR2 RWY06L/R VERSO2E waypoint constraints (Source: Spanish AIP [11])

depending on when the ATC gives the clearance to proceed to the interception of the final leg. In this verification process, it is considered that the airspace is not congested and ATC gives the shortest clearance. This way, the worst-case scenario is being assumed, because granting the aircraft clearance to approach sooner gives less time and space for trajectory corrections.

Hence, taking a look at Figure 3.3, the trajectory to be considered when planing the descent is: VIBIM → BL546 → BL542 → BL538 → BL517 → ASTEK → RWY06L.

3.1.2. Aircraft

The aircraft model used in this project is the Airbus A320-231. This model is chosen because it is from a local manufacturer and a very commonly used model by airlines.

In order to model and simulate its behaviour, its performance parameters over the entire flight envelope are taken from the Base of aircraft data (BADA) provided by Eurocontrol [13], an international reference for aircraft performance modelling for trajectory prediction and simulation. This model is used to compute parameters such as the lift coefficient, the drag force, the idle thrust, or the stall speed. However, a more in-depth look into this performance models is not provided, as this is not the objective of this project.

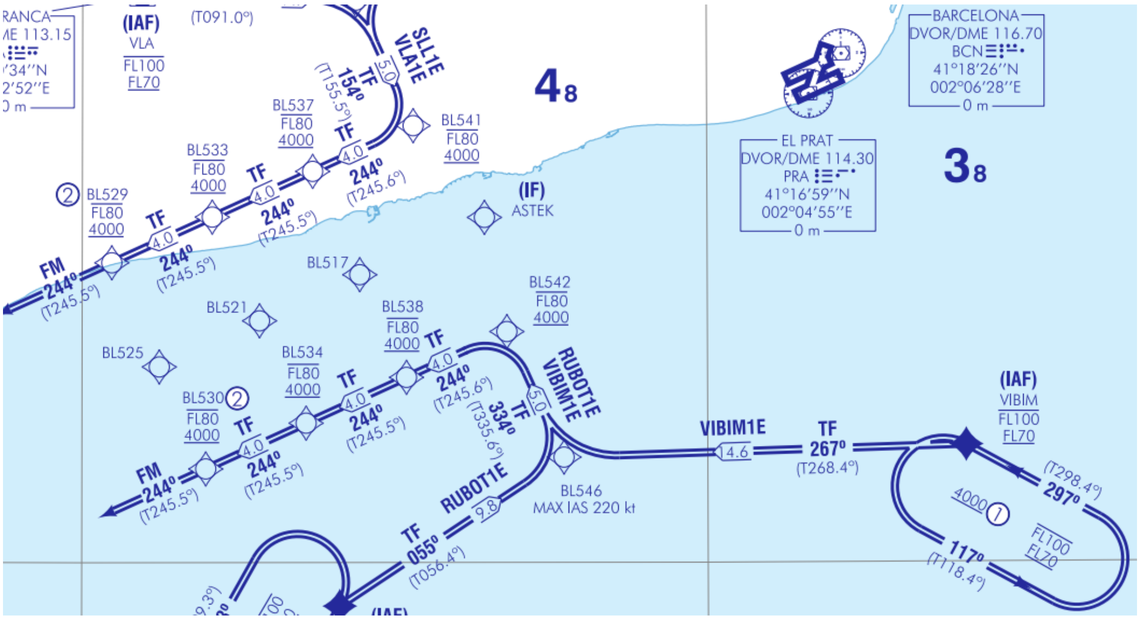


Figure 3.3: TRAN2 RWY06L VIBIM1E (Source: Spanish AIP [12])

VIBIM1E											
001	IF	VIBIM	–	–	-1.2	–	–	-FL100 +FL070	–	–	RNAV1
002	TF	BL546	–	267 (268.4)	-1.2	14.6	R	–	-220	–	RNAV1
003	TF	BL542	–	334 (335.6)	-1.2	5.0	L	-FL080 +4000	–	–	RNAV1
004	TF	BL538	–	244 (245.6)	-1.2	4.0	–	-FL080 +4000	–	–	RNAV1
005	TF	BL534	–	244 (245.5)	-1.2	4.0	–	-FL080 +4000	–	–	RNAV1
006	TF	BL530	–	244 (245.5)	-1.2	4.0	–	-FL080 +4000	–	–	RNAV1
007	FM	–	–	244 (245.5)	-1.2	–	–	-FL080 +4000	–	–	RNAV1

Figure 3.4: TRAN2 RWY06L VIBIM1E waypoint constraints (Source: Spanish AIP [12])

3.1.3. Unimpeded profile

The profile of the flight is the group of phases that the aircraft follows, in this project, during only the descent. As it is stated before, each phase has its own intents which are the parameters that describe the (intended) behaviour of the aircraft.

Although the profile is flown all together, it can be split in two parts: the initial approach, from the TOD to the FAP (final approach point); and the final approach, from the FAP to the landing runway. This difference is made because the final approach is flown following the glidepath of the ILS, so the aircraft is already flying a geometric path. Therefore, it would not make much sense to impose altitude restrictions there. Likewise, velocity constraints are not imposed, because this flight phase is already very critical to be subjected to any additional limitations. Consequently, as no constraints apply to this part of the approach, the IPA mainly acts in the initial part.

Taking a look at Figure 3.5, there are: in red squares the phase's identifications (names), in white the name of the block, in blue the pair of intents, in green the end conditions of each

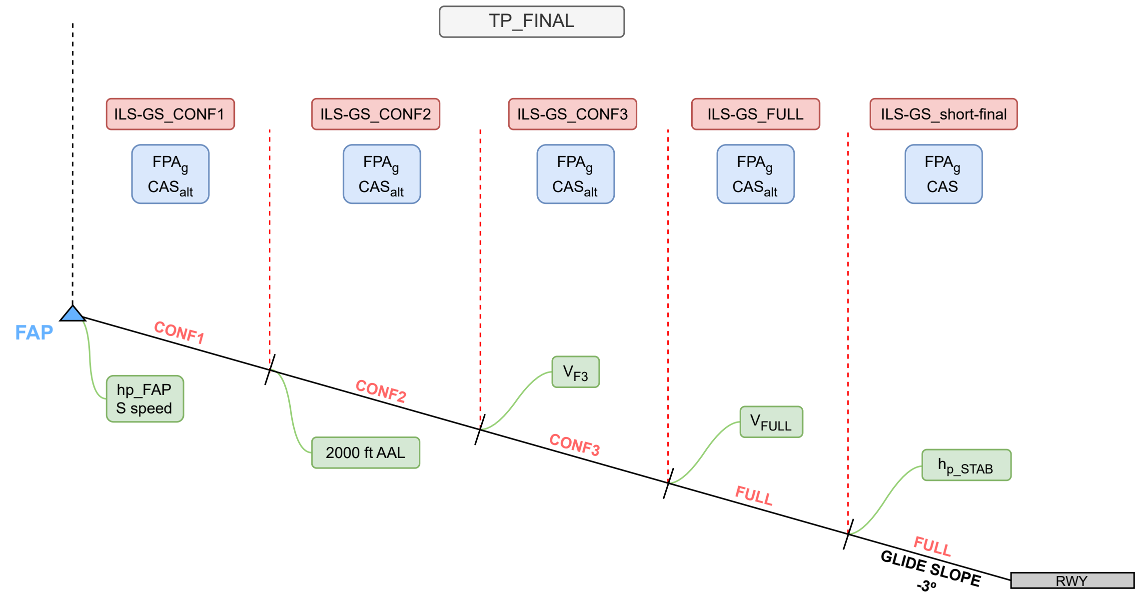


Figure 3.5: Final approach unimpeded profile phases

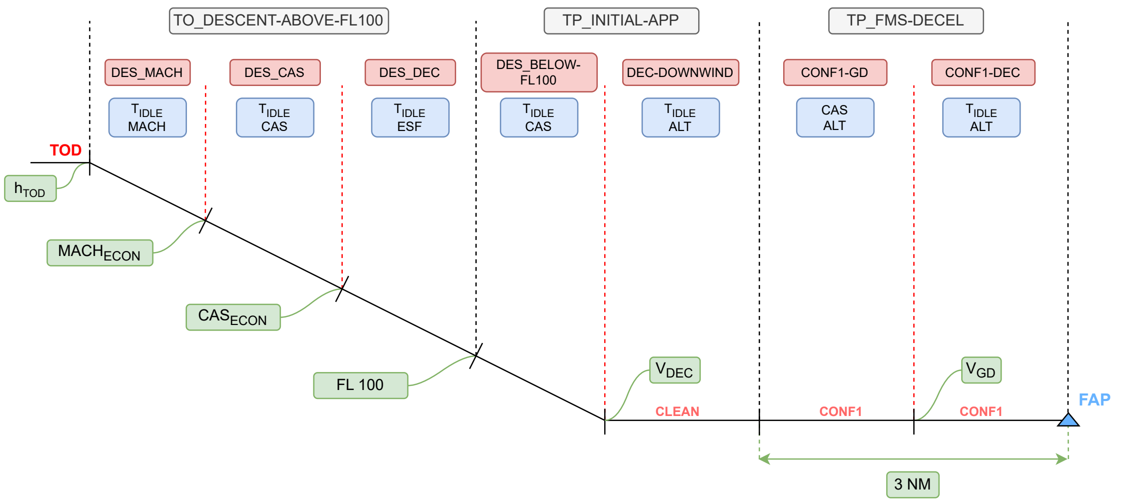


Figure 3.6: Initial approach unimpeded profile phases

phase and in red on the profile path the configuration of the aircraft (i.e., hiper-lift devices). As intents, in every phase the aircraft tries to fly with a FPA_g of -3º, because it is the glide slope published in the ILS procedure. Moreover, every phase has as a second intent a constant CAS acceleration over altitude, apart from the last one, which has a constant CAS over time. The speeds, such as V_{F3} or V_{FULL}, are taken from the aircraft performance model explained in section 3.1.2..

On the other hand, in Figure 3.6, there is the first part of the descent procedure, which is where the IPA gets involved. There are three different blocks: one for the first part of the descent (TO_DESCENT-ABOVE_FL100), one to start the approach procedure (TP_INITIAL-APP), and the last one mainly to decelerate before entering the last part of the approach (TP_FMS-DECEL). Even though it is only written in DEC-DOWNWIND phase, from this phase to the start the configuration of the aircraft is CLEAN, all flaps and landing gear

VARIABLE	VALUE
h_{TOD}	FL380
$MACH_{ECON}$	0.79
CAS_{ECON}	300 kt
V_{DEC}	250 kt
V_{GD}	APM
hp_{FAP}	3000 ft
V_{F3}	APM
V_{FULL}	APM
hp_{STAB}	1000 ft

Table 3.1: Values of each profile variable

retracted. Finally, even though there are end conditions with optimal values, such as CAS_{ECON}, which should be computed from aircraft models, in order to have a clear monitoring of the scenario, in this validation chapter these velocities are imposed to arbitrary values.

As a reminder, Dynamo runs the integration of the profile backwards, which means that it starts in the runway and ends at the TOD. Hence, the end conditions of the phases are placed according to the direction of the integration, not the direction of the aircraft performance.

Values given to the different variables of end conditions are shown in Table 3.1. Values which are designated ”APM” are computed using the performance models explained in section 3.1.2., so the value may vary from case to case. In addition, $MACH_{ECON}$ and CAS_{ECON} shown in the table are representative values of the ECON speeds that would be computed if optimal values were interesting.

3.1.4. Particular scenario setups

Aiming at a comprehensive IPA logic verification, some modifications to the original scenario presented above must be done in order to assure that the proper (sub)case triggers. This is due to various reasons, detailed in each of the sections below, but principally to have more readable and easier to understand results as well.

As it can be seen in Figure 3.7, there are three cases that have one waypoint changed. In addition, these scenarios follow the original route (white), but have a part of their route altered, each one depicted in a different colour. To clarify, apart from the coloured segments, all trajectories are identical, but the modified ones are overlapped by the original (white). Further details given in each particular scenario's setup section below.

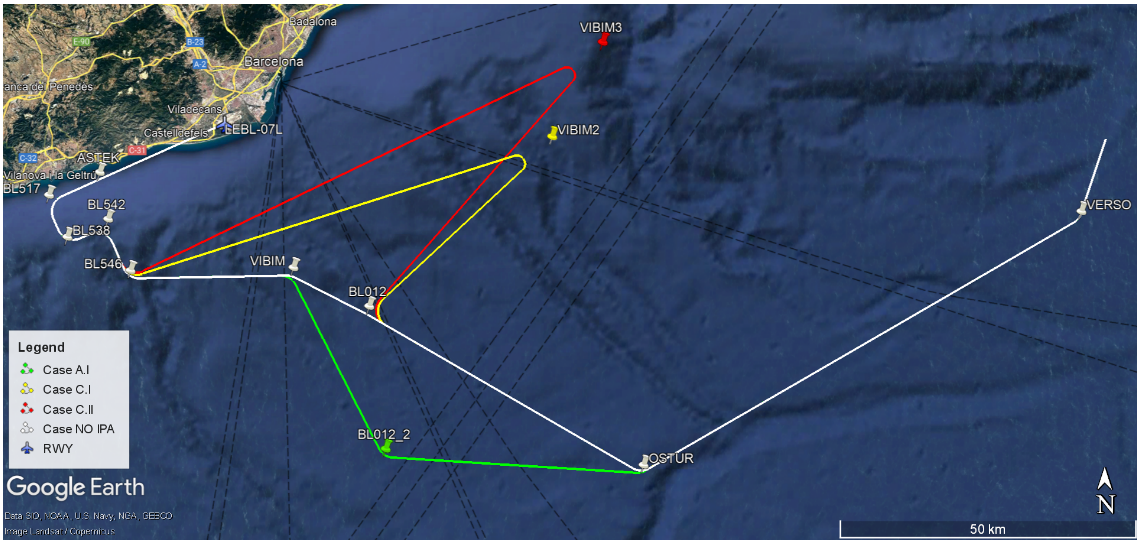


Figure 3.7: Descent original lateral route trajectory modifications

3.1.4.1. Case A.I: Maximum altitude

To validate this sub-case, the maximum altitude is set at 18,000 ft at OSTUR waypoint, which is propagated until the end of the route as there are no more constraints. In this particular case, nothing else needs to be changed from the original scenario.

3.1.4.2. Case A.II: Maximum altitude with intermediate end condition

As it has been explained in section 2.3.1., this is an exceptional scenario, which needs to be treated accordingly. Taking a look at Figure 3.6, it can be seen that phases which may be involved in this scenario are from DES_BELOW-FL100 to the beginning of the approach, because they are the phases in which the altitude is varying. However, the end conditions of these phases are altitudes or velocities, which makes this scenario unfeasible with these variables.

If the end condition is an altitude, when the new phase 2' (see Figure 2.5) is being integrated, it always reaches the waypoint before the end condition, because it is lower. In addition, if the end condition is a velocity, as it depends on altitude the same reasoning applies.

Therefore, in order to reproduce this particular sub-case, the unimpeded profile is slightly modified: h_{max} is set to 11,000 ft at BL012, and the end condition variable of phase DES_DEC is changed from CAS_{ECON} to a distance of 55 NM from the starting point (runway). As a result, as the end condition does not depend on altitude, the IPA is triggered in the point of interest.

Then, in order to have results that can be more easily studied, the location of waypoint BL012 is changed from 41º0'52"N 2º21'21"E to 40º48'12"N 2º23'23"E. Due to this relocation, the waypoint is further from the starting point, and so further from the trigger point of the IPA constraint. Therefore, the IPA is active for a longer time. This new waypoint is called BL012_2 and the difference in lateral trajectory from the original route can be seen in Figure 3.7, where this case's route modification is depicted in green.

3.1.4.3. Case B.I: Minimum altitude with discontinuity

In this sub-case, explained in section 2.3.2., the only constraint set up is an h_{min} of 26,000 ft at OSTUR. This scenario is set up like this because in the no-IPA profile integration, the aircraft passes by this waypoint at around 23,800 ft. In consequence, a higher minimum altitude value is imposed.

3.1.4.4. Case B.II: Minimum altitude

In this sub-case scenario, explained in section 2.3.2., the original unimpeded profile needs to be slightly modified in order to change sufficiently the slope of the trajectory. The minimum altitude is set at 21,000 ft at OSTUR again, and the thrust intent of phase DES_CAS is changed from $T = T_{IDLE}$ to $T = 2 \cdot |T_{IDLE}|$[1], so it is not flying at the minimum thrust, producing a shallower descent. This phase in particular is chosen to be tweaked because it is the one being integrated when the aircraft flies by the evaluated waypoint, OSTUR.

3.1.4.5. Case C.I: Maximum CAS

In this sub-case, explained in section 2.3.3., the only constraint that is set up is a CAS_{max} = 265 kt at waypoint VIBIM. This value is chosen because when the aircraft flies by VIBIM in the original unimpeded profile with the IPA deactivated, it has a CAS=300 kt. In addition, in order to make the results more visible and easier to read, waypoint VIBIM is moved from 41º4'15"N 2º12'23"E to 41º16'0"N 2º42'50"E and renamed to VIBIM2. The difference in the route between the original waypoint location and the modified one can be seen in Figure 3.7, where the yellow trajectory is the one integrated in this sub-case.

3.1.4.6. Case C.II: Maximum CAS with intermediate end condition

For this second sub-case, explained in section 2.3.3., the constraint set up is CAS_{max}=280 kt at waypoint VIBIM. As well as in the previous sub-case, the original location of VIBIM is changed, from 41º4'15"N 2º12'23"E to 41º24'24"N 2º48'47"E. This relocation can be seen in Figure 3.7, where the modified trajectory is in red. This new waypoint is renamed to VIBIM3 because it is located a little bit further than VIBIM2. As in the other scenario, this change is made with the objective of having clearer results.

In addition, taking look at Figure 2.9, a second activation of the IPA must be enforced when the end condition of the first new IPA phase is reached. Therefore, phase DES_CAS end condition is modified from M_{ECON}=0.79 to M_{ECON}=0.6. As a result of this change and the relocation of VIBIM, the end condition is triggered before arriving at the waypoint, as desired to test this specific sub-case in the IPA logic.

[1] In BADA, the APM used in this project, the aircraft thrust model also considers the contribution in aerodynamic drag generated by the engine nacelle. In low thrust situations (such as idle conditions) this drag could be bigger than the thrust itself. Hence, the modelled "thrust" by this APM could yield to negative values.

CASE	PROFILE MODIFICATION	CONSTRAINT	WAYPOINT	SOLUTION
Case A	—-	$h_{max} = 18000ft$	OSTUR	New FPA_g
Case A II	Relocation of BL012 DES_DEC: ec. $CAS_{ECON} \longrightarrow dist = 55NM$	$h_{max} = 11000ft$	BL012	
Case B	—-	$h_{min} = 26000ft$	OSTUR	$T = T_{IDLE}$ ERROR: TOO STEEP
Case B II	DES_DEC: $intent = 2 \cdot T_{IDLE}$	$h_{min} = 21000ft$	OSTUR	$T = T_{IDLE}$
Case C	Relocation of VIBIM	$CAS_{max} = 265kt$	VIBIM2	Next phase: CAS_{max}
Case C II	Relocation of VIBIM DES_CAS: ec. $M_{ECON} = 0.6$	$CAS_{max} = 280kt$	VIBIM3	Next phase: CAS_{max} and M_{ECON}

Table 3.2: IPA verification scenario setups

3.1.4.7. Particular scenario setups summary

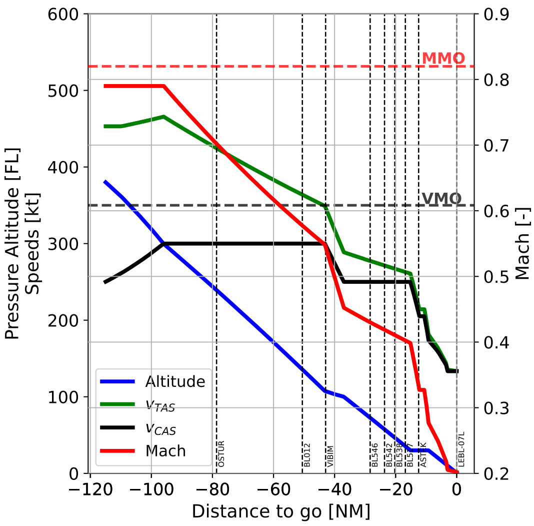
To conclude the scenario setup section, Table 3.2 shows a summary of all the (sub)cases addressed in this verification, where the most important information of each particular setup is shown, as well as the automatic solution provided by the IPA to solve each constraint infringement (explained in the following section).

3.2. Verification results

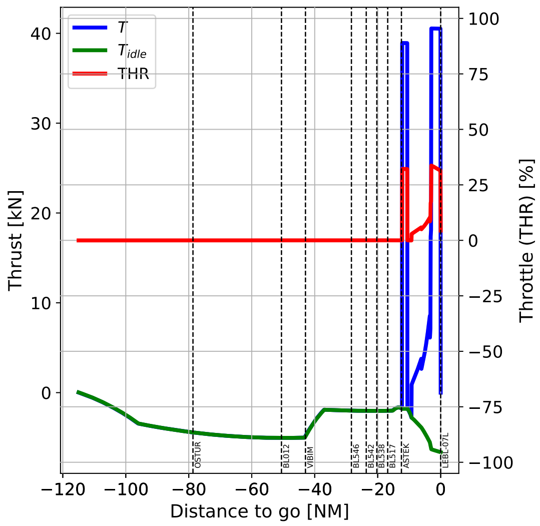
With all the verification scenarios explained, each IPA (sub)case is analysed independently and the results are reported in this section.

In Figure 3.8 it is shown the variation of many different flight parameters throughout the descent. This profile is flown with no constraints applied to the waypoints of the route, this is what Dynamo outputs without the activation of the IPA (i.e. unimpeded profile). All the following results are compared to this profile, to see the influence of the IPA in the planning of the descent trajectory vertical profile.

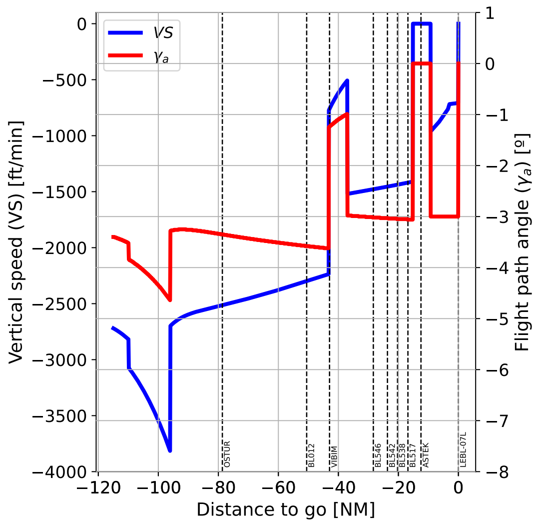
It is important to mention that all the values of the results, both in this chapter and the next one, are taken from a Dynamo output called Flight Data Recorder (FDR). This is a table with all the variables values (altitude, FPA, VS, CAS, Mach, etc.) for every step of integration of the profile. As this is a very big source of data which is unfeasible to show in this report, only some particularly interesting variables are analysed: altitude, speeds (CAS and TAS), thrust (T and T_{idle}), throttle, vertical speed and flight path angle.



(a) Altitude and speed profiles



(b) Engine thrust and throttle



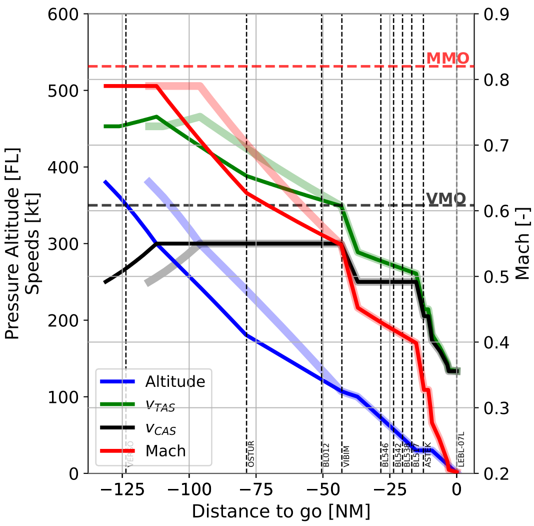
(c) Vertical speed and aerodynamic flight path angle

Figure 3.8: Vertical trajectory without IPA activation (i.e., unimpeded profile)

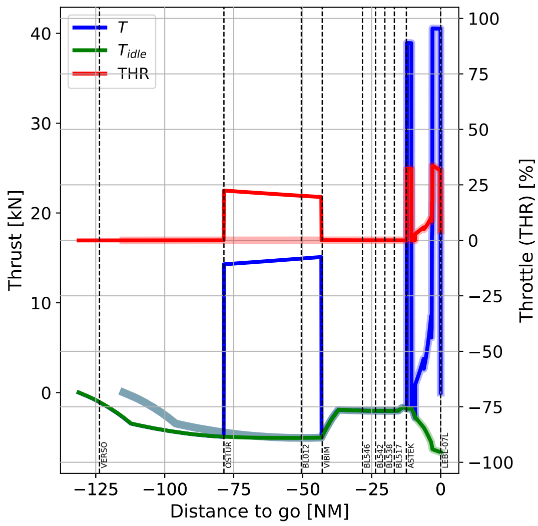
3.2.1. Case A.I: Maximum altitude

This first case refers to the situation explained in section 2.3.1., where the aircraft arrives at an integration point in which the pressure altitude is higher than the constraint that applies to that leg of the route. Therefore, the aircraft needs to fly again the last phase with an FPA_g low enough to not surpass the altitude constraint at the waypoint.

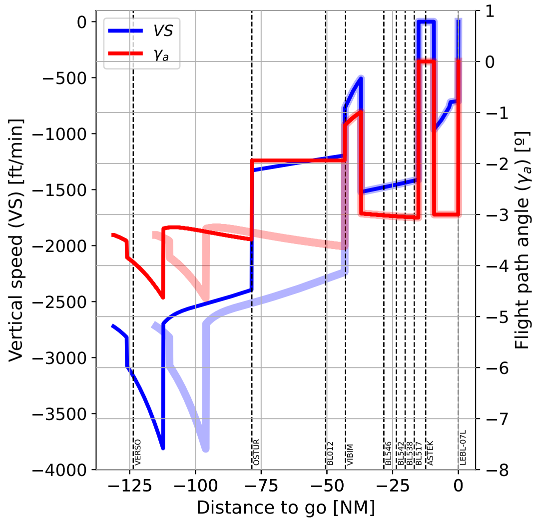
In Figure 3.9a it can be seen the comparison between the corrected profile (solid lines) and the unimpeded profile (transparent lines). It can be seen that there is a variation in altitude at OSTUR waypoint, where the initial altitude is 23,900 ft and the corrected is 18,000 ft. This change makes the aircraft arrive at TOD around 15 NM further, and the acceleration of TAS and Mach are also reduced. Even though it seems that the new phase where the aircraft climbs (in backwards) slower starts at waypoint VIBIM, it does not, the start of phase DES_CAS (the one modified) is really close to VIBIM.



(a) Altitude and speed profiles



(b) Engine thrust and throttle



(c) Vertical speed and aerodynamic flight path angle

Figure 3.9: Case A.I Vertical trajectory comparison (original profile in transparent lines and the IPA solution in solid lines)

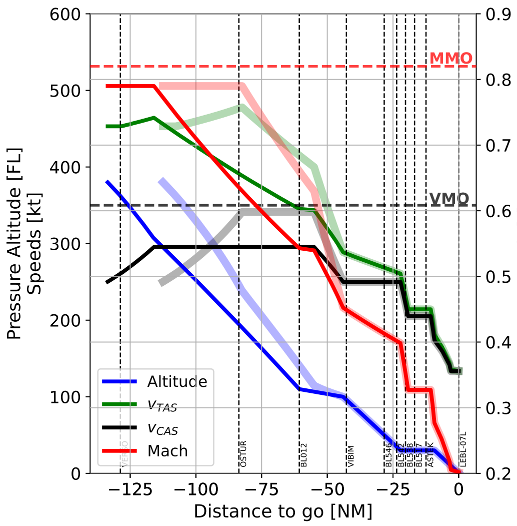
Finally, in Figure 3.9c it can be seen how the FPA_g[2] is modified throughout the new phase right before OSTUR. In this phase FPA_g changes from an around -3.5°, to a -1.94°. Due to this decrease in the FPA_g, it can also be seen how the VS (vertical speed) is also reduced from around -2400 ft/min to around -1700 ft/min.

3.2.2. Case A.II: Maximum altitude with intermediate end condition

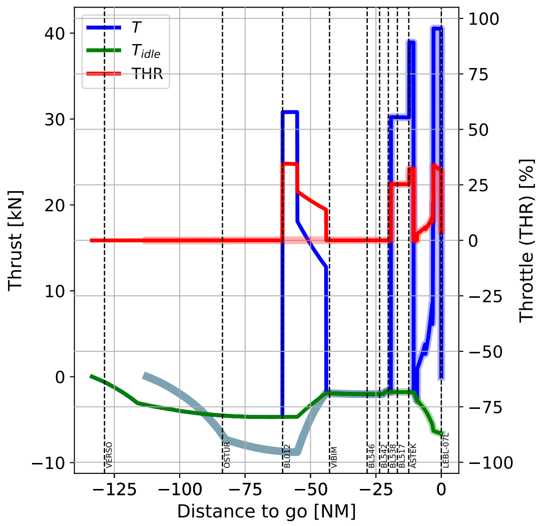
In this second sub-case of a trigger of a maximum altitude constraint, explained in section 2.3.1., the end condition is reached before the waypoint but after the trigger of h_{max}.

In Figure 3.10a, it can be seen how waypoint BL012_2 is further than 55 NM, so the end

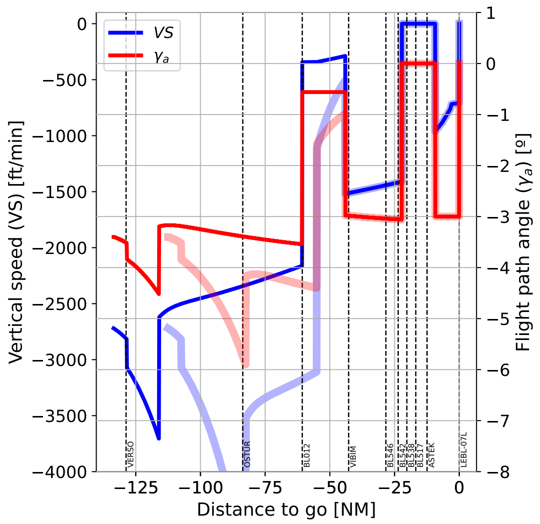
[2] As there is no wind model applied to these (sub)cases, $FPA_a = FPA_g$.



(a) Altitude and speed profiles



(b) Engine thrust and throttle



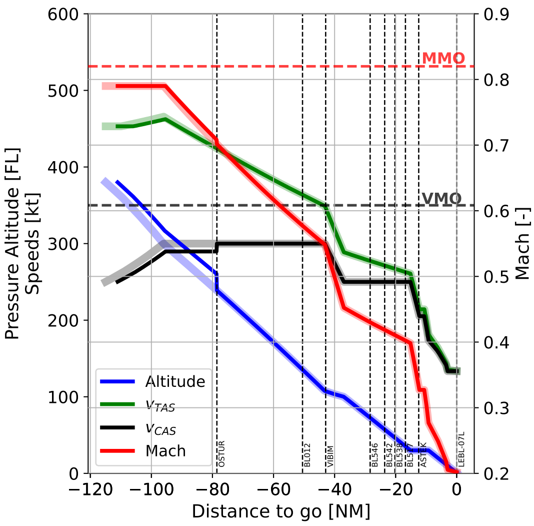
(c) Vertical speed and aerodynamic flight path angle

Figure 3.10: Case A.II Vertical trajectory comparison

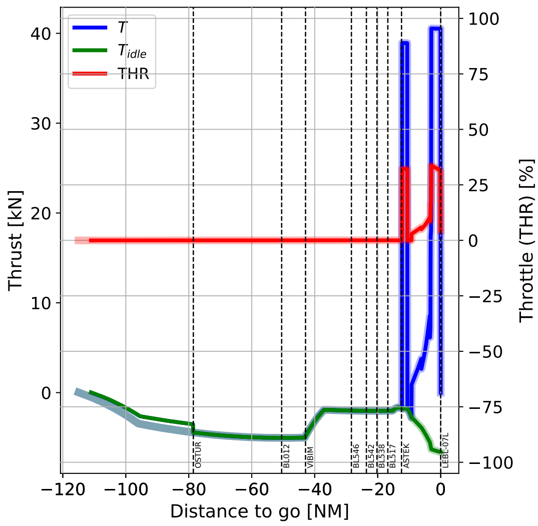
condition triggers before and adds the next phase. Consequently, the IPA interacts with the profile adding two modified phases. As it can be seen, firstly altitude profile gets modified to a slower rate of descent due to the IPA forcing both phases to have a specific FPA to satisfy the constraint at BL012. While in the modified profile the altitude is forced to be 11,000 ft at BL012, while in the original unimpeded profile h $\approx$ 14,000 ft.

Then, we can see the change to the second IPA phase when CAS becomes constant due to DES_CAS intent. In the modified profile it is constant at CAS = 295.6 kt, whereas in the original profile it is at CAS = 341.3 kt. This decrease is due to the decrease in the acceleration during DES_DEC. To maintain a smaller FPA, as it can be seen in Figure 3.10c, it needs to have T $\neq$ T_{idle}, and that is why the acceleration is smaller.

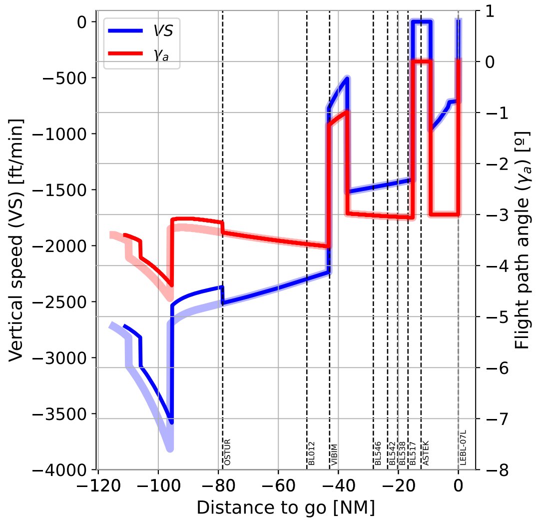
Taking a look at Figure 3.10b we can see how the thrust gets increased from being at T_{idle} to an average of T = 15,435 N, during DES_DEC phase, and to a constant value T = 30,800 N, during DES_CAS.



(a) Altitude and speed profiles



(b) Engine thrust and throttle



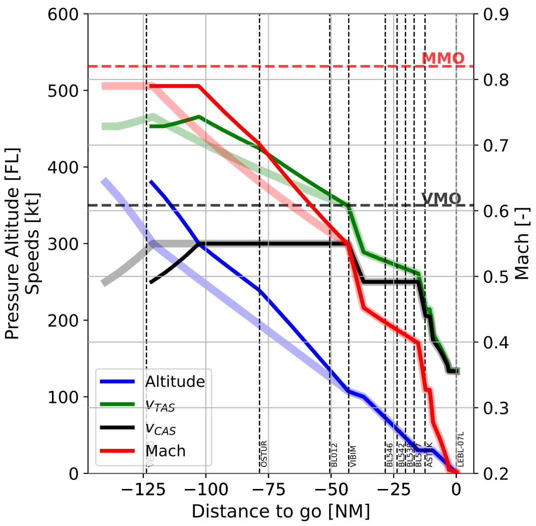
(c) Vertical speed and aerodynamic flight path angle

Figure 3.11: Case B.I Vertical trajectory comparison

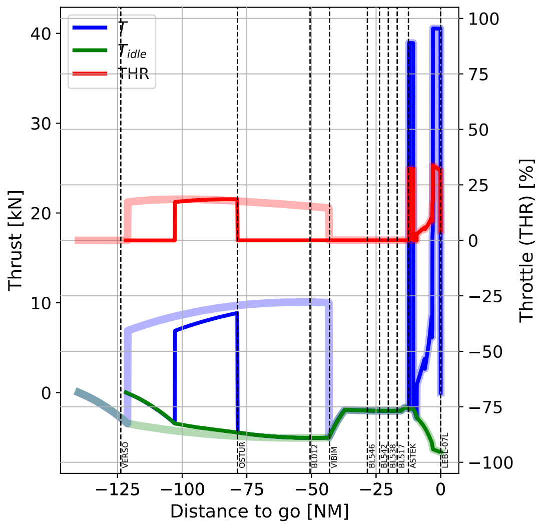
Finally, we can clearly see how due to the IPA intervention to comply with chart's specifications, the new profile reaches the TOD 21 NM later, because having a lower CAS during DES_CAS makes the vertical speed be smaller throughout the rest of the trajectory (see Figure 3.10c).

3.2.3. Case B.I: Minimum altitude with discontinuity

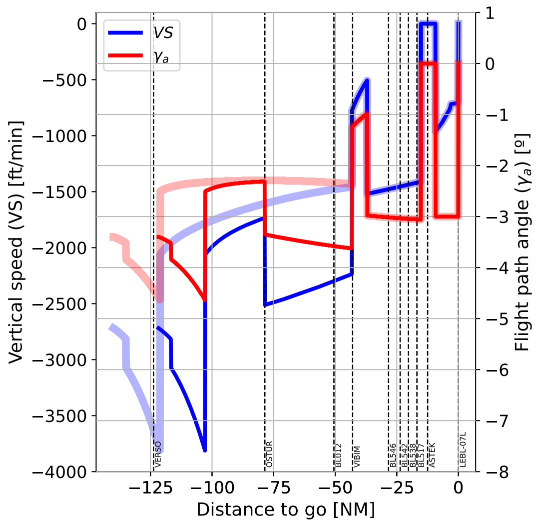
As it can be seen in Figure 3.11a, the new integrated altitude profile is exactly the same as the original one (up until OSTUR). This is due to the fact that in the original profile the aircraft is already flying at $T=T_{idle}$, so the intent override logic leaves the intents as they were. In fact, this result is expected, because the aircraft can not increase its FPA_g if its already flying at minimum thrust, as we can see in Figure 3.11b, where T and T_{IDLE} lines are overlapped.



(a) Altitude and speed profiles



(b) Engine thrust and throttle



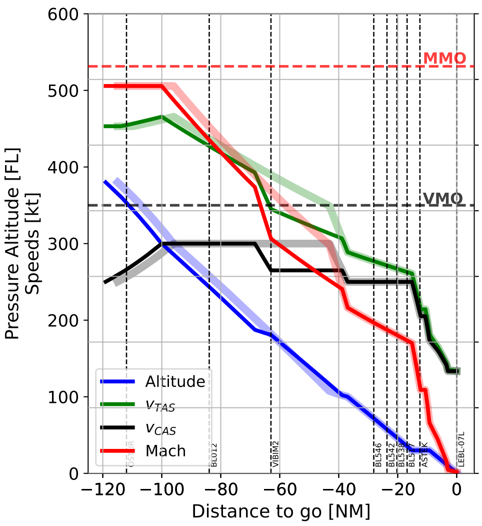
(c) Vertical speed and aerodynamic flight path angle

Figure 3.12: Case B.II Vertical trajectory comparison

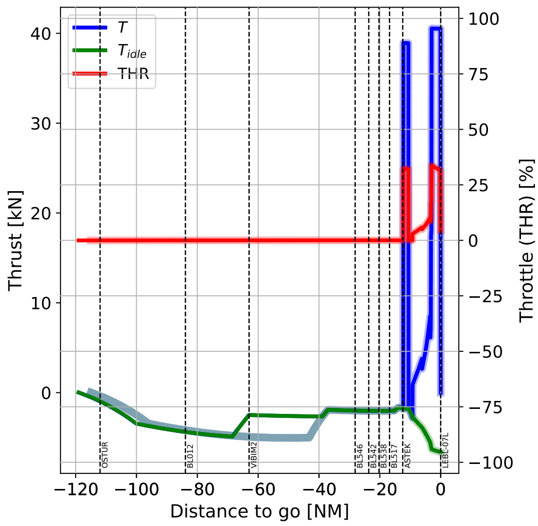
In addition, as the trajectory is not able to be readdressed, as shown in Figure 3.11a, when the aircraft reaches OSTUR the second time, it jumps from its current altitude of 23,800 ft to the 26,000 ft of the constraint. Finally, it continues the integration giving an error message to the TO layer in Dynamo. This error message notifies that the trajectory was unable to be corrected and therefore it exhibits a discontinuity. This logic follows the same behaviour of the FMS (recall Figure 1.5).

3.2.4. Case B.II: Minimum altitude

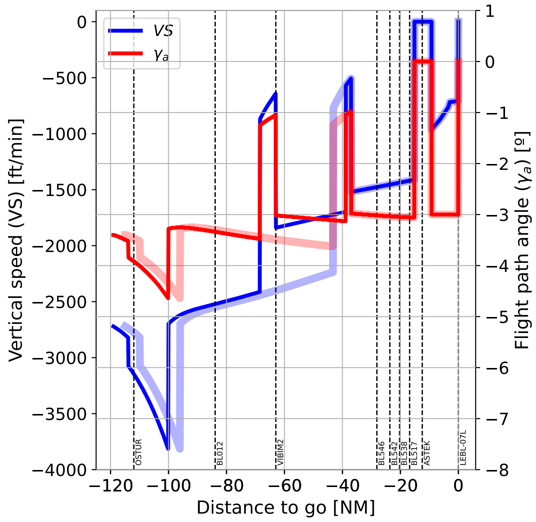
First of all, the setup change to the original unimpeded profile can be seen in Figure 3.12b, where as soon as it starts integrating the new phase, the T (blue) goes down to T_{IDLE} (green), due to the correction that the IPA does. In Figure 3.12a we can see how the trajectory steepens to fulfill h_{min} constraint. At OSTUR, $h_p = 23900ft > h_{min}$. Moreover,



(a) Altitude and speed profiles



(b) Engine thrust and throttle



(c) Vertical speed and aerodynamic flight path angle

Figure 3.13: Comparison of flight parameters throughout profile integration of case C

as the trajectory is steeper, Mach increases (backwards) faster. Hence, the end condition of phase DES_CAS is reached earlier, and consequently, T goes back to T_{IDLE} earlier too.

We can also see in Figure 3.12c, how the FPA_g of that phase changes from -2.3° to -3.5°, as well as the vertical speed from -1500 ft/min to -2400 ft/min, on average, due to the decrease of thrust. Indeed in this scenario, as a difference to the first one, the IPA is able to correct enough the trajectory not to have a discontinuity. Therefore, it does not need to give an error message to the TO layer.

3.2.5. Case C.I: Maximum CAS

Firstly and most important, it is shown in Figure 3.13a how the CAS is kept constant at 265 kt when the aircraft reaches it, and at VIBIM2 it flies again the phase DES_DEC to accelerate until the end condition of CAS=300 kt. During this phase, where the IPA is active, we can see how the increase in Mach is slower because this phase of constant CAS is reached earlier. The increase in Mach is slower at constant CAS than at accelerating CAS, and when it tries to accelerate again after VIBIM2, it is not able to reach the original profile Mach. Therefore, the end condition of phase DES_CAS (see Figure 3.6) is reached later.

Moreover, in Figure 3.13c we can also notice how VS and FPA are interrupted when CAS is kept constant. In fact, in the original profile they also increase when constant CAS is reached, but this new value in this case's profile is lower for both variables (notice negative domain). The FPA goes from an average of -3.5° to -3.1°, and the VS from an average of -2300 ft/min to -1800 ft/min. Hence, during this new phase the aircraft is ascending slower and with a relatively smaller FPA, than what it should be doing in a profile with no waypoint constraints.

3.2.6. Case C.II: Maximum CAS with intermediate end condition

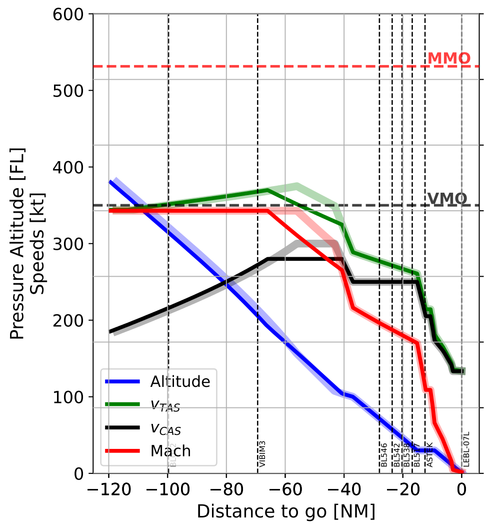
Taking a look at results shown in Figure 3.14a, we can see how CAS is kept constant when it reaches the value of the constraint set at VIBIM3. Then, before arriving at the waypoint, M=0.6 is reached so the IPA adds a new phase in which it is the Mach the variable kept constant. As the Mach reached is already M_{ECON}, after the waypoint, the TP goes back to integrate the original phase, which has the same intents. That is why, after the waypoint the integration parameters follow the same progress. We can also see that the M_{ECON} is reached later, because at constant CAS, Mach increases slower, and constant CAS phase is started earlier.

Moreover, in Figure 3.14c, we can see how VS and FPA are not very much modified. However, we can see that they both suffer a decrease in absolute value during constant CAS IPA phase, and as M_{ECON} is reached later, phase where they decrease in a non-linear manner is also started later and at lower absolute values. However, even though during this pair of new phases altitude is increased more slowly, we can see how the altitude profile is very slightly modified.

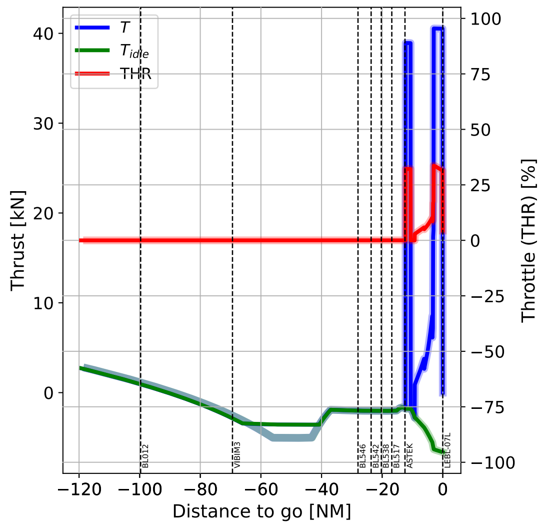
3.2.7. Case D: Waypoint leg

This last case, explained in section 2.4., essentially consists of updating the IPA constraints of each phase when a waypoint is reached. Hence, we can see how the IPA interacts with the profile phases looking at results from the other cases, such as in Figure 3.9 of Case A. In this particular case, after the IPA adds the new phase with a new FPA_g, when the integration arrives at OSTUR, old IPA constraints are erased and new ones from the next leg are added. As phase's end conditions are updated, the aircraft can continue climbing (in backwards) without triggering the h_{max} again right after the waypoint.

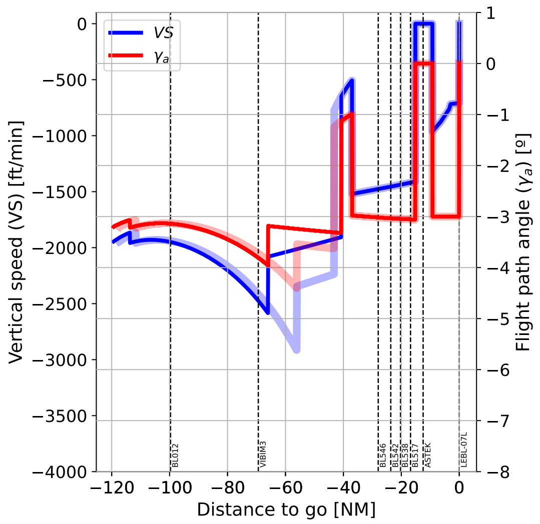
It is worth mentioning that when the IPA reaches a waypoint that is not setting the IPA constraint, it is able to notice it and not erase it for the next leg. In other words, in Figure 3.9a, when the new IPA phase is integrated it passes by BL012, but this is not the waypoint that is setting h_{max} constraint, so it continues, taking the constraint into account, until it reaches OSTUR, which is indeed the waypoint setting the limitation. Therefore, from the



(a) Altitude and speed profiles



(b) Engine thrust and throttle



(c) Vertical speed and aerodynamic flight path angle

Figure 3.14: Case C.II Vertical trajectory comparison

results of the particular cases, it can be verified that the next waypoint logic functions as expected.

3.3. Discussion and conclusion

After testing each type of constraint that might trigger the IPA logic individually, some conclusions can be drawn. Despite each case being an isolated scenario and potentially not being very representative of a real application of the IPA, the obtained results align precisely with the intended behavior of the IPA in each situation as designed in previous chapter.

It is important to mention that this verification process has not been very exhaustive. In other words, these are the most common situations that the IPA might encounter in an operational application scenario. The code does not consider cases that are particularly special or that may force the operational limitations of the implementation. In addition, a potential future verification process would be to apply different meteorology (e.g. applying wind to the scenario), to change the aircraft model (so it can be tested using another APM), or to use other chart procedures. Moreover, future work shall also address simultaneous and/or concurrent IPA triggers.

Nevertheless, the verification of the IPA implementation has been satisfactory for the purposes of this TFG, and so the initial goals of the project have been fulfilled.

CHAPTER 4. VALIDATION WITH A REAL CASE

In the previous chapter, an analysis was conducted on different scenarios in which the three types of constraints considered in the IPA implementation where triggered. In each scenario only one constraint was activated with the following intervention of the IPA solving the route constraint violation. All of them were isolated cases in which the parameters of the profile (end conditions, intents, waypoint location) were specifically designed to trigger the appropriate IPA function and/or to clearly show the IPA proposed solution.

The goal of this chapter is to test the IPA in a realistic case study, in which a real operational chart is used. Therefore, all the parameters of the chart, waypoints location and constraints, are kept as the original published at the AIP (aeronautical information publications). In addition, instead of having only one constraint at a time, here all constraints are active, which implies the IPA may activate and deactivate several times during the vertical trajectory profile integration.

4.1. Scenario

To avoid increasing the complexity of this analysis, the selected charts are the same that have been used in the previous chapter: Barcelona STAR VERSO2E (see Figure 3.1) and approach VIBIM1E (see Figure 3.3). These charts are chosen because Barcelona El Prat airport has a pretty congested airspace, which requires a lot of constraints to keep a certain traffic throughput. Due to having so many constraints, the IPA can be appropriately tested in a scenario that makes the implementation to be triggered multiple consecutive or even simultaneous times. Moreover, these charts have also been chosen because they have constraints of the three types considered in this project.

All the limitations set by the waypoints of the chosen route can be observed in Table 4.1. As it has been explained in Chapter 3, the approach chart corresponds to a tromboning procedure. Therefore, the aircraft must keep flying the downwind leg waypoints until ATC gives a clearance to turn into the base leg. In this validation, the worst-case situation (from an in-flight trajectory planning point of view) is assumed as well, so the aircraft turns to base in the first waypoint of the tromboning (BL538 in Figure 3.3).

The unimpeded profile used in this validation is the same as the one used in Chapter 3. Its phases and end conditions can be seen in Figure 3.5 and Figure 3.6, with the corresponding values in Table 3.1. The only alterations done to this profile for this validation exercise are:

- Phase DEC-DOWNWIND: end condition CAS = 250 kt $\longrightarrow$ CAS = 240 kt.
- Phase DES_BELOW-FL100: end condition h = FL100 $\longrightarrow$ h = FL95.

These subtle changes are done because there are some waypoints that have a constraint set at exactly the same value as the end condition of these phases. For instance, VIBIM has a maximum altitude set at FL100, the same value as the end condition of phase DES_BELOW-FL100. The logic required to handle multiple end condition triggers is out of the scope of this project. A future improvement of the IPA will generate a more robust logic for these kind of situations.

Waypoint	Distance to RWY [NM]	Min. alt	Max. alt	Max. CAS
ASTEK	12.4			
BL517	16.75			
BL538	20.2	4000 ft	FL80	
BL542	23.55	4000 ft	FL80	
BL546	28.31			220 kt
VIBIM	42.91	FL70	FL100	
BL012	50.43	FL100		250 kt
OSTUR	78.46		FL200	
VERSO	123.54	FL150	FL250	

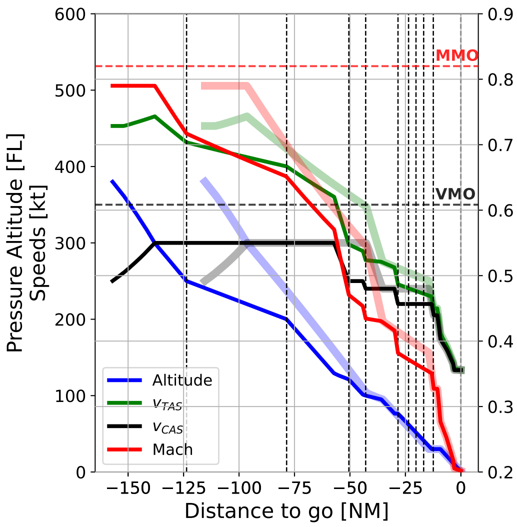
Table 4.1: Arrival and approach waypoints with their respective constraints

Phase ID	Pair of intents	Triggered end condition
CONF1-DEC	VS = 0 ft/min $T = T_{IDLE}$	$CAS = V_{GD} = 205.15$ kt
CONF1-GD	ACC_CAS_time = 0 kt/s VS = 0 ft/min	s = -12.23 NM
DEC-DOWNWIND	VS = 0 ft/min $T = T_{IDLE}$	$CAS = V_{DEC} = 240$ kt
DES_BELOW-FL100	ACC_CAS_time = 0 kt/s $T = T_{IDLE}$	h = FL95
DES_DEC	ESF = 0.3 $T = T_{IDLE}$	$CAS = CAS_{ECON} = 300$ kt
DES_CAS	ACC_CAS_time = 0 kt/s $T = T_{IDLE}$	$M = M_{ECON} = 0.79$
DES_MACH	ACC_Mach_time = 0 s^{-1} $T = T_{IDLE}$	$h = h_{CRU}$ = FL380

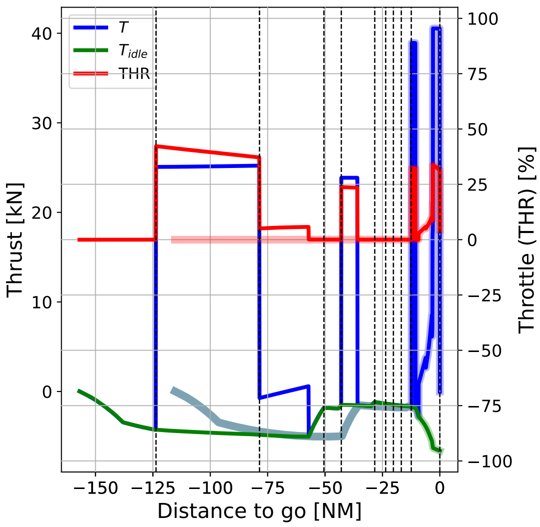
Table 4.2: Unimpeded profile phase sequence (IPA inactive)

In addition, Table 4.2 shows the sequence of phases integrated by Dynamo when IPA is not active, with their respective pair of intents. It also shows the end condition which has stopped the integration of the corresponding phase. It is, therefore, the result of the integration of the unimpeded profile. The phases corresponding to the final approach are not shown in this table, because they do not constitute an interesting part of the profile integration.

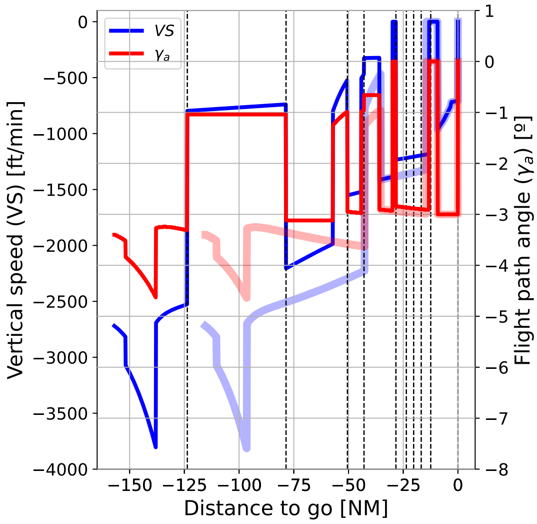
Finally, the aircraft model used in this validation case study is the same as it has been used in the rest of the project: an aircraft performance model (APM) from BADA4 of the Airbus A320-321 (see section 3.1.2.).



(a) Altitude and speed profiles



(b) Engine thrust and throttle



(c) Vertical speed and aerodynamic flight path angle

Figure 4.1: Real case validation vertical trajectory comparison (original profile in transparent lines and the IPA solution in solid lines)

4.2. Results

The results of this verification are particularly interesting, because the IPA is activated in four different moments during the integration of the profile. Not only is it activated so many times, but also there is one occasion in which the IPA is activated while an IPA phase is still being integrated, so when the IPA mode is still active. This case adds a level of complexity that is out of the scope of this project. Yet, since the IPA operates properly in this particular scenario with no further implementations, the decision made was to keep it and analyse it, as it is a very interesting function of the current implementation.

Figure 4.1 shows the general results after the complete profile integration, if compared with the integration of the same profile when IPA module is deactivated (no waypoint constraints taken into account and the unimpeded profile is flown). We can clearly identify the differences in these two profiles. As there are many details to be analysed, a more in-depth look in each IPA activation is provided in the following sections.

It is important to note that all comments in the following sections describe the trajectory evolution in backwards (i.e. from the runway threshold to the cruise altitude), since this is how the trajectory is being integrated numerically.

4.2.1. First IPA activation: CAS_{max} at DES_BELOW-FL100

The first IPA intervention in the integration is due to the triggering of the CAS_{max} = 220 kt at BL546 (see Figure 3.3). During DEC-DOWNWIND phase, the aircraft accelerates (in backwards) until the end condition CAS = 240 kt. However, at CAS = 220 kt, the IPA triggers and the CAS_{max} solution method, explained in section 2.3.3., is carried out.

As it can be seen in Figure 4.2a, CAS is kept constant when it reaches the CAS limitation (shortly after ASTEK) until BL546 is reached. In fact, it can be seen correct functioning of the constraint propagation, because there are three waypoints between ASTEK and BL546. Since no other waypoint sets a CAS_{max} constraint, this one is propagated from BL546 to the runway. Therefore, when the IPA is active and it reaches the next waypoint, it identifies if this waypoint is relevant (setting the constraint triggered) or not. Then, when the integration reaches BL546, the phase added by the IPA finishes and the normal integration resumes, accelerating again to reach the end condition of phase DEC-DOWNWIND (CAS = 240 kt) and integrating the remaining profile.

Moreover, taking a look at Figure 4.2c, we can see how it starts to ascend (in backwards) a little bit earlier, when the constraint is triggered. This happens because the IPA jumps to the next phase, DES_BELOW-FL100, which makes the aircraft start ascending with the intent commands: CAS = CAS_{max} and T = T_{idle}.

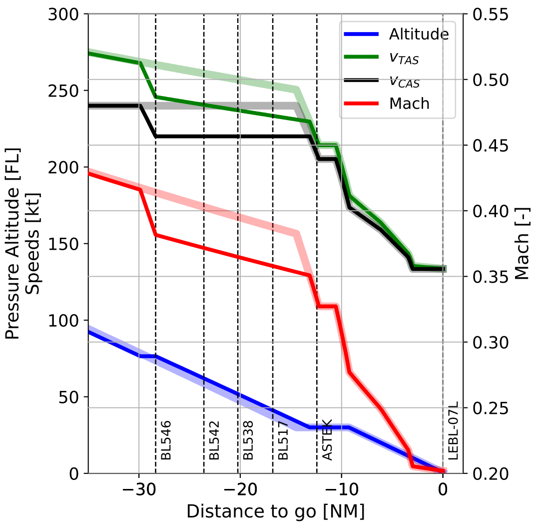
In terms of thrust usage, this IPA activation does not provide significant changes if compared with the unimpeded profile (see Fig. 4.2b).

4.2.2. Second IPA double activation: CAS_{max} and h_{max} at DES_DEC

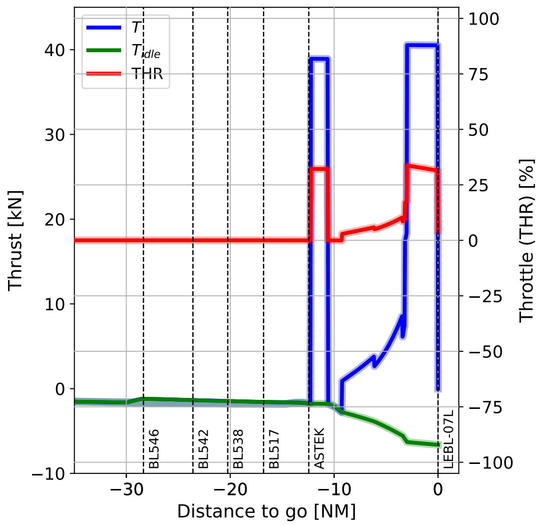
This second IPA activation is very interesting, because two constraints trigger at the same time. Firstly, when the TP is integrating the phase DES_DEC, it arrives at the CAS_{max} = 250 kt constraint of waypoint BL012. Consequently, the IPA activates and starts integrating the next phase, DES_CAS, at constant CAS, like in the IPA activation previously described.

Then, while integrating this new phase added by the IPA, it reaches the maximum altitude constraint h_{max} = FL100 set at VIBIM. Therefore, as it is explained in section 2.3.1., the current phase is reset and the aircraft intents are overridden. In this special case, where a CAS_{max} and afterwards an h_{max} constraints trigger, the intent commands of this new phase are: FPA_g and constant CAS. This second new phase starts right after phase DES_BELOW-FL100, because DES_CAS_IPA and DES_DEC are erased.

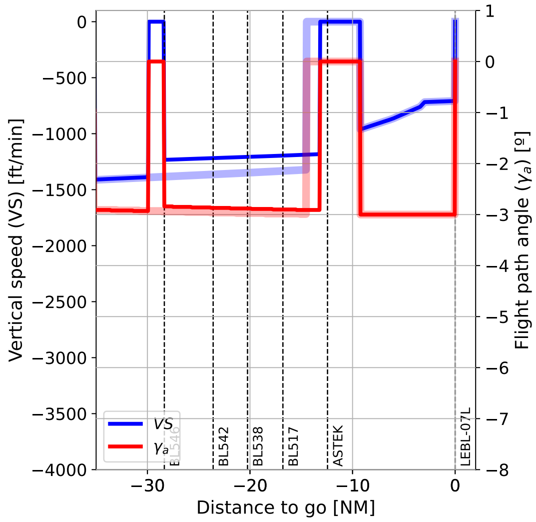
Figure 4.3a shows how the IPA starts integrating this second new phase at constant CAS = 240 kt, instead of the value of the limitation, because it is the aircraft speed at the end of the previous phase. Therefore, when it reaches VIBIM it accelerates again (DES_DEC) in spite of not having reached BL012 yet.



(a) Altitude and speed profiles



(b) Engine thrust and throttle



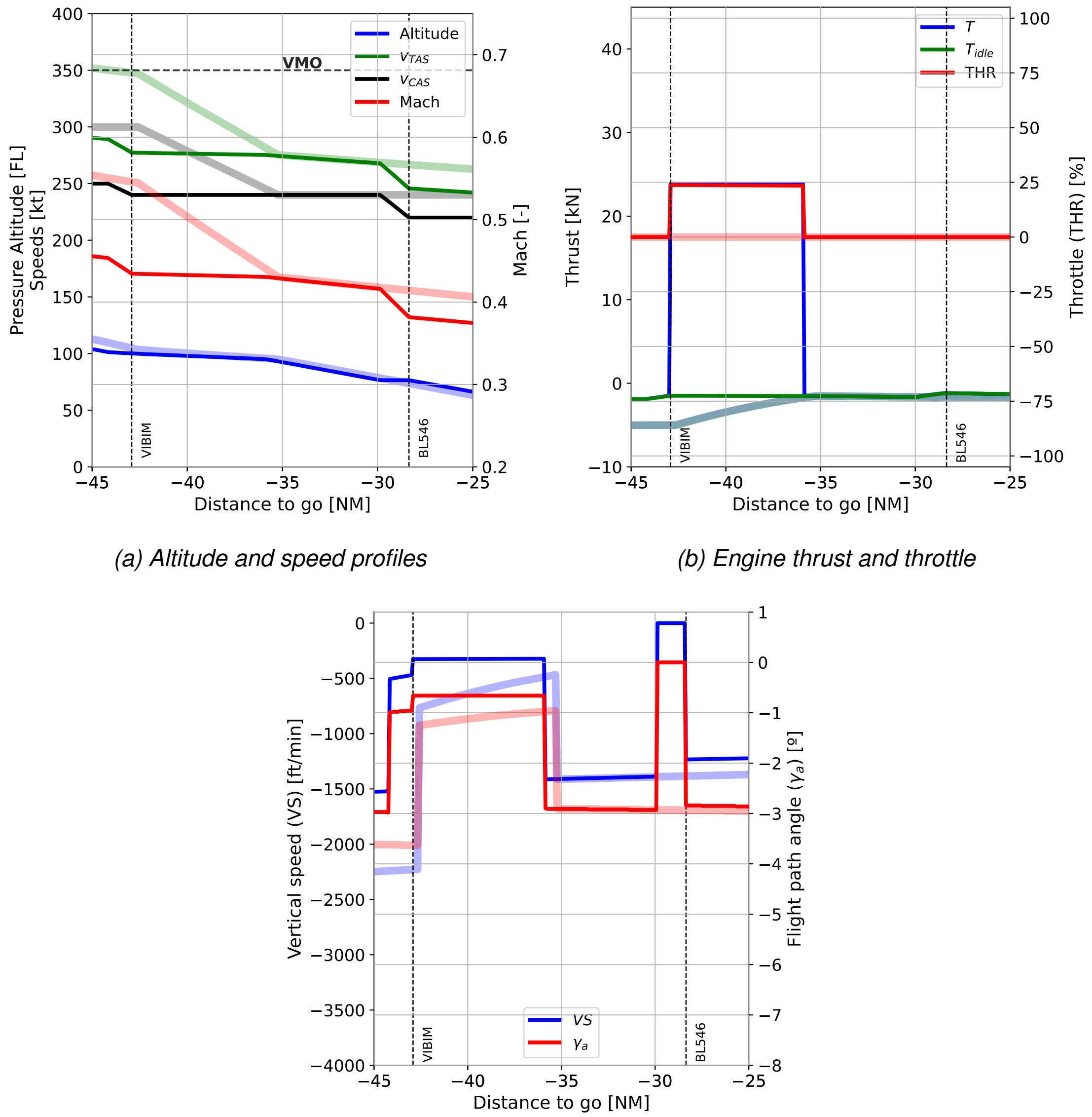
(c) Vertical speed and aerodynamic flight path angle

Figure 4.2: First IPA activation part vertical trajectory comparison

As it can be seen in Figure 4.3c, FPA_g[1] is computed to comply with the limitation at VIBIM. Although the variation is not very significant, it changes from an average of -1.1° to a constant value of -0.66°. At the same time, vertical speed changes from an average of -618 ft/min to a value of -324 ft/min.

Finally, taking a look at Figure 4.3b, it can be seen how in order to maintain the constant speed during this new phase integration, the aircraft needs a thrust of T = 23,864 N and it is not longer flying at T_{idle} as in the unimpeded profile.

[1] recall that in this validation exercise no winds are considered and therefore, the aerodynamic flight path angle (FPA_a or γ_a in the figures) equals the ground flight path angle FPA_g.



(a) Altitude and speed profiles

(b) Engine thrust and throttle

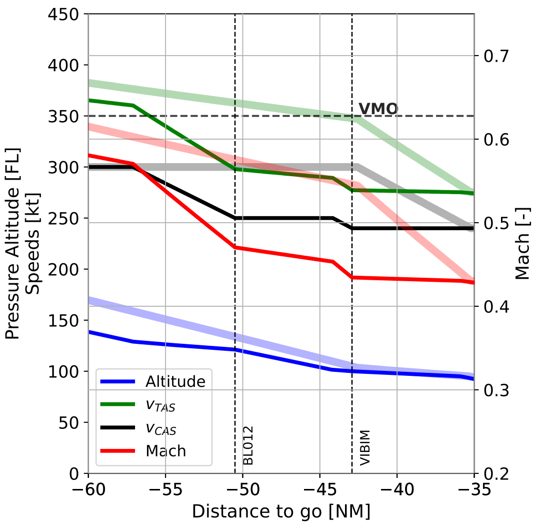
(c) Vertical speed and aerodynamic flight path angle

Figure 4.3: Second IPA double activation part vertical trajectory comparison

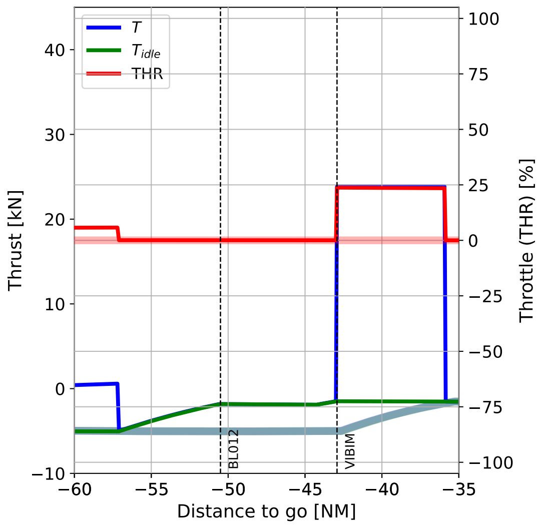
4.2.3. Third IPA activation: CAS_{max} at DES_DEC

In this third time that the IPA activates it, is due to the CAS_{max} = 250 kt constraint set at BL012. In the last section it was explained that when the integration of the new IPA phase is finished, the logic tries to accelerate again the aircraft (in backwards) to the value set into the unimpeded profile. Hence, it is in this second DES_DEC phase when the CAS reaches the limit value. In fact, this is the second time that it reaches this CAS, because in the previous IPA activation the first of the two constraints that trigger is the same as here, but since h_{max} triggers afterwards, the phase is reset.

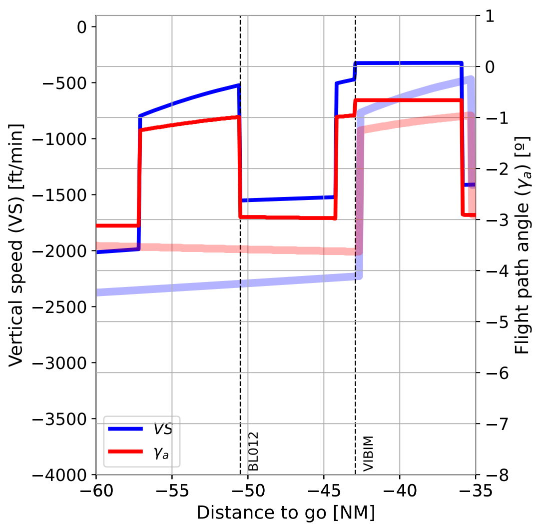
This case is not particularly interesting, because the result is the same as in the first activation. As it can be seen in Figure 4.4a, the aircraft tries to accelerate until it reaches CAS = 250 kt, whereas in the original profile CAS = 300 kt. This is the reason why VS and FPA_g are lower when applying IPA constraints, -1540 ft/min and -2.96° respectively, as it



(a) Altitude and speed profiles



(b) Engine thrust and throttle



(c) Vertical speed and aerodynamic flight path angle

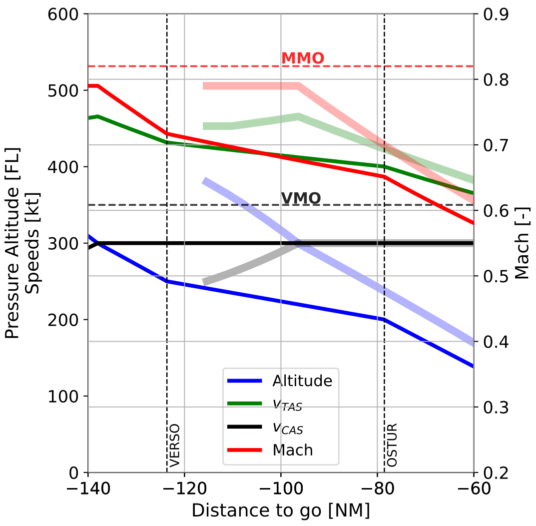
Figure 4.4: Third IPA activation part vertical trajectory comparison

can be seen in Figure 4.4c

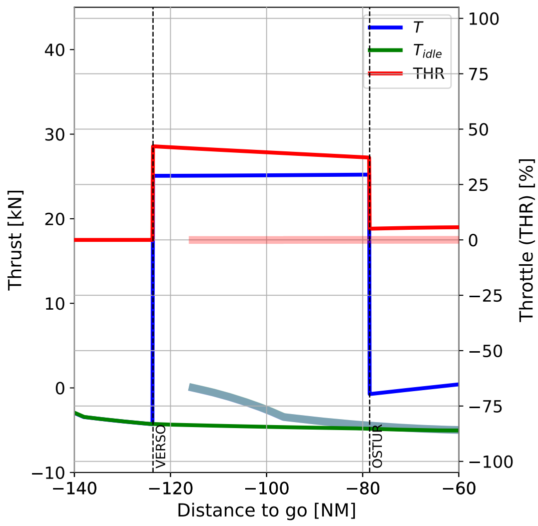
Lastly, we can see that when it reaches BL012 and the constraint is not active anymore, the aircraft carries out DES_{DEC} phase accelerating again to reach end condition CAS = 300 kt. Moreover, we can start to see the difference in both profiles when it comes to the distance needed to reach the same altitude.

4.2.4. Fourth IPA activation: h_{max} at DES_CAS

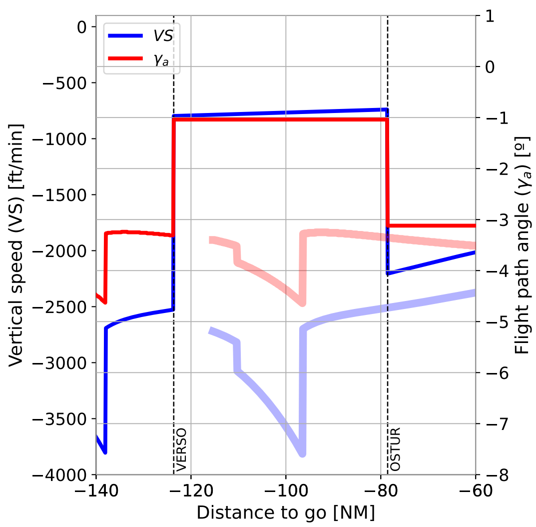
Last but not least, the IPA activates a fourth time. In fact, this case is particularly interesting because it is a double sequential activation. Firstly it is h_{max} = FL200 set by OSTUR that activates. This constraint triggers during the integration of phase DES_{CAS} and, as explained in section 2.3.1., the IPA resets the integration of this phase forcing an intent to



(a) Altitude and speed profiles



(b) Engine thrust and throttle



(c) Vertical speed and aerodynamic flight path angle

Figure 4.5: Fourth IPA activation part vertical trajectory comparison

have a specific FPA_g to comply with the limitation when it flies by OSTUR. Yet, when the IPA is deactivated after passing by this waypoint, in the following integration of DES_{CAS}, constraint h_{max} = FL250 set by VERSO triggers, and the same solution method is applied.

As it can be seen in Figure 4.5a, the vertical profile needs to be modified to reach OSTUR, and then VERSO, at a lower altitude. In fact, the original profile with no IPA constraints, never reaches VERSO because the aircraft arrives (in backwards) to the cruise altitude of 38,000 ft before. Therefore, the difference between both profiles gets bigger in this part of the descent.

In Figure 4.5c we can see how the FPA_g differs from the first part of the activation (beginning of the phase to OSTUR) to the second (OSTUR to VERSO), -3.12° and -1.04° respectively. This is due to the fact that the first time that the IPA activates it computes a FPA_g with the constraint and location of the first waypoint, OSTUR, and the second time with the second waypoint, VERSO. A similar behaviour happens with the VS, the first part has an average value of -2087 ft/min, whereas the second part has an average value of -768 ft/min. The second part has a lower VS and FPA_g because the distance between the waypoint and the initial point is bigger and the altitude difference between the constraint and the initial altitude is smaller.

Finally, it is worth mentioning that there is big decrease in both VS and FPA_g in comparison with the profile with no IPA constraints. In fact, this difference is much bigger in the second part of this fourth IPA activation.

4.3. Discussion and conclusion

To conclude this chapter, in this real case validation it has been shown how the IPA is constantly interacting with the TP while it is doing the integration. The fact that it activates four times in this particular example, points out the importance of taking into account the constraints set by the ANSP in the operational charts, as well as the importance of adding the IPA layer into the structure of Dynamo.

In the end, the difference between the optimal descent profile (i.e. a continuous descent with engines at idle thrust, as specified in the unimpeded profile) and the profile taking into account operational limitations in Barcelona El Prat TMA airspace, is considerable. Therefore, the fact of having a high volume of traffic in the airspace, and the need to maintain a certain capacity, leads to a loss of efficiency in the descent procedures.

Table 4.3 shows the sequence of phases after the IPA interacts with the profile, adding its modified phases (highlighted). Each phase has its pair of intent commands as well as the end condition that stopped their integration. This table can be compared with Table 4.2, where five phases are added to the unimpeded profile phase sequence. Therefore, the impact of the IPA in the profile integration is clearly noticeable.

Phase ID	Pair of intents	Triggered end condition	IPA constr.
CONF1-DEC	VS = 0 ft/min $T = T_{IDLE}$	$CAS = V_{GD} = 205.15$ kt	
CONF1-GD	ACC_CAS_time = 0 kt/s VS = 0 ft/min	s = -12.23 NM	
DEC-DOWNWIND	VS = 0 ft/min $T = T_{IDLE}$	$CAS = CAS_{max} = 220$ kt	
DES_BELOW-FL100_IPA	ACC_CAS_time = 0 kt/s $T = T_{IDLE}$	$s_{BL546} = -28.35$ NM	CAS_{max}
DEC-DOWNWIND	VS = 0 ft/min $T = T_{IDLE}$	$CAS = V_{DEC} = 240$ kt	
DES_BELOW-FL100	ACC_CAS_time = 0 kt/s $T = T_{IDLE}$	h = FL95	
DES_CAS_IPA	ACC_CAS_time = 0 kt/s $FPA_g = -0.66^\circ$	$s_{VIBIM} = -42.92$ NM	h_{max}
DES_DEC	ESF = 0.3 $T = T_{IDLE}$	$CAS = CAS_{max} = 250$ kt	
DES_CAS_IPA	ACC_CAS_time = 0 kt/s $T = T_{IDLE}$	$s_{BL012} = -50.5$ NM	CAS_{max}
DES_DEC	ESF = 0.3 $T = T_{IDLE}$	$CAS = CAS_{ECON} = 300$ kt	
DES_CAS_IPA	ACC_CAS_time = 0 kt/s $FPA_g = -3.12^\circ$	$s_{OSTUR} = -78.55$ NM	h_{max}
DES_CAS_IPA	ACC_CAS_time = 0 kt/s $FPA_g = -1.04^\circ$	$s_{VERSO} = -123.64$ NM	h_{max}
DES_CAS	ACC_CAS_time = 0 kt/s $T = T_{IDLE}$	$M = M_{ECON} = 0.79$	
DES_MACH	ACC_Mach_time = 0 s^{-1} $T = T_{IDLE}$	$h = h_{CRU} = FL380$	

Table 4.3: Final profile phase sequence

CONCLUSIONS

This project has focused on the conception, development and validation of a new feature within Dynamo aiming at increasing its Technology readiness level (TRL). The new feature, called IPA (intent procedure adapter), incorporates into the trajectory computation process the constraints established by the ANSP in the operational navigation charts. The goals established for the project have been successfully achieved and the IPA has been implemented for descent operations.

The primary aim of this project was to develop a new feature capable of enhancing the maturity level of the Dynamo software, making it more realistic and similar to an FMS. The resultant implementation can serve as a tool for prediction and optimization of vertical profiles within busy TMAs, faithfully replicating real-world conditions. Consequently, a potential future utilization of the IPA entails facilitating efficiency vs. capacity assessments in TMAs, empowering users to manipulate operational constraints as desired. For instance, it is possible to analyze the impact on fuel consumption of utilizing an IPA solution versus a continuous descent operation (CDO). Therefore, the IPA presents itself as an enabler to assess ATM performance within a TMA.

The program now possesses the capability to receive user input, encompassing aircraft configuration, vertical profile, and lateral route, and subsequently provide a viable vertical profile that properly considers all relevant constraints. Additionally, diverse application cases have been identified and they have served to validate the implementation.

The existing implementation of this project contains numerous essential functionalities that cover the most common scenarios. Nevertheless, there remains scope for improvement within the implementation. It is imperative to enhance its robustness to include a broader spectrum of situations and expand its range of functionalities. As an illustration, the present implementation exclusively focuses on descent operations, whereas it could be extended to encompass both climb and cruise operations.

Moreover, concerning the verification and real-case validation sections, the assumption is made that the IPA activates only once, a part from the second activation in the validation process. An enhanced implementation could enable the IPA to activate simultaneously several number of times, although this would introduce a level of complexity that may solely prove necessary in highly specific circumstances.

Furthermore, future work should investigate the influence of wind on the IPA profile solution and/or non-standard atmospheres. Presently, the validation scenarios assume ISA (international standard atmosphere) conditions and the absence of wind; yet in practical operational scenarios, non-standard atmospheres and wind constitute significant factors that must be taken into account.

Finally, from a personal standpoint, this project has presented a valuable opportunity to undertake the design and development of a software implementation within Icarus research group. It has afforded an opportunity to acquire proficiency in collaborative work, proficient communication, progress reporting, and decision-making. The process of designing and constructing the implementation has underscored the significance to consider all factors of the creation of a successful product, intended for utilization by diverse stakeholders. This project has yielded considerable personal and professional satisfaction, while the acquired expertise has successfully prepared for future endeavors in the industry.

BIBLIOGRAPHY

[1] Raúl Sáez García. *Traffic synchronization with controlled time of arrival for cost-efficient trajectories in high-density terminal airspace*. PhD thesis, UPC, Departament de Física, Oct 2021. 1

[2] Javier Lopez Leones. *Definition of an aircraft intent description language for air traffic management applications*. PhD thesis, University of Glasgow, 2008. 1

[3] J. Lopez-Leones, A. M. Vilaplana, E. Gallo, F. A. Navarro, and C. Qerejeta. The Aircraft Intent Description Language: A key enabler for air-ground synchronization in Trajectory-Based Operations. In *Proc. of the IEEE/AIAA Digital Avionics Systems Conference*, 2007. 1

[4] R. Dalmau, M. Melgosa, S. Vilardaga, and X. Prats. A fast and flexible aircraft trajectory predictor and optimiser for atm research applications. In *ICRAT 2018 - 8th International Conference for Research in Air Transportation, Castelldefels, Espanya, 26-29 Juny 2018*, 2018. 1, 7, 8

[5] Icarus research group. Dynamo wiki. Universitat Politecnica de Catalunya. [Accessed: February 2023]. 1, 10

[6] ICAO. *Procedures for air navigation services: Aircraft Operations. Doc 8168, Sixth edition*. ICAO, 2018. 3

[7] ENAIRE. Carta de llegada normalizada vuelo por instrumentos (STAR)-OACI RNAV1, MADRID-Barajas. AIRAC 2301, 2023. 3, 4

[8] AIRBUS. Flight crew operating manual, a318/a319/a320/a321, 2005. 5, 6

[9] H. Khaledian, R. Saez, J. Vilà-Valls, and X. Prats. *On the impact of guidance commands mismatch in IMM-Based guidance modes identification for aircraft trajectory prediction*, page 1–6. 2022. 8, 9

[10] X. Prats and R. Dalmau. FASTOP Mathematical Model. Technical report, 2014. Deliverable D2.2, Ed. 1.3. 9

[11] ENAIRE. Carta de llegada normalizada vuelo por instrumentos (STAR)-OACI, BARCELONA-El Prat. RWY 06L/06R, 2023. 14, 26

[12] ENAIRE. Carta de transición a la aproximación vuelo por instrumentos RNAV1, BARCELONA-El Prat. RWY 06L, 2023. 27

[13] A. Nuic and V. Mouillet. *User Manual for the Base of Aircraft Data (BADA) Family 4*. EUROCONTROL, 2016. 26