



HAL
open science

Space trajectory optimization. Last stage of a launcher – Space debris cleaning.

Max Cerf

► To cite this version:

Max Cerf. Space trajectory optimization. Last stage of a launcher – Space debris cleaning.. Optimization and Control [math.OA]. Université Pierre et Marie Curie - Paris VI, 2012. English. NNT : . tel-00736748

HAL Id: tel-00736748

<https://theses.hal.science/tel-00736748v1>

Submitted on 29 Sep 2012

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

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

**THESE DE DOCTORAT DE
L'UNIVERSITE PIERRE ET MARIE CURIE**

Spécialité

Mathématiques
(Ecole doctorale : P6 Sciences Mathématiques de Paris Centre
UFR 929 Mathématiques)

Présentée par

M. Max CERF

Pour obtenir le grade de

DOCTEUR de l'UNIVERSITÉ PIERRE ET MARIE CURIE

Sujet de la thèse :

Optimisation de trajectoires spatiales
Vol d'un dernier étage de lanceur – Nettoyage des débris spatiaux

soutenue le 28 septembre 2012

devant le jury composé de : (préciser la qualité de chacun des membres).

M. Emmanuel TRÉLAT, Thomas HABERKORN
M. Yacine CHITOUR, Moritz DIEHL
M. Jean-Baptiste CAILLAU, Jean-Michel CORON,
Joseph GERGAUD, Yvon MADAY

Directeurs de thèse
Rapporteurs

Examineurs

Résumé

Ce travail porte sur deux problèmes d'optimisation de trajectoires spatiales: le vol d'un dernier étage de lanceur, et le nettoyage des débris spatiaux. L'objectif est de développer pour ces deux problèmes des méthodes de résolution et des logiciels utilisables dans un contexte industriel. Les travaux comportent une partie théorique de formulation et une partie appliquée de résolution numérique. Les domaines abordés sont la mécanique spatiale, l'optimisation discrète, l'optimisation continue en dimension finie et le contrôle optimal.

Vol d'un dernier étage de lanceur

Le problème de minimiser la masse de propergol consommée par un lanceur pour rejoindre l'orbite visée est d'une importance cruciale. Cette masse de propergol conditionne en effet la masse totale du véhicule et donc son coût. Vu son importance pratique, ce problème a fait l'objet de nombreux développements depuis les années 1960. Il peut être aujourd'hui considéré comme bien compris et différentes méthodes de résolution numérique sont disponibles, avec leurs points forts et leurs points faibles. Ces méthodes sont généralement classées en deux catégories : méthodes directes et méthodes indirectes.

- Les méthodes directes consistent à discrétiser le problème de commande optimale pour le formuler comme un problème d'optimisation paramétrique. Le problème résultant est de dimension finie et il peut être résolu par un algorithme d'optimisation non linéaire sous contraintes. Cette approche est relativement simple à mettre en œuvre et robuste, car l'utilisateur peut directement fournir une loi de commande initiale et la modifier en fonction du comportement de l'algorithme. Les deux principaux défauts de ces méthodes sont le temps de calcul (lié à la dimension du problème discrétisé) et la précision de convergence. La qualité de l'optimum obtenu est en effet affectée par la discrétisation du problème de commande optimale, et par la précision numérique du solveur.
- Les méthodes indirectes sont basées sur le Principe du Maximum de Pontryagin qui exprime des conditions nécessaires d'optimalité. Le problème est formulé comme un problème aux deux bouts : les inconnues sont les composantes du vecteur adjoint initial qui doivent satisfaire un système d'équations non linéaires à l'instant final. La résolution s'appuie sur une méthode de tir, associée à un solveur de type Newton. L'optimum est généralement obtenu de façon très rapide et précise, la principale difficulté étant de fournir une initialisation suffisamment proche de l'optimum afin de permettre la convergence.

Dans un contexte industriel, l'accent est mis sur la capacité à résoudre rapidement et si possible automatiquement ce problème. Le temps nécessaire pour un ingénieur à l'optimisation d'une trajectoire de lancement impacte directement :

- le coût de préparation d'une mission réelle (comme pour le lanceur Ariane5),
- le nombre de configurations (étagement, propergols) qui pourra être examiné lors de la conception d'un lanceur futur.

Même si l'expérience acquise permet de calculer des trajectoires optimales de lanceur de façon relativement rapide, l'ingénieur reste confronté de façon récurrente aux difficultés usuelles des méthodes d'optimisation numériques : robustesse de l'initialisation, vitesse de convergence, optima locaux. Le savoir-faire pratique reste indispensable en l'absence de méthode automatique suffisamment robuste pour couvrir l'ensemble des missions et des véhicules envisagés.

Le travail proposé consiste à élaborer une méthode de résolution automatique pour le vol du dernier étage d'un lanceur. Les principales difficultés d'optimisation sont en effet liées à cette phase de vol, soumise à des contraintes d'état et des contraintes finales.

La méthode envisagée repose sur des algorithmes de continuation dont le principe est de partir d'un problème simple dont la solution est connue pour aller progressivement vers le problème réel, en initialisant chaque problème intermédiaire avec la solution du problème précédent. Les méthodes de continuation appliquées aux problèmes de trajectoires spatiales ont atteint aujourd'hui une maturité suffisante. Mais leur mise en œuvre n'est pas automatique et demande une expérience pratique importante, avec des réglages au cas par cas, au cours des phases de continuation successives. Elles ne sont en l'état pas suffisamment efficaces pour être employées de façon systématique dans un contexte industriel.

Pour la résolution du vol d'un dernier étage de lanceur, deux approches différentes par continuation sont envisagées à partir de simplifications des modèles :

- La première approche consiste à résoudre le problème en Terre plate, avec un champ de gravité uniforme, puis à revenir au problème en Terre ronde par une double continuation sur la gravité et sur la courbure de la Terre.
- La deuxième approche consiste à résoudre le problème impulsional, correspondant à un niveau de poussée infini, puis à revenir au problème en poussée continue par une continuation sur le niveau de poussée.

L'objectif est de définir un processus de résolution le plus automatique possible, applicable pour différents jeux de données du problème réel : caractéristiques du lanceur, orbite à atteindre.

Mots-clés :

Transfert orbital, Commande optimale, Principe du Maximum de Pontryagin, Méthode de tir, Continuation

Nettoyage des débris spatiaux

La prolifération non contrôlée des débris orbitaux depuis les débuts de la conquête spatiale rend problématique l'accès à l'espace dans les années à venir. Les différentes études sur la population des débris spatiaux et son évolution montrent qu'il serait nécessaire de retirer chaque année 5 gros débris (étages de lanceurs, satellites) en orbite basse pour stabiliser la situation.

Des missions de nettoyage sont donc aujourd'hui envisagées, consistant à envoyer un véhicule pour capturer successivement plusieurs gros débris et les désorbiter en les plaçant sur des orbites de retombée.

Pour minimiser le coût de la mission de nettoyage, mesuré par la consommation d'ergols nécessaire, il faut sélectionner parmi une liste de candidats les 5 débris à désorbiter, choisir l'ordre dans lequel ceux-ci seront visités, et optimiser les manoeuvres du véhicule pour se rendre d'un débris à l'autre. La mission doit être réalisée en moins d'un an.

L'ensemble des débris candidats forme un graphe orienté complet. Les valuations des arcs sont les coûts de transfert pour aller d'un débris à un autre, chaque débris étant défini par son orbite osculatrice à l'instant de début de la mission. Ces orbites évoluent pendant la durée de la mission sous l'effet en particulier des perturbations du champ gravitationnel terrestre (aplatissement).

Chaque évaluation d'un coût de transfert nécessite de résoudre le problème de contrôle optimal associé qui consiste, en partant du débris initial, à réaliser un rendez-vous avec le débris final. Ce problème intrinsèquement difficile lorsque les orbites sont quelconques, est de plus dépendant de l'instant initial, en raison des perturbations qui modifient les orbites des débris.

La sélection des débris et de l'ordre de visite est un problème de type plus court chemin dans un graphe à coûts dépendant du temps. Des algorithmes efficaces existent pour ce type de problème lorsque les coûts des arcs sont constants (« k-shortest path »), mais ils ne peuvent pas s'appliquer directement au problème de la mission de nettoyage.

Le problème global comporte donc un fort couplage entre des aspects d'optimisation combinatoire et des aspects de contrôle optimal, chacun de ces problèmes étant intrinsèquement difficile. La résolution de ce problème dans toute sa généralité est inenvisageable avec les moyens actuels, et des hypothèses simplificatrices sont nécessaires.

Le travail proposé consiste à élaborer une méthode de résolution avec des objectifs d'efficacité, de représentativité et d'optimalité. En vue du développement d'un futur véhicule de nettoyage, on souhaite en effet être capable de résoudre différentes instances du problème en des temps raisonnables, et que les solutions obtenues présentent des garanties d'optimalité et de réalisme.

Les principales hypothèses permettant de simplifier la formulation du problème portent sur les orbites des débris, le moteur du véhicule et la durée de la mission. Une grande partie des satellites à désorbiter se trouvent sur des orbites héliosynchrones. On oriente donc le dimensionnement du véhicule prioritairement sur ce type de débris et les orbites

peuvent ainsi être supposées basses et quasi circulaires. Par ailleurs, l'utilisation d'un moteur à poussée forte permet une approximation impulsienne des manoeuvres de transfert d'orbite. Enfin, la contrainte de durée totale de 1 an pour la mission permet des phases d'attente entre transferts successifs, de façon à effectuer les manoeuvres aux instants les plus favorables. On pourra ainsi définir une stratégie-type applicable à chaque transfert. Cette stratégie type permet de ramener le problème de contrôle optimal à un problème d'optimisation continue en dimension finie, dont la résolution peut être couplée au problème de chemin. Une linéarisation locale permet ensuite de mettre en œuvre des algorithmes de séparation-évaluation. En itérant le processus de linéarisation – résolution à partir d'une solution initiale, on pourra converger vers la solution du problème global.

Mots-clés :

Débris spatiaux, Mécanique orbitale, Séparation et Evaluation, Programmation linéaire

Articles écrits au cours de la thèse

[1] M. Cerf, T. Haberkorn, E. Trélat

Continuation from a flat to a round Earth model in the coplanar orbit transfer problem

Optimal Control Application and Methods, 36 pages, DOI: 10.1002/oca

Published online in Wiley InterScience (03/2011)

[2] M. Cerf

Multiple space debris collecting mission, debris selection and trajectory optimization

Journal of Optimization Theory and Applications, 33 pages, DOI: 10.1007/s10957-012-0130-6

Published online in Springer InterScience (08/2012)

[3] M. Cerf, T. Haberkorn, E. Trélat

Initialization of the high-thrust minimal consumption orbit transfer problem by the impulsive solution.

In progress.

Abstract

This work addresses two space trajectory optimization problems : the flight of a launcher upper stage and the collecting of space debris. For these two problems the goal is to develop solution methods and softwares usable in an industrial framework. The work consists of a theoretical part to formulate the problems and an applied part to develop numerical algorithms. The related fields are space mechanics, integer programming, nonlinear programming and optimal control.

Launcher upper stage flight

The problem of minimizing the fuel consumed by a launcher to reach the targeted orbit is of major importance. Indeed the propellant mass influences directly the launcher gross mass and therefore the launch cost. Due to its practical importance, this problem has attracted numerous theoretical and algorithmic developments since the 1960ies. It can be considered as well-mastered nowadays and various solution methods are available, with their respective advantages and disadvantages. These methods are generally divided into two categories : direct or indirect methods.

- Direct methods consist in a discretization of the optimal control problem in order to transform it into a parametric optimization problem. The resulting problem is of finite dimension and it can be addressed by a nonlinear programming algorithm. This approach is relatively straightforward and robust, because it allows the user to provide directly an initial command law and to modify it depending on the algorithm behaviour. The two main issues of these methods are the computation time (due to the size of the discretized problem) and the convergence accuracy. The quality of the optimum obtained is indeed downgraded by the discretization of the optimal control problem and by the numerical accuracy of the nonlinear programming algorithm.
- Indirect methods are based on the Pontryagin Maximum Principle which provides necessary optimality conditions. The problem is formulated as a two point boundary value problem: the unknowns are the components of the initial costate vector that must satisfy a nonlinear system at the final date. The solution is based on a shooting method using a Newton-like solver. The optimum is generally obtained very quickly and with a very high accuracy. The main issue lies in the costate initialization that must be sufficiently close to the optimum in order to allow the convergence.

In an industrial context, a major concern is the ability to solve efficiently and if possible automatically this problem. The time required by an engineer to assess an optimal launch trajectory has a direct influence on :

- The cost to prepare an operational mission (as for an Ariane 5 launch),
- The number of configurations (staging, propellants) that can be investigated in a preliminary design study for a future launcher.

Even if the experience on such problems allows fast assessments of optimal launcher trajectories, the engineer still faces the usual algorithmic difficulties related to numerical methods : initialization robustness, convergence rate, local optima. A practical know-how remains necessary as long as automatic and robust methods lack to cover the full range of possible missions and launchers configurations.

The proposed work aims at designing an automatic solution method for a launcher upper stage flight. The main optimization issues are indeed related to this flight phase, subject to path constraints and to final constraints.

The method envisioned is based on continuation algorithms. The principle of these algorithms consists in starting from a simplified problem whose solution is known to move progressively towards the real problem. Each intermediate problem is initialized with the solution of the previous problem.

Continuation methods applied to launcher trajectories can be considered nowadays as mature. But their implementation is not straightforward and it requires some practical know-how, with case by case adjustments, throughout the stages of the continuation process. They are therefore not directly applicable in an industrial context.

For the problem of a launcher upper stage flight, two continuation approaches are envisioned from model simplifications :

- The first approach consists in solving the flat Earth problem with uniform gravity, then to come back to the round Earth problem through a double continuation on the gravity and on the earth curvature.
- The second approach consists in solving the impulsive problem, corresponding to an infinite thrust level, , then to come back to the continuous thrust problem through a continuation on the thrust level.

The goal is to define a solution procedure as automatic as possible for various instances of the real problem : launcher configuration, targeted orbit.

Keywords :

Orbital transfer, Optimal control, Pontryagin Maximum Principle, Shooting method, Continuation

Space debris collecting

The uncontrolled proliferation of space debris since the beginning of the space age makes the access to space critical in the coming years. Various studies related to the space debris population and its evolution indicate that it would be mandatory to deorbit 5 heavy debris per year (launcher stages, satellites) from the low Earth orbits to stabilize the evolution.

Cleaning missions are therefore studied nowadays, consisting in launching a dedicated vehicle to capture successively several heavy debris and deorbit them by setting them on fall down trajectories.

In order to minimize the cost of such a cleaning mission, measured by the required propellant consumption, it is necessary to select the 5 debris among a list of candidates, to optimize the visiting order, and to optimize the vehicle maneuvers to go from a debris to the other. The whole mission must be achieved in less than one year.

The set of the candidate debris represents an oriented graph. The edge valuations are the transfer cost to go from a debris to the other, each debris being defined by its osculating orbit at the mission initial date. These orbits evolve during the mission under the influence of the perturbations of Earth gravitational field (Earth flattening).

Assessing the cost of a single transfer requires solving the associated optimal control problem which consists, starting from the initial debris to perform a rendezvous with the final debris. This problem is already intrinsically difficult in the general case. It depends moreover on the initial date because of the perturbation that modify the debris orbit throughout the time.

The selection of the debris and the visiting order is a shortest path problem in a time-dependent valuated graph. Efficient algorithms exist for this category of problems when the edge valuations are constants (« k-shortest path »), but they are not directly applicable to the space cleaning mission problem.

The global problem exhibits therefore a strong coupling between combinatorial aspects and optimal control aspects, each of these problems being intrinsically difficult. Trying to solve the problem in the general case is not reasonable with the known methods, and simplifying assumptions are necessary.

The proposed work consists in designing a solution method with the concerns of efficiency, representativity and optimality. With the perspective of a future cleaning vehicle, it is indeed desirable to be able to solve various instances of the problem in reasonable computation times, with some guarantee of optimality and practical feasibility.

The main assumptions that allow simplifying the problem formulation are related to the debris orbit, the vehicle engine and the mission duration. Most used satellites orbit on sun-synchronous orbits. The cleaning vehicle design is driven by this category of debris and the debris orbits are therefore assumed as low and circular. On the other hand using a high thrust engine allows modeling the transfer orbit maneuvers as impulses. Finally the one year allocation

for the whole mission gives enough time to insert waiting phases between the successive transfers. The maneuvers can thus be performed at the optimal dates in order to minimize the fuel consumption.

With these assumptions a generic strategy is defined that is applicable to every transfer. This generic strategy reduces the optimal control problem to a nonlinear programming problem of finite dimension, which can be coupled to the path problem. A local linearization around an initial solution allows to set up a Branch and Bound algorithm. The reference solution is then updated with the linearized solution. Iterating on the linearization-solution process from the pre-optimized initialization allows converging to the solution of the global problem.

Keywords :

Space debris, Orbital mechanics, Branch and bound, Linear programming

Related papers

[1] M. Cerf, T. Haberkorn, E. Trélat

Continuation from a flat to a round Earth model in the coplanar orbit transfer problem

Optimal Control Application and Methods, 36 pages, DOI: 10.1002/oca

Published online in Wiley InterScience (03/2011)

[2] M. Cerf

Multiple space debris collecting mission, debris selection and trajectory optimization

Journal of Optimization Theory and Applications, 33 pages, DOI: 10.1007/s10957-012-0130-6

Published online in Springer InterScience (08/2012)

[3] M. Cerf, T. Haberkorn, E. Trélat

Initialization of the high-thrust minimal consumption orbit transfer problem by the impulsive solution.

In progress.

Remerciements

Ma reconnaissance va en premier lieu à Emmanuel et Thomas, qui ont accepté d'être mes directeurs de thèse. Nous entretenons depuis plusieurs années des échanges très fructueux. Au-delà des travaux présentés dans ce document, notre collaboration a débouché sur des méthodes et logiciels utilisés aujourd'hui à Astrium dans un cadre opérationnel. Sans leur ingéniosité et leur intérêt pour résoudre des problèmes parfois peu mathématiques, tout ceci n'aurait pu voir le jour.

A leur contact, j'ai énormément progressé sur les méthodes numériques de commande optimale. Il me reste encore beaucoup à apprendre, et nous avons encore de belles perspectives de travail en commun sur les problèmes d'optimisation de trajectoires que je rencontre régulièrement à Astrium.

Je remercie également l'ensemble du personnel du laboratoire Jacques-Louis Lions pour son accueil, et en particulier son directeur Yvon Maday, pour avoir accepté ma candidature.

Yacine Chitour et Moritz Diehl ont accepté d'être rapporteurs pour mon mémoire de thèse. Je tiens à les remercier pour le temps qu'ils y ont consacré, leur lecture attentive et leurs commentaires constructifs.

Les travaux présentés dans ce document ont été suscités par des problèmes rencontrés dans le cadre de mon travail à Astrium. Il me faut remercier mes supérieurs successifs et mes collègues du service d'analyse de mission à Astrium. En me sollicitant régulièrement pour des problèmes variés, souvent difficiles, mais qu'il faut impérativement résoudre dans le cadre d'applications industrielles, tous me témoignent constamment de leur confiance. Ces sollicitations sont ma principale stimulation, et elles m'encouragent à constamment remettre mes connaissances en questions pour améliorer nos méthodes de travail. J'espère répondre le mieux possible à leurs attentes.

Enfin, et je garde le plus important pour la fin, rien de tout ceci ne serait possible sans Laurence, Michaël, Fabien et Sophie qui m'entourent chaque jour de leur affection et de leur joie de vivre.

Part 1

Launcher upper stage flight

Optimal coplanar transfer by continuation

from the flat Earth model or from the impulsive model.

Abstract We consider the problem of the minimum consumption trajectory for a launcher upper stage. The upper stage mission is a coplanar transfer consisting in bringing the payload on the targeted orbit, starting from given injection conditions. This problem has been very much investigated in the literature. We are interested here in solving this problem by means of a shooting method, based on the application of the Pontryagin Maximum Principle. The convergence of the shooting method is made difficult, because of initialization issues and discontinuities of the optimal control law. Several processes are already known in order to overcome this convergence problem. We consider here two different approaches, starting either from the flat Earth model with constant gravity, or from the impulsive model.

These simplified problems have a nearly explicit solution that can be found efficiently without any specific initialization work. Their solutions are used as starting point for a numerical continuation transforming the simplified problem into the initial problem. The whole process is carried out automatically from scratch and yields the upper stage optimal trajectory with reduced computation times. The methods are exemplified on practical application cases.

Keywords

Orbit transfer problem, Optimal control, Pontryagin Maximum Principle, Shooting method, Continuation

Acronyms

PMP	Pontryagin Maximum Principle
BVP	Boundary Value Problem
NLP	Non Linear Programming
SSO	Sun-Synchronous Orbit

Summary

1	Introduction	17
1.1	Problem statement.....	17
1.2	Shooting method.....	18
1.3	Continuation from simplified models.....	18
2	Problem formulation.....	20
2.1	Pontryagin Maximum Principle (PMP).....	20
2.2	Boundary value problem (BVP) and shooting method	21
2.3	General transfer problem	23
2.4	Hamiltonian and costate.....	25
2.5	Application of the PMP.....	26
2.6	Switching function.....	29
2.7	Abnormal extremals.....	29
2.8	Coplanar transfer problem.....	30
2.9	Orbital parameters.....	31
3	Flat Earth model	32
3.1	Problem statement.....	32
3.2	Application of the Pontryagin Maximum Principle.....	34
3.3	Switching function.....	35
3.4	Singular arcs	35
3.5	Switching function variation	36
3.6	Case of a negative minimum	38
3.7	Optimal sequences	42
3.8	Algorithmic procedure	44
4	Continuation from the flat Earth to the round Earth model.....	50
4.1	Flat Earth model	50
4.2	Round Earth model	51
4.3	Modified flat Earth model.....	55
4.4	Continuation procedure.....	59
4.5	Application of the PMP.....	62
4.6	Practical implementation.....	65
4.7	Round Earth problem costate	66
4.8	Final conditions	70

5	Bi-impulse model	71
5.1	Problem statement.....	71
5.2	Initial costate assessment	75
5.3	Duration assessment.....	78
5.4	Thrust level continuation	79
6	Application examples	80
6.1	Example data	80
6.2	Continuation procedure from the flat Earth model	82
6.3	Continuation procedure from the impulsive model.....	91
6.4	3-dimensional transfer	96
7	Conclusion.....	101
7.1	Methods proposed.....	101
7.2	Current status.....	102
7.3	Perspectives	102

List of Figures

Figure 1 : Flat Earth model.....	32
Figure 2 : Round Earth model	52
Figure 3 : Coordinate change from the round Earth to the flat Earth model.....	56
Figure 4 : Lambert transfer.....	74
Figure 5 : Hohmann-like transfer.....	77
Figure 6 : Altitude increase in the round Earth model.....	84
Figure 7 : Continuation path.....	85
Figure 8 : Commutation dates and final mass.....	86
Figure 9 : Trajectory variables.....	87
Figure 10 : Switching function - Control.....	87
Figure 11 : Final time continuation	88
Figure 12 : Bi-impulse transfer performance vs transfer time (SSO).....	91
Figure 13 : Optimal trajectories vs thrust level (SSO).....	94

List of Tables

Table 1 : Switching function variation in the case $p_r \neq 0$	37
Table 2 : Vertical velocity variation in the case $p_n < 0$	41
Table 3 : Initial and final conditions (SSO).....	80
Table 4 : Initial and final conditions (GEO).....	81
Table 5 : Initialization.....	82
Table 6 : Gravity continuation.....	82
Table 7 : Dynamics continuation.....	83
Table 8 : Final time continuation.....	89
Table 9 : Results recap.....	90
Table 10 : Thrust level continuation (SSO 2-dimensional transfer).....	93
Table 11 : Velocity losses (SSO 2-dimensional transfer).....	95
Table 12 : Initial costate guess (GEO).....	96
Table 13 : Thrust level continuation (out of plane velocity).....	97
Table 14 : Velocity losses (out of plane velocity).....	98
Table 15 : Thrust level continuation (out of plane position).....	99
Table 16 : Velocity losses (out of plane position).....	99

1 Introduction

The trajectory of a launcher from the lift-off to the orbit is split into successive sequences corresponding to the flight of the propulsive stages. The global trajectory optimization problem handles a large number of variables and constraints with different scales throughout the trajectory. Achieving an accurate convergence is generally a painful task requiring the supervision of an experienced user. The goal of the present work is to build an automatic solution method applied to the upper stage flight. The resulting solver should be fast, accurate and robust enough to be incorporated within a simulation tool. By this way the global trajectory optimization problem could reduce to the lower stages flight, speeding up the solution and easing the practical use.

1.1 Problem statement

The problem of the orbit transfer at minimum consumption has been widely studied from the 1960's^{1,2} and it is still a topic of current research. We can distinguish mainly between two formulations of this problem. The first one considers that the vehicle produces instantaneous changes of velocity and is referred to as the impulse orbit transfer^{3,4,5}. The second formulation takes into account the fact that an actual engine has a limited thrust so that the vehicle's dynamics is continuous in the position and velocity coordinates⁶. In this continuous approach, we also separate the high-thrust and the low-thrust transfer, depending on the available acceleration level.

We focus here on the high thrust orbit transfer that we furthermore restrict to be coplanar. This problem is naturally written as an optimal control problem. There exist various numerical methods to solve such problems, and we usually separate them in two classes: direct and indirect methods.

- Direct methods⁷ consist in discretizing the optimal control problem in order to rewrite it as a parametric optimization problem. Then a nonlinear large scale optimization solver is applied. The advantage of this approach is that it is straightforward and usually quite robust. New variables or constraints can also be added to the problem with reduced programming effort, making the method very flexible. Due to this flexibility and the availability of numerous efficient software packages such as IPOPT (with AMPL), BOCOP, GESOP, ..., the direct methods are widely applied for space trajectory optimization, as well for operational launchers like Ariane 5, as for preliminary studies of future launchers. The main drawback is that those methods are computationally demanding and that they are not very accurate when compared with the indirect approach⁷. They are therefore not well suited to build the envisioned solver.
- Indirect methods are based on the Pontryagin Maximum Principle⁸ (PMP) which gives a set of necessary conditions for a candidate trajectory and control strategy to be optimal. The idea is to use those necessary conditions to reduce the search of a solution to the search of the zero of the so-called shooting function (indirect methods are also called shooting methods in this context). The advantage is that shooting methods are very fast when they converge and that they produce high accuracy solutions. Their main drawback is that, since they use a Newton-like algorithm to look for the zero of the shooting function, they are hard to properly initialize. We can also mention mixed methods that use a discretization of the PMP necessary conditions and then apply a large-scale equation solver⁹.

1.2 Shooting method

The goal is to build an automatic, fast and accurate solver applied to an upper stage flight performing a coplanar orbit transfer. Because of its fast convergence and high accuracy, we will turn to a shooting method to solve the problem. There already exist some methods to cope with the initialization drawback of this method. In Ref¹⁰, the impulse transfer solution is used to provide a good initial guess to the shooting algorithm. This method is based on the fact that a continuous high-thrust orbit transfer tries to mimic an impulse transfer, as outlined in Refs^{4,11}. However this approach is only valid for nearly circular initial and final orbits. In Ref¹², a multiple shooting method parameterized by the number of thrust arcs, is used to solve an Earth-Mars transfer. In Ref^{13,14}, a differential continuation method linking the minimization of the L^2 -norm of the control to the minimization of the consumption is used to solve the low-thrust orbit transfer around the Earth. However this approach is not adapted for a high-thrust transfer. In Ref¹⁵ simplified formulas are established by interpolating many numerical experiments, which permit to initialize successfully the shooting method for the minimal time orbit transfer problem, in a certain range of values for nearly circular initial and final orbits. Based on that initial guess and on averaging techniques, the authors of Ref¹⁶ implement in the software T3D continuation and smoothing processes in order to solve minimal time or minimal fuel consumption orbit transfer problems.

1.3 Continuation from simplified models

Two different approaches are envisioned to initialize a shooting method for a high-thrust coplanar orbit transfer. They are based on model simplifications, which allow solving easily the transfer problem, followed by continuation methods to come back to the 'real' problem. The simplifications envisioned are respectively related to the Earth modeling and to the thrust modeling :

- The first approach consists in starting from the flat Earth model (infinite curvature radius) and continuously decreasing the curvature radius to end up with the 'real' round Earth model.
- The second approach consists in starting from the impulsive model (infinite thrust level) and continuously decreasing the thrust level to end up with the 'real' continuous thrust model.

We restrict ourselves to fixed final time problems since it has been shown numerically that the minimum consumption orbit transfer does not have a solution at free final time^{11,12}.

The text is organized as follows. The Pontryagin Maximum Principle is first applied to the general minimum consumption orbit transfer without any simplification. Since we are interested in the flight of a launcher upper stage, we restrict the problem to the coplanar orbit transfer which is representative of a large part of practical applications. The problem is further analyzed considering model simplifications : flat Earth model or impulsive model. Efficient algorithms are proposed that solve these simplified problems from scratch. A parameterized problem is then defined that allows passing continuously from the simplified problem to the 'real' problem :

- The flat Earth model is modified by additive terms so as to introduce curvature and make it diffeomorphic to the round Earth model. A continuation procedure is set up that introduces progressively the additive terms in the model. This allows passing from the flat Earth problem solution to the targeted round Earth problem solution.
- The impulsive model is used to initialize the continuous thrust model considering a fictitious very high thrust level. The continuation procedure consists in decreasing progressively the thrust level until the targeted vehicle thrust level.

The methods are illustrated on application examples representative of launcher upper stages targeting either a sun-synchronous or a geostationary orbit.

2 Problem formulation

This chapter introduces the mathematical formulation of the optimal control problem. The Pontryagin Maximum Principle is first recalled. It is applied to the general 3-dimensional transfer problem for a launcher upper stage, yielding a boundary value problem. The coplanar transfer problem is then directly derived from the general 3-dimensional formulation by reducing the state and control dimensions.

2.1 Pontryagin Maximum Principle (PMP)

We consider a dynamic system, evolving between a fixed initial date t_0 and a free final date t_f . The system is represented at any date $t \in [t_0, t_f]$ by a n_x -state vector $X(t)$ and it is controlled by a n_u -command vector $U(t)$ belonging to a subset U_a of \mathbb{R}^{n_u} representing the admissible controls. The system dynamics is modeled by a first order ODE :

$$\begin{aligned} \dot{X}(t) &= f[X(t), U(t), t] \quad \text{for } t \in [t_0, t_f] \\ f &: \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R} \rightarrow \mathbb{R}^{n_x} \end{aligned} \quad (1)$$

The initial state is fixed : $X(t_0) = X_0$, whereas the final state is subject to m final constraints :

$$\begin{aligned} \Psi[X(t_f), t_f] &= 0 \\ \Psi &: \mathbb{R}^{n_x} \times \mathbb{R} \rightarrow \mathbb{R}^m \end{aligned} \quad (2)$$

The optimal control problem denoted **(OCP)** is formulated under the Mayer form with a final cost J to minimize :

$$\begin{aligned} \underset{U(\cdot), t_f}{\text{Min}} J[X(t_f), t_f] \quad \text{s.t.} \quad & \begin{cases} \dot{X}(t) = f[X(t), U(t), t] & \text{for } t \in [t_0, t_f] \\ X(t_0) = X_0 \\ \Psi[X(t_f), t_f] = 0 \end{cases} \\ J &: \mathbb{R}^{n_x} \times \mathbb{R} \rightarrow \mathbb{R} \end{aligned} \quad (3)$$

According to the Pontryagin Maximum Principle, every optimal trajectory $X(\cdot)$ of **(OCP)** is the projection of an extremal $(X(\cdot), P(\cdot), p^0, U(\cdot))$ on the state space. An extremal is associated with :

- a control $U(\cdot)$ on $[t_0, t_f]$,
- an absolutely continuous mapping $P(\cdot): [t_0, t_f] \rightarrow \mathbb{R}^{n_x}$, named the costate (or adjoint) vector,
- a non positive real number p^0 , with $(P(\cdot), p^0) \neq (0, 0)$
- a Hamiltonian function

$$H[X(t), P(t), p^0, U(t), t] = P(t)^t \cdot \dot{X}(t) = P(t)^t \cdot f[X(t), U(t), t] \quad (4)$$

and it satisfies :

- the Hamiltonian system :

$$\begin{cases} \dot{X}(t) = \frac{\partial H}{\partial P} [X(t), P(t), p^0, U(t), t] \\ \dot{P}(t) = -\frac{\partial H}{\partial X} [X(t), P(t), p^0, U(t), t] \end{cases} \quad \text{for almost every } t \in [t_0, t_f] \quad (5)$$

- the maximization of the Hamiltonian :

$$H[X(t), P(t), p^0, U(t), t] = \text{Max}_{W \in U_a} H[X(t), P(t), p^0, W, t] \quad \text{for almost every } t \in [t_0, t_f] \quad (6)$$

- the transversality conditions on the Hamiltonian (if the final date is free) and on the costate :

$$\begin{cases} H(t_f) = -p^0 \frac{\partial J}{\partial t} [X(t_f), t_f] - v^t \frac{\partial \Psi}{\partial t} [X(t_f), t_f] \\ P(t_f) = p^0 \frac{\partial J}{\partial X} [X(t_f), t_f] + v^t \frac{\partial \Psi}{\partial X} [X(t_f), t_f] \end{cases} \quad \text{with scalar multipliers } v \in \mathbb{R}^m \text{ and } p^0 \leq 0 \quad (7)$$

Similar transversality conditions hold at the initial date if the initial conditions are not fixed. When the initial state is fixed, the initial costate is free.

The extremal $(X(\cdot), P(\cdot), p^0, U(\cdot))$ is said normal if $p^0 \neq 0$, abnormal if $p^0 = 0$. In the normal case, it is usual to normalize the costate vector so that $p^0 = -1$, but any other choice with p^0 strictly negative is valid.

For an autonomous system (i.e. the dynamics, the cost and the final constraints do not depend explicitly on the time), the Hamiltonian is constant along any extremal. It is moreover equal to zero if the final time is free.

2.2 Boundary value problem (BVP) and shooting method

2.2.1 Boundary value problem

Applying the PMP yields a boundary value problem representing the necessary conditions for a minimum cost :

- Unknowns : $P(t_0)$ $\rightarrow n_x$ unknowns (state dimension)
 v $\rightarrow m$ unknowns (final constraint multipliers)
 t_f $\rightarrow 1$ unknown if the final time is free
- Equations : Transversality on $P(t_f)$ $\rightarrow n_x$ equations
Final constraints $\Psi[X(t_f), t_f] = 0$ $\rightarrow m$ equations
Transversality on $H(t_f)$ $\rightarrow 1$ equation if the final time is free

The BVP is a nonlinear system of dimension $n_x + m$ (+1 if the final time is free). The single shooting method consists in finding a zero of the shooting function $S : \mathbb{R}^{n_x + m + 1} \rightarrow \mathbb{R}^{n_x + m + 1}$ defined as :

$$S[P(t_0), \nu, t_f] = \begin{pmatrix} \Psi[X(t_f), t_f] \\ P(t_f) - p^0 \frac{\partial J}{\partial X}[X(t_f), t_f] - \nu^t \frac{\partial \Psi}{\partial X}[X(t_f), t_f] \\ H(t_f) + p^0 \frac{\partial J}{\partial t}[X(t_f), t_f] + \nu^t \frac{\partial \Psi}{\partial t}[X(t_f), t_f] \end{pmatrix} \quad \text{with } p^0 \leq 0 \quad (8)$$

2.2.2 Fixed final state

A simplification arises when the final constraints are expressed directly on the final state components. Assuming that the final state $X(t_f)$ has its first m components $X_{1-m}(t_f)$ specified to X_{1-m}^f , while the $n_x - m$ last components $X_{m+1-n_x}(t_f)$ are free, the final constraints take the form :

$$\Psi[X(t_f), t_f] = X_{1-m}(t_f) - X_{1-m}^f = 0 \Rightarrow \begin{cases} \frac{\partial \Psi}{\partial X_i}[X(t_f), t_f] = 1 & \text{for } i = 1 \text{ to } m \\ \frac{\partial \Psi}{\partial X_i}[X(t_f), t_f] = 0 & \text{for } i = m+1 \text{ to } n_x \end{cases} \quad (9)$$

In that case, the cost function no longer depends on the specified components $X_{1-m}(t_f)$:

$$\frac{\partial J}{\partial X_i}[X(t_f), t_f] = 0 \quad \text{for } i = 1 \text{ to } m \quad (10)$$

The transversality conditions take the form :

$$\begin{cases} P_i(t_f) = \nu_i & \text{for } i = 1 \text{ to } m \\ P_i(t_f) = p^0 \frac{\partial J}{\partial X_i}[X(t_f), t_f] & \text{for } i = m+1 \text{ to } n_x \end{cases} \quad (11)$$

The first m transversality conditions mean that the corresponding final costate components are “free” (or unknown), i.e. they are determined implicitly by solving the BVP. These conditions can thus be discarded from the shooting function together with the unknown multipliers ν_{1-m} . The BVP dimension is reduced to n_x (+1 if the final time is free) and the shooting function $S: \mathbb{R}^{n_x+1} \rightarrow \mathbb{R}^{n_x+1}$ is defined as :

$$S[P(t_0), t_f] = \begin{pmatrix} X_i(t_f) - X_i^f & \text{for } i = 1 \text{ to } m \\ P_i(t_f) - p^0 \frac{\partial J}{\partial X_i}[X(t_f), t_f] & \text{for } i = m+1 \text{ to } n_x \\ H(t_f) \end{pmatrix} \quad \text{with } p^0 \leq 0 \quad (12)$$

2.2.3 Shooting method

Computing the shooting function values requires integrating the state and costate equations starting from the fixed initial state $X(t_0)$ and a guessed initial costate $P(t_0)$. The Hamiltonian system (5) is integrated numerically from the initial date t_0 until a guessed final date t_f . The control $U(t)$ along the trajectory is determined ‘on line’ by the maximization condition (6) on the Hamiltonian.

This numerical integration is combined with a Newton-like algorithm to find a zero of the shooting function S . The main issue comes from the Newton method convergence properties. For highly nonlinear equations, the algorithm converges only in the near vicinity of the solution, so that a very accurate initial guess must be provided to achieve the solution. An additional difficulty comes from the control discontinuities that occur when the Hamiltonian is linear with respect to the control, as it is the case for the upper stage transfer problem. A poor guess of the initial costate generally fails to catch the optimal control structure, and therefore does not allow converging on the solution.

2.3 General transfer problem

The general transfer problem for a launcher upper stage is first formulated in the 3-dimensional space, without specifying the coordinate system. As this stage, the formulation is indeed identical whatever the dimension (2 or 3). At the end of Chapter §2 the problem will be reduced to the 2-dimensional space in order to set up the continuation procedure for the coplanar transfer problem.

2.3.1 General 3-dimensional formulation

The general transfer problem consists in finding the minimal consumption trajectory to go from given injection conditions to a targeted orbit defined by orbital parameters. For this 3-dimensional problem, the Earth is modeled as a sphere, with its center taken as the origin of an inertial frame. The vehicle is considered as a material point

represented by the 7-state vector $X(t) = \begin{pmatrix} \vec{r}(t) = \text{position} \\ \vec{v}(t) = \text{velocity} \\ m(t) = \text{mass} \end{pmatrix} \in \mathbb{R}^7$. The cost function is the final mass $J[X(t_f)] = m(t_f)$ to

maximize. The vehicle is submitted to the Earth acceleration gravity $\vec{g}(\vec{r})$ and the transfer trajectory is controlled by the 3-command vector $U(t) \in \mathbb{R}^3$ representing the vehicle thrust. Denoting the burned propellant exhaust velocity by v_e , the thrust magnitude by $T(t)$, $T(t) = -\dot{m}(t)v_e$, $0 \leq T(t) \leq T_{\max}$, the thrust direction by $\vec{u}(t)$, $\|\vec{u}(t)\| = 1$, the command vector is $U(t) = T(t)\vec{u}(t)$.

Applying the fundamental dynamics principle in the inertial frame yields :

$$\frac{d^2 \vec{r}(t)}{dt^2} = \frac{T(t)}{m(t)} \vec{u}(t) + \vec{g}(\vec{r}(t)) \Leftrightarrow \begin{cases} \dot{\vec{r}}(t) = \vec{v}(t) \\ \dot{\vec{v}}(t) = \frac{T(t)}{m(t)} \vec{u}(t) + \vec{g}(\vec{r}(t)) \end{cases}$$

The vehicle dynamics is thus represented by the first order ODE :

$$\dot{X}(t) = f[X(t), U(t)] \Leftrightarrow \begin{cases} \dot{\vec{r}}(t) = \vec{v}(t) \\ \dot{\vec{v}}(t) = \frac{T(t)}{m(t)} \vec{u}(t) + \vec{g}(\vec{r}(t)) \\ \dot{m}(t) = -\frac{T(t)}{v_e} \end{cases} \quad \text{for } t \in [t_0, t_f] \quad (13)$$

The system is autonomous since the dynamics does not depend explicitly on the time.

The initial state is given, corresponding to the position, velocity and gross mass at the upper stage injection :

$$X(t_0) = X_0 \Leftrightarrow \begin{cases} \vec{r}(t_0) = \vec{r}_0 \\ \vec{v}(t_0) = \vec{v}_0 \\ m(t_0) = m_0 \end{cases} \quad (14)$$

The final state is constrained on the targeted orbit, specified by its energy K_f , its angular momentum \vec{h}_f and its eccentricity vector \vec{e}_f . These 5 final constraints do not depend explicitly on the time and they take the form¹⁷ :

$$\Psi[X(t_f)] = 0 \Leftrightarrow \begin{cases} \frac{v(t_f)^2}{2} - \frac{\mu}{r(t_f)} = K_f \\ \vec{r}(t_f) \wedge \vec{v}(t_f) = \vec{h}_f \\ \vec{v}(t_f) \wedge \vec{h}(t_f) - \frac{\mu}{r(t_f)} \vec{r}(t_f) = \mu \vec{e}_f \end{cases} \quad \text{with } \Psi : \mathbb{R}^6 \rightarrow \mathbb{R}^5 \quad (15)$$

where μ is the gravitational constant of the attracting body (here the Earth).

There is generally no rendezvous constraint applied to the upper stage trajectory. The rendezvous constraint is managed later by the satellite once released by the upper stage. The injection anomaly θ_f is therefore free.

Assumption 1.

In order to have a 'real' optimal control problem to solve, we assume that the initial state vector $X(t_0)$ does not satisfy the final constraints $\Psi[X]=0$. In other terms, the initial point does not belong to the targeted orbit. Consequently the null thrust solution is discarded and any solution must have at least one thrust arc.

2.3.2 Final time

Without any bound on the mission duration, the minimal consumption solution is in infinite time^{6,7} and it consists in an infinite series of impulsive maneuvers at each revolution.

For a practical application there are always operational constraints bounding the mission duration, e.g. :

- the engine limited re-ignition capabilities, which limit the number of thrust arcs,
- the tracking capabilities which limit the number of revolutions.

With such an upper bound, the minimum consumption transfer problem is well-posed. Fixing the final time at the upper bound yields an equivalent problem. Indeed it is always possible to add a coast arc to any solution, without changing the total consumption. From now on we thus consider that the optimal control problem is with a fixed final time.

Since the above actual constraints are rather formulated in terms of maximal number of ignitions or of revolutions, the final time value cannot be fixed directly. It becomes necessary to solve a series of optimal control problems with

different values of the final time t_f in order to get the optimal solution of the transfer problem compliant with the operational constraints.

2.3.3 Final anomaly

If a particular injection anomaly θ_f is chosen at the fixed final time t_f , the final position and velocity vectors are completely fixed by 6 final constraints. In that case the final constraints can be expressed directly using the state vector components instead of the orbital parameters :

$$\Psi[X(t_f)] = 0 \Leftrightarrow \begin{cases} \bar{\mathbf{r}}(t_f) = \bar{\mathbf{r}}_f \\ \bar{\mathbf{v}}(t_f) = \bar{\mathbf{v}}_f \end{cases} \quad \text{with } \Psi : \mathbb{R}^6 \rightarrow \mathbb{R}^6 \quad (16)$$

This formulation (16) of the final constraints is easier to handle than (15) in the numerical solution. Indeed the position and velocity costates become free, so that the associated transversality conditions can be discarded from the problem formulation (cf §2.2.2). Practically there is generally no rendezvous constraint and the injection anomaly θ_f is free. The formulation (16) can still be used, but it becomes necessary to solve a series of optimal control problems with different values of the injection anomaly θ_f in order to get the optimal solution of the transfer problem.

For a launcher upper stage, the optimal injection is generally located in the vicinity of the perigee of the targeted orbit. Thus there exists a local optimum corresponding to a prescribed number of revolutions.

2.4 Hamiltonian and costate

The costate vector is defined as : $\mathbf{P}(t) = \begin{pmatrix} \bar{\mathbf{p}}_r(t) = \text{position costate} \\ \bar{\mathbf{p}}_v(t) = \text{velocity costate} \\ \mathbf{p}_m(t) = \text{mass costate} \end{pmatrix} \in \mathbb{R}^7$.

The Hamiltonian of the transfer problem is :

$$\begin{aligned} \mathbf{H}[X(t), \mathbf{P}(t), \mathbf{U}(t)] &= \bar{\mathbf{p}}_r(t)^t \dot{\bar{\mathbf{r}}}(t) + \bar{\mathbf{p}}_v(t)^t \dot{\bar{\mathbf{v}}}(t) + \mathbf{p}_m(t) \dot{m}(t) \\ &= \bar{\mathbf{p}}_r(t)^t \bar{\mathbf{v}}(t) + \bar{\mathbf{p}}_v(t)^t \left(\frac{\mathbf{T}(t)}{m(t)} \bar{\mathbf{u}}(t) + \bar{\mathbf{g}}(\bar{\mathbf{r}}(t)) \right) - \mathbf{p}_m(t) \frac{\mathbf{T}(t)}{v_e} \\ &= \bar{\mathbf{p}}_r(t)^t \bar{\mathbf{v}}(t) + \bar{\mathbf{p}}_v(t)^t \bar{\mathbf{g}}(\bar{\mathbf{r}}(t)) + \mathbf{T}(t) \left(\frac{\bar{\mathbf{p}}_v(t)^t \bar{\mathbf{u}}(t)}{m(t)} - \frac{\mathbf{p}_m(t)}{v_e} \right) \end{aligned} \quad (17)$$

The costate equations are :

$$\begin{cases} \dot{\bar{\mathbf{p}}}_r(t) = -\frac{\partial \bar{\mathbf{g}}(\bar{\mathbf{r}}(t))}{\partial \bar{\mathbf{r}}} \bar{\mathbf{p}}_v(t) \\ \dot{\bar{\mathbf{p}}}_v(t) = -\bar{\mathbf{p}}_r(t) \\ \dot{\mathbf{p}}_m(t) = \frac{\mathbf{T}(t)}{m(t)^2} \bar{\mathbf{p}}_v(t)^t \bar{\mathbf{u}}(t) \end{cases} \quad \text{for } t \in [t_0, t_f] \quad (18)$$

For a central gravitational field : $\bar{\mathbf{g}}(\bar{\mathbf{r}}) = -\frac{\mu}{r^3} \bar{\mathbf{r}}$, the gravity gradient is the 3x3 matrix¹⁷ :

$$\frac{\partial \bar{g}(\bar{r})}{\partial \bar{r}} = -\frac{\mu}{r^5} (r^2 \mathbf{I} - 3\bar{r}\bar{r}^t) = \frac{\mu}{r^5} \begin{pmatrix} 3x^2 - r^2 & 3xy & 3xz \\ 3xy & 3y^2 - r^2 & 3yz \\ 3xz & 3yz & 3z^2 - r^2 \end{pmatrix} \quad (19)$$

with $\bar{r} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$ in rectangular coordinates.

2.5 Application of the PMP

Applying the PMP to the upper stage transfer problem yields successively the transversality conditions, the optimal control, and the Hamiltonian value.

2.5.1 Transversality conditions

In the case of fixed injection position and velocity (when the anomaly injection is specified), the final constraints take the simplified form (9) :

$$\Psi[\mathbf{X}(t_f)] = \begin{pmatrix} \bar{r}(t_f) - \bar{r}_f \\ \bar{v}(t_f) - \bar{v}_f \end{pmatrix} = 0 \Rightarrow \begin{cases} \frac{\partial \Psi}{\partial \bar{r}}[\mathbf{X}(t_f)] = 1 \\ \frac{\partial \Psi}{\partial \bar{v}}[\mathbf{X}(t_f)] = 1 \end{cases} \quad (20)$$

The cost function depends only on the mass :

$$J[\mathbf{X}(t_f)] = m(t_f) \Rightarrow \begin{cases} \frac{\partial J}{\partial \bar{r}}[\mathbf{X}(t_f)] = 0 \\ \frac{\partial J}{\partial \bar{v}}[\mathbf{X}(t_f)] = 0 \end{cases} \quad (21)$$

The transversality conditions on the position and velocity costates can then be discarded from the problem formulation together with their multipliers v_r and v_v (cf §2.2.2).

There remains only the transversality condition on the mass costate :

$$p_m(t_f) = p^0 \frac{\partial J}{\partial m}[\mathbf{X}(t_f)] = -p^0 \geq 0 \quad (22)$$

The costate vector is defined up to a multiplicative scalar. The usual normalization is either $p^0=0$ (abnormal case) or $p^0=-1$ whenever the corresponding extremal is normal. In what follows we will prove that, under an adequate assumption, the abnormal case does not occur in our problem.

2.5.2 Control

Before looking for the optimal control, we establish the following lemma.

Lemma 1. The function $t \mapsto \|\bar{\mathbf{p}}_v(t)\|$ does not vanish identically on any subinterval of $[t_0, t_f]$.

Proof. The argument goes by contradiction. Assume that $\|\bar{\mathbf{p}}_v(t)\| = 0$ on a subinterval I of $[t_0, t_f]$.

$$\text{Replacing } \bar{\mathbf{p}}_v(t) = 0 \text{ in (18) gives : } \begin{cases} \bar{\mathbf{p}}_v(t) = 0 \\ \dot{\bar{\mathbf{p}}}_v(t) = \bar{\mathbf{p}}_r(t) = 0 & \text{on I.} \\ \dot{\mathbf{p}}_m(t) = 0 \Rightarrow \mathbf{p}_m(t) = -\mathbf{p}^0 \end{cases}$$

$$\text{By Cauchy uniqueness of the solution : } \begin{cases} \bar{\mathbf{p}}_v(t) = 0 \\ \bar{\mathbf{p}}_r(t) = 0 & \text{on } [t_0, t_f]. \\ \mathbf{p}_m(t) = -\mathbf{p}^0 \end{cases}$$

Replacing in the Hamiltonian (17) yields :

$$H(t) = T(t) \frac{\mathbf{p}^0}{v_e}, \quad \forall t \in [t_0, t_f]$$

From the PMP the scalar \mathbf{p}^0 is either negative or null :

- If $\mathbf{p}^0 < 0$, then the Hamiltonian maximization requires $T(t) = 0, \forall t \in [t_0, t_f]$.
This raises a contradiction with Assumption 1 which discards the null thrust solution.
- If $\mathbf{p}^0 = 0$, then $\mathbf{p}_m(t) = 0, \forall t \in [t_0, t_f]$.

We have simultaneously $\begin{cases} P(t) = 0, \forall t \in [t_0, t_f] \\ \mathbf{p}^0 = 0 \end{cases}$ which causes a contradiction with the PMP. \square

Applying now the PMP, the optimal control $U(\cdot)$ maximizes the Hamiltonian for almost every $t \in [t_0, t_f]$:

$$\begin{aligned} \text{Max}_{U(t)} H[X(t), P(t), U(t)] &= \text{Max}_{T(t), \bar{\mathbf{u}}(t)} \left[\bar{\mathbf{p}}_r(t)^t \bar{\mathbf{v}}(t) + \bar{\mathbf{p}}_v(t)^t \bar{\mathbf{g}}(\bar{\mathbf{r}}(t)) + T(t) \left(\frac{\bar{\mathbf{p}}_v(t)^t \bar{\mathbf{u}}(t)}{m(t)} - \frac{\mathbf{p}_m(t)}{v_e} \right) \right] \\ \text{s.t. } &\begin{cases} 0 \leq T(t) \leq T_{\max} \\ \|\bar{\mathbf{u}}(t)\| = 1 \end{cases} \end{aligned} \quad (23)$$

The Hamiltonian maximization yields firstly the optimal thrust direction along the velocity costate :

$$\begin{aligned} \text{Max}_{T(t), \bar{\mathbf{u}}(t)} \left[T(t) \left(\frac{\bar{\mathbf{p}}_v(t)^t \bar{\mathbf{u}}(t)}{m(t)} - \frac{\mathbf{p}_m(t)}{v_e} \right) \right] \text{ s.t. } &\begin{cases} 0 \leq T(t) \leq T_{\max} \\ \|\bar{\mathbf{u}}(t)\| = 1 \end{cases} \\ \Rightarrow \bar{\mathbf{u}}(t) = \frac{\bar{\mathbf{p}}_v(t)}{\|\bar{\mathbf{p}}_v(t)\|} &\text{ if } T(t) > 0 \text{ and } \|\bar{\mathbf{p}}_v(t)\| \neq 0 \end{aligned} \quad (24)$$

For a null thrust the Hamiltonian is insensitive to the thrust direction, and from Lemma 1, the function $t \mapsto \|\bar{\mathbf{p}}_v(t)\|$ does not vanish identically on any subinterval of $[t_0, t_f]$. The thrust direction $\bar{\mathbf{u}}$ is thus well defined almost everywhere by (24).

Replacing the optimal thrust direction $\bar{\mathbf{u}}$ in the Hamiltonian (17) leads us to define the switching function Φ on $[t_0, t_f]$:

$$\Phi(t) = \frac{\|\bar{\mathbf{p}}_v(t)\|}{m(t)} - \frac{p_m(t)}{v_e} \quad (25)$$

The Hamiltonian can then be expressed as :

$$H[\mathbf{X}(t), \mathbf{P}(t), \mathbf{U}(t)] = \bar{\mathbf{p}}_r(t)^t \bar{\mathbf{v}}(t) + \bar{\mathbf{p}}_v(t)^t \bar{\mathbf{g}}(\bar{\mathbf{r}}(t)) + T(t)\Phi(t) \quad \text{if } T(t) > 0 \text{ and } \|\bar{\mathbf{p}}_v(t)\| \neq 0 \quad (26)$$

The Hamiltonian maximization requires secondly choosing the thrust magnitude so that :

$$\begin{cases} T(t) = 0 & \text{if } \Phi(t) < 0 \\ T(t) = T_{\max} & \text{if } \Phi(t) > 0 \end{cases} \quad (27)$$

The PMP does not yield directly the thrust magnitude when the switching function is null.

Singular arcs correspond to a switching function that vanishes identically on a subinterval I of on $[t_0, t_f]$. The non-occurrence of singular arcs must be checked from the numerical simulations. A study of the controllability aspects of the orbit transfer problems can be found in Refs^{1,13,18,19}.

In the simplified case of a flat Earth model (cf §3), it will be shown that singular arcs can be discarded through an adequate assumption on the transfer endpoints.

Replacing the optimal thrust direction $\bar{\mathbf{u}}$ in the mass costate equation (18) yields :

$$\dot{p}_m(t) = \frac{T(t)\|\bar{\mathbf{p}}_v(t)\|}{m(t)^2} \geq 0 \quad , \quad \forall t \in [t_0, t_f] \quad (28)$$

The mass costate is thus non-decreasing along any extremal.

2.5.3 Hamiltonian value

Since the transfer problem is autonomous, the Hamiltonian is constant :

$$H[\mathbf{X}(t), \mathbf{P}(t), \mathbf{U}(t)] = \bar{\mathbf{p}}_r(t)^t \bar{\mathbf{v}}(t) + \bar{\mathbf{p}}_v(t)^t \bar{\mathbf{g}}(\bar{\mathbf{r}}(t)) + T(t)\Phi(t) = H_f \quad , \quad \forall t \in [t_0, t_f] \quad (29)$$

Moreover the cost J and the final constraints Ψ do not depend explicitly on the time, so that if the final time is free, the Hamiltonian is null along any extremal. For the general transfer problem, it is necessary to fix the final time in order to get a practically feasible solution (cf §2.3.2). The Hamiltonian takes then a non-null value.

2.6 Switching function

The sign of the switching function : $t \in [t_0, t_f] \mapsto \Phi(t) = \frac{\|\bar{p}_v(t)\|}{m(t)} - \frac{p_m(t)}{v_e}$ determines the sequences of thrust and coast arcs. The function Φ is differentiable almost everywhere in $[t_0, t_f]$. For every $t \in [t_0, t_f]$ such that $\|\bar{p}_v(t)\| \neq 0$, the first derivative is given by :

$$\begin{aligned} \dot{\Phi}(t) &= -\frac{\dot{m}(t)}{m(t)^2} \|\bar{p}_v(t)\| + \frac{1}{m(t)} \frac{d}{dt} (\|\bar{p}_v(t)\|) - \frac{\dot{p}_m(t)}{v_e} \\ &= \frac{T(t) \|\bar{p}_v(t)\|}{v_e m(t)^2} + \frac{1}{m(t)} \frac{\bar{p}_v(t)^t \dot{\bar{p}}_v(t)}{\|\bar{p}_v(t)\|} - \frac{T(t) \|\bar{p}_v(t)\|}{v_e m(t)^2} \\ &= -\frac{\bar{p}_v(t)^t \bar{p}_r(t)}{m(t) \|\bar{p}_v(t)\|} \end{aligned} \quad (30)$$

The sign of the derivative is the sign of the function $t \mapsto \Phi_1(t)$ defined for every t in $[t_0, t_f]$ by :

$$\Phi_1(t) = -\bar{p}_v(t)^t \bar{p}_r(t) \quad (31)$$

This function Φ_1 is differentiable almost everywhere in $[t_0, t_f]$:

$$\dot{\Phi}_1(t) = \bar{p}_r(t)^t \bar{p}_r(t) + \bar{p}_v(t)^t \frac{\partial \bar{g}(\bar{r}(t))}{\partial \bar{r}} \bar{p}_v(t) \quad (32)$$

At this stage, it is necessary either to turn to numerical methods or to make simplifying assumptions to go further in the extremal analysis. This analysis is continued in §3 in the case of a flat Earth model with a uniform gravitational field, for which $\frac{\partial \bar{g}(\bar{r}(t))}{\partial \bar{r}} = 0$.

2.7 Abnormal extremals

For abnormal extremals we can establish the following lemma.

Lemma 2. There are no coast arcs on abnormal extremals.

Proof. For an abnormal extremal, we have $p^0=0$ and from (28) :

$$\begin{cases} p_m(t_f) = 0 \\ \dot{p}_m(t) \geq 0 \text{ for } t \in [t_0, t_f] \end{cases} \Rightarrow p_m(t) \leq 0 \quad (33)$$

The switching function is then positive or null :

$$\Phi(t) = \frac{\|\bar{p}_v(t)\|}{m(t)} - \frac{p_m(t)}{v_e} \geq 0, \quad \forall t \in [t_0, t_f] \quad (34)$$

Assume by contradiction that there exists a coast arc, corresponding to a sub-interval I of $[t_0, t_f]$. Along such an arc, the switching function cannot be strictly positive from (27). This arc is therefore singular, and using (33) :

$$\Phi(t)=0 \text{ on } I \Rightarrow \frac{\|\bar{p}_v(t)\|}{m(t)} - \frac{p_m(t)}{v_e} = 0 \text{ on } I \Rightarrow \begin{cases} \bar{p}_v(t)=0 \Rightarrow \dot{\bar{p}}_v(t) = -\bar{p}_r(t)=0 \\ p_m(t)=0 \end{cases} \text{ on } I \quad (35)$$

By the Cauchy theorem, the solution of the costate equations is unique on I for given endpoint conditions. The

costate vector is then identically null on $[t_0, t_f]$: $P(t) = \begin{pmatrix} \bar{p}_r(t) \\ \bar{p}_v(t) \\ p_m(t) \end{pmatrix} = 0$, $\forall t \in [t_0, t_f]$.

We have simultaneously $\begin{cases} P(t)=0, \forall t \in [t_0, t_f] \\ p^0 = 0 \end{cases}$ which raises a contradiction with the PMP. □

Every possible abnormal extremal consists therefore of a single thrust arc. Such solutions can be found by solving the minimal time transfer problem.

2.8 Coplanar transfer problem

In practice, launcher trajectories are designed in order to bring the upper stage as close as possible to the targeted orbital plane. Plane changes at high velocities are indeed very fuel consuming. Such maneuvers should be either avoided by an adequate choice of the launch azimuth, or achieved as soon as possible during the lower stage flight.

We assume from now on that the upper stage transfer problem reduces to a coplanar transfer, which is representative of a large part of the practical applications. This problem is a 2-dimensional instance of the general transfer problem defined in §2.3.1. The state vector and command vector reduce to respectively 5 and 2 components.

Considering that the final time is fixed, the boundary value problem is a nonlinear system of dimension 5, with the initial costate $P(t_0)$ as unknowns. Despite this reduction, finding a satisfying guess for the shooting method remains an involved task.

On the other hand, it happens that the coplanar transfer problem can be nearly analytically solved in two simplified cases :

- The first simplified model is the flat Earth model. In the flat Earth model the gravity does not depend on the position and the switching function variations can be explicitly analyzed, yielding the sequences of the optimal trajectory. An automatic solution procedure can be set up that avoids the usual issues of the shooting method. This is the object of Chapter §3. A continuation procedure on the dynamics equations, presented in Chapter §4, allows passing from the flat Earth model solution to the round Earth model solution.
- The second simplified model is the impulsive model. In the impulsive model, the optimal control problem reduces to a nonlinear programming problem of small dimension that can be easily solved. The solution is

used to derive an initial guess for the costate vector. A continuation procedure on the thrust level, presented in Chapter §5, allows passing from the impulsive solution to the continuous thrust solution.

2.9 Orbital parameters

This section recalls the definition of the orbital parameters²⁵ that will be considered in the next chapters.

The keplerian motion assumes an inverse square gravity field. With that assumption the trajectory consists in a conic curve whose nature depends on the initial position and velocity. This conic is called the osculating orbit. It can be characterized by 5 constant orbital parameters defining the orbit plane (2 angles), shape (2 lengths) and orientation (1 angle), plus a varying angle along the motion.

- The perigee of the osculating orbit is the point on the orbit corresponding to the minimum radius vector.
- For a closed conic (ellipse), the apogee of the osculating orbit is the point on the orbit corresponding to the maximum radius vector.

Considering the general motion of a space vehicle submitted to various forces (drag, thrust, gravity, ...), the osculating orbit is defined at any date as the trajectory that would be followed by the vehicle from its current position and velocity if the keplerian assumption of an inverse square gravity field became valid from the current date. This osculating orbit represents a fairly good prediction of the true motion when the additional non keplerian forces (called ‘perturbations’) are small wrt the keplerian inverse square gravity.

Depending on the mission, specific properties may be required on the satellite orbit :

- Telecommunication or meteorological missions are performed from geostationary orbits (GEO). On such an orbit the satellite appears fixed to an observer from the ground. A geostationary orbit is necessarily equatorial, circular at the altitude of 35786 km, so that its period matches the Earth rotation period.
- Earth observation missions are performed from sun-synchronous orbits (SSO). On such an orbit the orbital plane keeps a constant angle with the sun direction throughout the year so that the observed Earth areas are always flown over at the same local solar hour. A sun-synchronous orbit is generally circular at altitudes between 500 km and 1500 km with an inclination around 100 deg. The inclination is tuned in order that the orbital plane precession due to the Earth flattening matches the rotation of the sun direction of about 1deg/day. This property is detailed in the Chapter §3 on the second part of this document, devoted to the collecting of space debris on sun-synchronous orbits.

3 Flat Earth model

This chapter investigates the flat Earth problem. This problem is a simplified instance of the coplanar transfer problem presented in §2.8 with the additional assumptions of a flat Earth surface and a uniform vertical gravity field. Due to these assumptions the extremal equations can be analyzed further leading to a significant problem reduction. An algorithm is then proposed to find the solution whatever the data, without prior specific initialization task. This algorithm will be used as first step of a self-starting solution procedure for the round Earth problem.

3.1 Problem statement

The Earth is modeled as a horizontal line with a uniform vertical gravity field. The inertial reference frame is (H, \vec{i}, \vec{j}) with the origin H being any point on the line representing the Earth surface and with horizontal and vertical vectors respectively \vec{i} and \vec{j} . In (H, \vec{i}, \vec{j}) (see Figure 1) :

- The position components are the downrange d and the altitude h :
$$\overrightarrow{HM} = \vec{r} = \begin{pmatrix} d \\ h \end{pmatrix}$$
- The velocity components are the horizontal velocity v_d and the vertical velocity v_h :
$$\vec{v} = \begin{pmatrix} v_d = \dot{d} \\ v_h = \dot{h} \end{pmatrix}$$

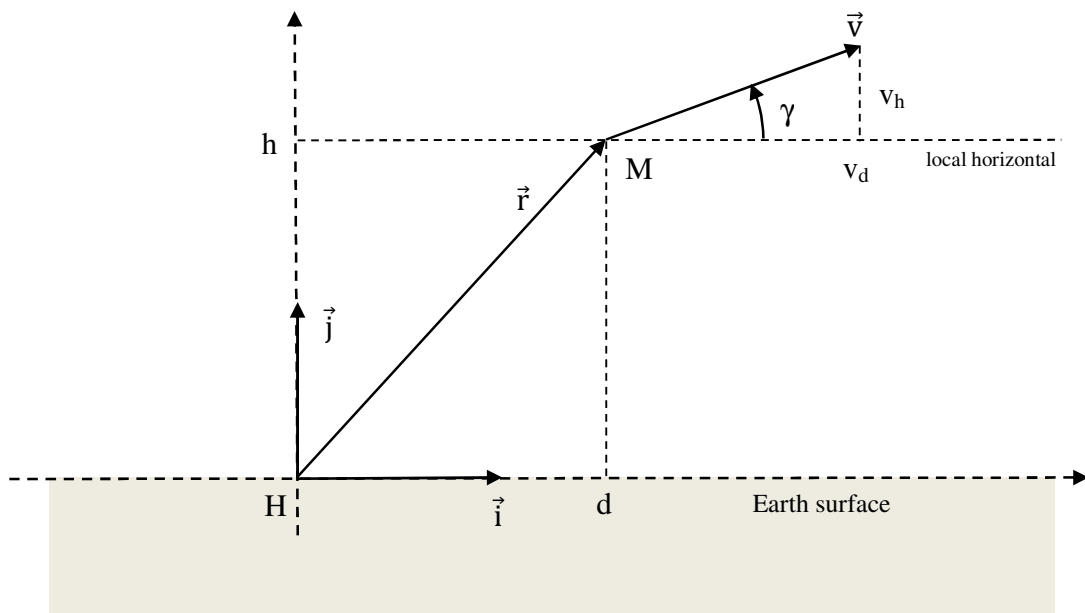


Figure 1 : Flat Earth model

The gravity field components are : $\vec{g}(\vec{r}) = \vec{g}_0 = \begin{pmatrix} 0 \\ -g_0 \end{pmatrix}$. The value $g_0 = \frac{\mu}{r_E^2}$ is the gravity at the round Earth surface,

with r_E the Earth radius, and μ the Earth gravitational constant.

The initial position $\vec{r}(t_0)$ and velocity $\vec{v}(t_0)$ are fixed. The targeted position and velocity correspond to an injection at the perigee in the round Earth model, at an altitude h_p and with a horizontal velocity v_p . The final downrange is free, corresponding to the horizontal distance travelled between the initial position and the injection. The final conditions are thus :

$$\begin{cases} \vec{r}(t_f) = \begin{pmatrix} \text{free} \\ h_p \end{pmatrix} \\ \vec{v}(t_f) = \begin{pmatrix} v_p \\ 0 \end{pmatrix} \end{cases} \quad (36)$$

Contrary to the round Earth model (§2.3.2), there exists no orbit in the flat Earth model, so that the problem has a finite time solution letting the final time free. The problem is then stated as : $\text{Max}_{U(t), t_f} J[X(t_f), t_f] = m(t_f)$

In the flat Earth model with a uniform gravity field, the free motion is a parabola. The apogee altitude h_A is reached at t_A when the vertical velocity vanishes :

$$\begin{cases} \dot{v}_h(t) = -g_0 \\ \dot{h}(t) = v_h(t) \end{cases} \Rightarrow \begin{cases} v_h(t) = v_h(t_0) - g_0 \cdot (t - t_0) \\ h(t) = h(t_0) + v_h(t_0) \cdot (t - t_0) - \frac{1}{2} g_0 \cdot (t - t_0)^2 \end{cases}$$

$$\Rightarrow \begin{cases} v_h(t_A) = 0 \text{ at } t_A = t_0 + \frac{v_h(t_0)}{g_0} > t_0 \text{ if } v_h(t_0) > 0 \\ h_A = h(t_A) = h(t_0) + \frac{v(t_0)^2}{2g_0} \end{cases}$$

For the upcoming analysis of the flat Earth problem, we make the following assumption.

Assumption 2.

We assume that the initial apogee altitude $h_A(t_0)$ is lower than the targeted injection altitude $h(t_f) = h_p$.

$$h_p > h_A(t_0) = h(t_0) + \frac{v(t_0)^2}{2g_0} \quad (37)$$

This assumption is practically always satisfied. Indeed the lower stages of a launcher must be separated on instable orbits (i.e. with a negative or low perigee altitude) in order to control their fall-down. The upper stage starts from such an orbit after the last but one stage injection. This orbit has generally a low apogee altitude and the last stage must provide a significant vertical velocity increment to reach the targeted altitude. It can be noticed that Assumption 2 discards the solutions at null thrust vertical component.

3.2 Application of the Pontryagin Maximum Principle

Applying the PMP to the flat Earth problem is identical to the general transfer problem, considering a 5-state vector and a uniform gravity $\vec{g}(\vec{r}) = \vec{g}_0$. The gravity gradient is then null : $\frac{\partial \vec{g}(\vec{r})}{\partial \vec{r}} = 0$, so that the position and velocity costate (18) can be integrated :

$$\begin{cases} \dot{\bar{p}}_r(t) = 0 \\ \dot{\bar{p}}_v(t) = -\bar{p}_r(t) \end{cases} \Rightarrow \begin{cases} \bar{p}_r(t) = \bar{p}_r(t_f) \\ \bar{p}_v(t) = \bar{p}_v(t_f) + \bar{p}_r(t_f) \cdot (t_f - t) \end{cases}, \forall t \in [t_0, t_f] \quad (38)$$

From the transversality conditions, the costate components at the endpoints are free whenever the state components are fixed, and conversely. The fixed components of the costate vector are given by :

$$\begin{cases} P(t_0) = p^0 \frac{\partial J}{\partial X} [X(t_0), t_0] \\ P(t_f) = -p^0 \frac{\partial J}{\partial X} [X(t_f), t_f] \end{cases} \quad \text{with } p^0 \leq 0 \quad (39)$$

For the flat Earth problem the transversality conditions yield :

- Initial conditions at $t=t_0$:
$$\begin{cases} d(t_0) = d_0 & \text{fixed} \\ h(t_0) = h_0 & \text{fixed} \\ v_d(t_0) = v_{d0} & \text{fixed} \\ v_h(t_0) = v_{h0} & \text{fixed} \\ m(t_0) = m_0 & \text{fixed} \end{cases} \Rightarrow \begin{cases} p_d(t_0) & \text{free} \\ p_h(t_0) & \text{free} \\ p_{vd}(t_0) & \text{free} \\ p_{vh}(t_0) & \text{free} \\ p_m(t_0) & \text{free} \end{cases} \quad (40)$$

- Final conditions at $t=t_f$:
$$\begin{cases} d(t_f) & \text{free} \\ h(t_f) = h_p & \text{fixed} \\ v_d(t_f) = v_p & \text{fixed} \\ v_h(t_f) = 0 & \text{fixed} \\ m(t_f) & \text{free} \end{cases} \Rightarrow \begin{cases} p_d(t_f) = 0 & \text{fixed} \\ p_h(t_f) & \text{free} \\ p_{vd}(t_f) & \text{free} \\ p_{vh}(t_f) & \text{free} \\ p_m(t_f) = -p^0 & \text{fixed} \end{cases} \quad (41)$$

The term “free” (or unknown) means that the corresponding state or costate component is not given a priori, but it will be determined by solving the BVP.

The position costate is thus constant with a null downrange component :

$$\bar{p}_r(t) \underset{\text{noted}}{=} \bar{p}_r = \begin{pmatrix} 0 \\ p_h \end{pmatrix}, \forall t \in [t_0, t_f] \quad (42)$$

And the velocity horizontal costate is constant :

$$p_{vd}(t) \underset{\text{noted}}{=} p_{vd}, \forall t \in [t_0, t_f] \quad (43)$$

Since the system is autonomous and the final time is free, the Hamiltonian is constant and null along the optimal trajectory :

$$H[X(t), P(t), U(t)] = \bar{p}_r^t \bar{v}(t) + \bar{p}_v(t)^t \bar{g}_0 + T(t)\Phi(t) = 0, \quad \forall t \in [t_0, t_f] \quad (44)$$

with the switching function Φ defined by (25) : $t \in [t_0, t_f] \mapsto \Phi(t) = \frac{\|\bar{p}_v(t)\|}{m(t)} - \frac{p_m(t)}{v_e}$

and the thrust magnitude determined by (27) : $\begin{cases} T(t)=0 & \text{if } \Phi(t) < 0 \\ T(t)=T_{\max} & \text{if } \Phi(t) > 0 \end{cases}$

3.3 Switching function

The derivative of the switching function has the sign of the function : $t \mapsto \Phi_1(t)$ defined for almost every $t \in [t_0, t_f]$ by :

$$\begin{aligned} \Phi_1(t) &= -\bar{p}_v(t)^t \bar{p}_r \\ &= -[\bar{p}_v(t_f) + \bar{p}_r \cdot (t_f - t)]^t \bar{p}_r \\ &= -\bar{p}_v(t_f)^t \bar{p}_r - \|\bar{p}_r\|^2 \cdot (t_f - t) \end{aligned} \quad (45)$$

This function Φ_1 is differentiable almost everywhere in $[t_0, t_f]$ and it is either constant or strictly increasing :

$$\dot{\Phi}_1(t) = \|\bar{p}_r\|^2 \geq 0 \quad (46)$$

The values of Φ_1 and Φ at the final time t_f are :

$$\begin{cases} \Phi_1(t_f) = -\bar{p}_v(t_f)^t \bar{p}_r \\ \Phi(t_f) = \frac{\|\bar{p}_v(t_f)\|}{m(t_f)} + \frac{p^0}{v_e} \end{cases} \quad \text{with } p^0 \leq 0 \quad (47)$$

3.4 Singular arcs

The two following lemmas prove that there are no singular arcs for the flat Earth problem with the Assumption 2.

Lemma 3. The function $t \mapsto \Phi(t)$ is constant on any subinterval of $[t_0, t_f]$ if and only if $\bar{p}_r = \vec{0}$.

Proof. If $\bar{p}_r = \vec{0}$ then : $\forall t \in [t_0, t_f], \Phi_1(t) = 0 \Rightarrow \forall t \in [t_0, t_f], \dot{\Phi}(t) = 0$, and Φ is constant on $[t_0, t_f]$.

Conversely if Φ is constant on a subinterval I of $[t_0, t_f]$, then :

$$\forall t \in I, \dot{\Phi}(t) = 0 \Rightarrow \forall t \in I, \Phi_1(t) = 0 \Rightarrow \forall t \in I, \dot{\Phi}_1(t) = \|\bar{p}_r\|^2 = 0 \Rightarrow \bar{p}_r = \vec{0}. \quad \square$$

Lemma 4. The function $t \mapsto \Phi(t)$ does not vanish identically on any subinterval of $[t_0, t_f]$.

Proof. The argument goes by contradiction. Assume that $\Phi(t) = 0$ on a subinterval I of $[t_0, t_f]$.

Then from Lemma 3 (necessary condition) $\bar{p}_r = \vec{0}$ and again from Lemma 3 (sufficient condition) Φ is constant on any subinterval of $[t_0, t_f]$. Therefore Φ is constant and vanishes identically on $[t_0, t_f]$.

From the costate equations (38) : $\bar{p}_r = 0 \Rightarrow \forall t \in [t_0, t_f], \bar{p}_v(t) = \bar{p}_v(t_f)$

Replacing in the Hamiltonian (44) yields :

$$\begin{aligned} H(t) &= \bar{p}_r^t \bar{v}(t) + \bar{p}_v(t)^t \bar{g}_0 + T(t)\Phi(t) = 0, \quad \forall t \in [t_0, t_f] \\ \Rightarrow \bar{p}_v(t_f)^t \bar{g}_0 &= \begin{pmatrix} p_{vd} \\ p_{vh}(t_f) \end{pmatrix}^t \begin{pmatrix} 0 \\ -g_0 \end{pmatrix} = -p_{vh}(t_f)g_0 = 0 \\ \Rightarrow p_{vh}(t_f) &= 0 \end{aligned}$$

And $\forall t \in [t_0, t_f], p_{vh}(t) = p_{vh}(t_f) = 0$ implying $\bar{u}(t) = \frac{\bar{p}_v(t)}{\|\bar{p}_v(t)\|} = \begin{pmatrix} \pm 1 \\ 0 \end{pmatrix}$ almost everywhere in $[t_0, t_f]$.

The thrust vertical component is then constantly null which is not compliant with Assumption 2. □

Lemma 4 discards singular arcs. The optimal control can thus always be determined from the PMP, using the velocity costate for the thrust direction and the switching function for the thrust magnitude.

3.5 Switching function variation

The sequences of the optimal trajectory are determined by the sign of the switching function. We analyze the switching function variation considering successively the case $\bar{p}_r = \vec{0}$ and the case $\bar{p}_r \neq \vec{0}$.

3.5.1 Null position costate

Lemma 5. If $\bar{p}_r = \vec{0}$ then the thrust is constant and maximal on $[t_0, t_f]$.

Proof. If $\bar{p}_r = \vec{0}$ then from Lemma 3, Φ is constant on $[t_0, t_f]$ and from Lemma 4, Φ is not equal to 0.

If $\Phi < 0$, the thrust is null on $[t_0, t_f]$ which is not compliant with Assumption 2.

Therefore Φ must be strictly positive and the thrust is constant equal to T_{\max} on $[t_0, t_f]$. □

In the case of a null position costate, any extremal consists therefore of a single thrust arc. The minimal consumption transfer problem is then equivalent to the minimal time transfer problem.

Furthermore the velocity costate \bar{p}_v is constant, and therefore the optimal thrust direction is constant. The problem reduces to two unknowns, namely the thrust pitch angle (angle with the horizontal) and the final time. The state equations can be analytically integrated to find the optimal solution.

3.5.2 Non null position costate

Lemma 6. If $\bar{p}_r \neq \bar{0}$ then Φ has a unique minimum at t_m . Furthermore :
$$\begin{cases} p_{vh}(t_m) = 0 \\ p_h v_h(t_m) + T(t_m)\Phi(t_m) = 0 \end{cases}$$

Proof. The sign of $\dot{\Phi}(t)$ is the sign of $\Phi_1(t) = -\bar{p}_v(t)^t \bar{p}_r = -\bar{p}_v(t_f)^t \bar{p}_r - \|\bar{p}_r\|^2 (t_f - t)$.

The function $t \mapsto \Phi_1(t)$ is linear and strictly increasing since $\bar{p}_r \neq \bar{0}$. It has a unique zero at t_m , is negative for $t < t_m$ and positive for $t > t_m$. The switching function Φ is therefore decreasing for $t < t_m$, increasing for $t > t_m$ with a unique minimum at $t = t_m$:

$$\dot{\Phi}(t_m) = 0 \Rightarrow \Phi_1(t_m) = 0 \Rightarrow \bar{p}_v(t_m)^t \bar{p}_r = \begin{pmatrix} p_{vd} \\ p_{vh}(t_m) \end{pmatrix}^t \begin{pmatrix} 0 \\ p_h \end{pmatrix} = p_{vh}(t_m)p_h = 0 \Rightarrow p_{vh}(t_m) = 0 \quad (48)$$

since by assumption $p_h \neq 0$. Replacing in the Hamiltonian (44) yields :

$$\begin{aligned} H(t_m) &= \bar{p}_r^t \bar{v}(t_m) + \bar{p}_v(t_m)^t \bar{g}_0 + T(t_m)\Phi(t_m) \\ &= \begin{pmatrix} 0 \\ p_h \end{pmatrix}^t \begin{pmatrix} v_d(t_m) \\ v_h(t_m) \end{pmatrix} + \begin{pmatrix} p_{vd} \\ p_{vh}(t_m) \end{pmatrix}^t \begin{pmatrix} 0 \\ -g_0 \end{pmatrix} + T(t_m)\Phi(t_m) \\ &= p_h v_h(t_m) + T(t_m)\Phi(t_m) \end{aligned}$$

H being constantly null, we get the required formula : $p_h v_h(t_m) + T(t_m)\Phi(t_m) = 0$ □

The variation of the switching function in the case $\bar{p}_r \neq \bar{0}$ is depicted in Table 1.

t	t_m		
$\dot{\Phi}(t)$	< 0	0	> 0
$\Phi(t)$			

Table 1 : Switching function variation in the case $p_r \neq 0$

The possible sequences of an optimal strategy when $\bar{p}_r \neq \bar{0}$ depend on :

- the sign of the minimum $\Phi(t_m)$
- the date of the minimum t_m with respect to the interval $[t_0, t_f]$

If the minimum $\Phi(t_m)$ is positive, then the switching function is always positive and the thrust is constant equal to T_{\max} on $[t_0, t_f]$. The problem is equivalent to a minimum-time problem.

The remaining case when \bar{p}_r is non null and the minimum value $\Phi(t_m)$ is negative is investigated in the next section.

For that purpose, the velocity costate is expressed as follows using $p_{vh}(t_m) = 0$ yielded by Lemma 6 :

$$\begin{aligned}
\bar{p}_v(t) &= \bar{p}_v(t_f) + \bar{p}_r \cdot (t_f - t) \\
&= \bar{p}_v(t_f) + \bar{p}_r \cdot (t_f - t_m) + \bar{p}_r \cdot (t_m - t) \\
&= \bar{p}_v(t_m) + \bar{p}_r \cdot (t_m - t) \\
&= \begin{pmatrix} p_{vd} \\ p_{vh}(t_m) \end{pmatrix} + \begin{pmatrix} 0 \\ p_h \end{pmatrix} \cdot (t_m - t) = \begin{pmatrix} p_{vd} \\ p_h(t_m - t) \end{pmatrix}
\end{aligned} \tag{49}$$

3.6 Case of a negative minimum

In this section, we assume that $\bar{p}_r \neq \vec{0}$ and that Φ has a strictly negative minimum $\Phi(t_m) < 0$.

The switching function Φ being continuous, it remains negative in a neighborhood of t_m . There exists thus a non null interval I containing t_m such that : $\forall t \in I, \Phi(t) < 0$ corresponding therefore to a coast arc : $T(t) = 0$.

Lemma 7. The switching function is symmetric with respect to t_m in the interval I . It has two zeros at distinct switching dates t_1 and t_2 symmetric with respect to t_m .

Proof. The switching function is given by (25) :

$$\Phi(t) = \frac{\|\bar{p}_v(t)\|}{m(t)} - \frac{p_m(t)}{v_e}$$

On the interval I , T is null and we have respectively for $\bar{p}_v(t)$, $p_m(t)$ and $m(t)$:

- $\bar{p}_v(t) = \begin{pmatrix} p_{vd} \\ p_h(t_m - t) \end{pmatrix}$ from (49)
- $\dot{p}_m(t) = \frac{T(t)\|\bar{p}_v(t)\|}{m(t)^2} = 0 \Rightarrow p_m(t) = p_m(t_m)$ constant
- $\dot{m}(t) = \frac{T(t)}{v_e} = 0 \Rightarrow m(t) = m(t_m)$ constant

Replacing in the switching function yields :

$$\Phi(t) = \frac{\sqrt{p_{vd}^2 + p_h^2(t_m - t)^2}}{m(t_m)} - \frac{p_m(t_m)}{v_e}, \forall t \in I. \text{ This function is symmetric with respect to } t_m \text{ (first part of the}$$

Lemma).

$$\text{On the other hand, } \Phi(t_m) < 0 \Rightarrow \frac{|p_{vd}|}{m(t_m)} - \frac{p_m(t_m)}{v_e} < 0 \Rightarrow \left(\frac{p_m(t_m)}{v_e}\right)^2 - \left(\frac{p_{vd}}{m(t_m)}\right)^2 > 0$$

and since $p_h \neq 0$ (Assumption for §3.6) Φ has two zeros at t_1 and t_2 such that :

$$(t_m - t_{1,2})^2 = \left[\left(\frac{p_m(t_m)}{v_e} \right)^2 - \left(\frac{p_{vd}}{m(t_m)} \right)^2 \right] \left(\frac{m(t_m)}{p_h} \right)^2 > 0 \quad (50)$$

These switching dates are distinct and symmetric with respect to t_m : $t_1 < t_m = \frac{t_1 + t_2}{2} < t_2$ □

The trajectory on $[t_1, t_2]$ is a parabola reaching its apogee at t_m when $v_h(t_m) = 0$:

$$\begin{cases} \dot{v}_h(t) = -g_0 \\ \dot{h}(t) = v_h(t) \end{cases} \Rightarrow \begin{cases} v_h(t) = v_h(t_m) - g_0 \cdot (t - t_m) \\ h(t) = h(t_m) + v_h(t_m) \cdot (t - t_m) - \frac{1}{2} g_0 \cdot (t - t_m)^2 \end{cases} \Rightarrow \begin{cases} v_h(t) = -g_0 \cdot (t - t_m) \\ h(t) = h(t_m) - \frac{1}{2} g_0 \cdot (t - t_m)^2 \end{cases} \quad (51)$$

In order to determine the optimal trajectory sequences, we need to locate the final date t_f with respect to the coast arc $[t_1, t_2]$.

For that purpose we analyze the variation of the vertical velocity component $v_h(t)$ in order to determine when the final condition $v_h(t_f) = 0$ can be met.

3.6.1 Vertical velocity

The function $t \mapsto v_h(t)$ is differentiable almost everywhere, except at the switching dates. Its first derivative is given by the state equation :

$$\dot{v}_h(t) = -g_0 + \frac{T(t)}{m(t)} u_h(t) \quad \text{with} \quad \begin{cases} u_h(t) = \frac{p_{vh}(t)}{\|\bar{p}_v(t)\|} = \frac{p_h(t_m - t)}{\|\bar{p}_v(t)\|} \\ \bar{p}_v(t) = \begin{pmatrix} p_{vd} \\ p_h(t_m - t) \end{pmatrix} \end{cases} \quad (52)$$

Lemma 8. The derivative of $t \mapsto v_h(t)$ is monotonous for $t > t_2$:

- decreasing if $p_h > 0$,
- increasing if $p_h < 0$.

Proof. To establish the lemma, we assess the second derivative of $v_h(t)$.

For $t > t_2$, $\bar{p}_v(t) \neq \vec{0}$ (using 49) so that the function $t \mapsto u_h(t)$ is differentiable and its derivative is :

$$\begin{aligned} \dot{u}_h(t) &= \frac{\dot{p}_{vh}(t) \|\bar{p}_v(t)\| - p_{vh}(t) \frac{d\|\bar{p}_v(t)\|}{dt}}{\|\bar{p}_v(t)\|^2} \quad \text{with} \quad \frac{d\|\bar{p}_v(t)\|}{dt} = \frac{\dot{\bar{p}}_v(t)^t \bar{p}_v(t)}{\|\bar{p}_v(t)\|} = \frac{-\bar{p}_r^t \bar{p}_v(t)}{\|\bar{p}_v(t)\|} = \frac{-p_h p_{vh}(t)}{\|\bar{p}_v(t)\|} \\ \Rightarrow \dot{u}_h(t) &= \frac{-p_h \|\bar{p}_v(t)\| + p_{vh}(t) \frac{p_h p_{vh}(t)}{\|\bar{p}_v(t)\|}}{\|\bar{p}_v(t)\|^2} = -\frac{p_h}{\|\bar{p}_v(t)\|} \left(1 - \frac{p_{vh}(t)^2}{\|\bar{p}_v(t)\|^2} \right) \end{aligned}$$

which can be expressed as :

$$\dot{u}_h(t) = -\frac{p_h}{\|\bar{p}_v(t)\|} (1 - u_h(t)^2) \quad (53)$$

On the other hand, the switching function is null at t_2 and increasing. For $t > t_2$, it is strictly positive and thrust is constant equal to T_{\max} .

Using $\dot{m}(t) = -\frac{T(t)}{v_e} = -\frac{T_{\max}}{v_e}$, the second derivative of $t \mapsto v_h(t)$ is given for $t > t_2$ by :

$$\begin{aligned} \ddot{v}_h(t) &= \frac{T_{\max}}{m(t)^2} (\dot{u}_h(t)m(t) - u_h(t)\dot{m}(t)) \\ &= \frac{T_{\max}}{m(t)^2} \left(-\frac{p_h}{\|\bar{p}_v(t)\|} (1 - u_h(t)^2) m(t) + \frac{p_h(t_m - t)}{\|\bar{p}_v(t)\|} \frac{T_{\max}}{v_e} \right) \\ &= -p_h \frac{T_{\max}}{m(t)^2 \|\bar{p}_v(t)\|} \left((1 - u_h(t)^2) m(t) + \frac{T_{\max}}{v_e} (t_m - t) \right) \end{aligned} \quad (54)$$

$$\text{with } \|\bar{u}(t)\| = 1 \Rightarrow u_h(t) \leq 1$$

For $t > t_2 > t_m$, the term inside the parenthesis are all positive. The sign of $\ddot{v}_h(t)$ is constant, opposite to the sign of p_h and the function $t \mapsto \dot{v}_h(t)$ is monotonous, decreasing if $p_h > 0$, increasing if $p_h < 0$. \square

Lemma 9. The function $t \mapsto v_h(t)$ vanishes at most once for $t > t_2$. In that case, p_h is negative.

Proof. The function $v_h(t)$ is continuous everywhere, and from (51) it is negative at t_2 : $v_h(t_2) = -g_0 \cdot (t_2 - t_m) < 0$.

We consider successively the cases $p_h > 0$ or $p_h < 0$:

- Case $p_h > 0$

In that case, $u_h(t) = \frac{p_h(t_m - t)}{\|\bar{p}_v(t)\|}$ is negative for $t > t_2 > t_m$, and $\dot{v}_h(t) = -g_0 + \frac{T(t)}{m(t)} u_h(t)$ is also negative for $t > t_2$.

$v_h(t)$ is decreasing and it can therefore not vanish for $t > t_2$.

- Case $p_h < 0$

In that case, from Lemma 8 $\dot{v}_h(t)$ is monotonous and increasing. If it becomes positive at a date $t_3 > t_2$, it remains positive for $t \geq t_3$. $v_h(t)$ is then strictly increasing for $t \geq t_3$ and it can vanish at most once at a date $t_4 \geq t_3$. \square

Lemma 9 implies that the final date t_f satisfying the condition $v_h(t_f) = 0$, can be greater than t_2 only if $p_h < 0$. In that case, $v_h(t)$ is negative on $[t_2, t_f]$ and the variations of $u_h(t)$ and $v_h(t)$ are depicted in Table 2.

t	t_1	t_m	t_2	t_3	$t_f=t_4$	
$u_h(t)$	-	-	0	+	+	+
$\dot{v}_h(t)$	-	-	-	-	0	+
$v_h(t)$	+	+	0	-	-	-

Table 2 : Vertical velocity variation in the case $p_h < 0$

We next analyze the variation of the osculating apogee $h_A(t)$ in order to prove that the case $p_h < 0$ cannot occur.

3.6.2 Osculating apogee

The osculating apogee altitude is defined at any date t as the maximal altitude reached on a free motion starting from the kinematic conditions at the date t . It is given by (37) :

$$h_A(t) = h(t) + \frac{v_h(t)^2}{2g_0} \quad (55)$$

On a downward leg, the vertical velocity is negative and the osculating apogee is located at a date prior to the current date t .

The function $t \mapsto h_A(t)$ is differentiable almost everywhere, except at the switching dates. Its first derivative is given by :

$$\dot{h}_A(t) = \dot{h}(t) + \frac{v_h(t)\dot{v}_h(t)}{g_0} = v_h(t) + \frac{v_h(t) \left[-g_0 + \frac{T(t)}{m(t)} u_h(t) \right]}{g_0} = \frac{T(t)}{m(t)} u_h(t) v_h(t) \quad (56)$$

with :

- $u_h(t) = \frac{p_h(t_m - t)}{\|\bar{p}_v(t)\|}$
- $\dot{v}_h(t) = -g_0 + \frac{T(t)}{m(t)} u_h(t)$ and $v_h(t_m) = 0$

Lemma 10. If $p_h < 0$, the final condition $h(t_f) = h_p$ cannot be met.

Proof. From Table 2, in the case $p_h < 0$, the product $u_h(t)v_h(t)$ is constantly negative or null. On the other hand, the thrust $T(t)$ is null in the interval $[t_1, t_2]$, and maximal outside, so that :

- $\dot{h}_A(t) = 0$ for $t_1 < t < t_2$
- $\dot{h}_A(t) \leq 0$ for $t < t_1$ or $t > t_2$

The osculating apogee $h_A(t)$ is decreasing with the time, and from Assumption 2 : $h_A(t_0) < h_p$.

Therefore : $\forall t \geq t_0, h_A(t) \leq h_A(t_0) < h_p$

Since $h_A(t) = h(t) + \frac{v_h(t)^2}{2g_0} \geq h(t)$, we get : $\forall t \geq t_0, h(t) < h_p$ and the final condition $h(t_f) = h_p$ is never met \square

3.7 Optimal sequences

The previous results are summed up in the following theorem.

Theorem 1. The possible sequences of the optimal trajectory are :

- Either $T_{\max} - 0$ if $t_1 < t_f < t_2$
- Or T_{\max} if $t_f < t_1$

Proof.

In the abnormal case, there is a single thrust sequence at maximal thrust (Lemma 2).

In the normal case, p_h cannot be negative (Lemma 10), else the two final conditions $\begin{cases} v_h(t_f) = 0 \\ h(t_f) = h_p \end{cases}$ cannot be met simultaneously.

If $p_h = 0$, there is a single thrust sequence at maximal thrust (Lemma 5). Furthermore the thrust direction is constant and the problem can be solved analytically.

If $p_h > 0$, the switching function has a minimum (Lemma 6) that is :

- Either positive. In that case, there is a single thrust sequence at maximal thrust.
- Or negative. In that case, $v_h(t)$ does not vanish for $t > t_2$ (Lemma 9). The final date t_f is therefore prior to t_2 and the possible sequences of the optimal trajectory are :
 - Either $T_{\max} - 0$ if $t_1 < t_f < t_2$
 - Or T_{\max} if $t_f < t_1$ \square

Remark

The conclusions of Theorem 1 might seem counterintuitive. Indeed a 2-boost strategy $T_{\max} - 0 - T_{\max}$ is generally optimal in the round Earth model. In the extreme case of impulsive maneuvers, the optimal strategy is a Hohmann-like transfer consisting in a first boost to increase the apogee altitude up to the targeted injection altitude, and after a coast arc, a second boost to increase the semi-major axis. The injection occurs at the perigee of the targeted orbit. This remains the optimal strategy as long as the thrust level is sufficiently high, and the targeted orbit is significantly higher than the initial orbit.

The flat Earth model with uniform vertical gravity does not admit any orbit like the keplerian orbits in the round Earth model. The targeted final conditions do therefore not correspond to the injection on a stable orbit in the vicinity of the perigee, but rather to a specified apogee to reach. These final conditions could be equivalently specified in term of impact downrange and velocity, as for a ballistic missile. For such ballistic problems, all the necessary maneuvers can be realized with a single initial boost. Delaying the thrust maneuvers by waiting on a coast arc is sub-optimal. The minimum consumption strategy consists in realizing the maneuver as far as possible from the final point.

Theorem 1 allows a significant reduction of the BVP for the flat Earth model. The optimal trajectory is at most composed of two sequences :

- a single thrust sequence starting at t_0 and ending when the osculating apogee h_A becomes equal to the targeted final altitude h_P ,
- possibly a coast arc until the final altitude is reached.

In the case of a final coast arc the problem can be further reduced due to the following Lemma.

Lemma 11. If there is a coast arc on the optimal trajectory, then :

- p_{vd} has the sign of $v_d(t_f) - v_d(t_0)$,
- $p_{vh}(t_0)$ is strictly positive,
- the final time is given by :

$$t_f = t_0 + \frac{p_{vh}(t_0)}{p_h} \quad (57)$$

- the switching date t_1 satisfies :

$$p_h v_h(t_1) + g_0 p_h(t_1 - t_0) = g_0 p_{vh}(t_0) \quad (58)$$

Proof. The horizontal velocity derivative is almost everywhere : $\dot{v}_d(t) = \frac{T(t)}{m(t)} u_d(t) = \frac{T(t)}{m(t)} \frac{p_{vd}}{\|\bar{p}_v(t)\|}$ where p_{vd} is

constant. It keeps the sign of p_{vd} along the whole trajectory. In order to pass from $v_d(t_0)$ to $v_d(t_f)$, p_{vd} must have the sign of $v_d(t_f) - v_d(t_0)$, which is the first result of the Lemma.

The Hamiltonian is null along the optimal trajectory. At t_f we have :

$$\begin{aligned} H(t_f) &= \bar{p}_r^t \bar{v}(t_f) + \bar{p}_v(t_f)^t \bar{g}_0 + T(t_f) \Phi(t_f) \\ &= p_h v_h(t_f) - p_{vh}(t_f) g_0 + T(t_f) \Phi(t_f) = 0 \end{aligned}$$

If there is a coast arc, the thrust is null at t_f . On the other hand : $\begin{cases} v_h(t_f) = 0 \\ p_{vh}(t_f) = p_{vh}(t_0) - p_h(t_f - t_0) \end{cases}$

Replacing in the Hamiltonian yields : $p_{vh}(t_0) = p_h(t_f - t_0)$

From Theorem 1, there can be a coast arc only if $p_h > 0$. Therefore : $p_{vh}(t_0) > 0$ and $t_f = t_0 + \frac{p_{vh}(t_0)}{p_h}$ which are

the second and third results of the Lemma.

Similarly at the switching date t_1 the Hamiltonian is null and the switching function vanishes :

$$\begin{aligned} H(t_1) &= \bar{p}_r^t \bar{v}(t_1) + \bar{p}_v(t_1)^t \bar{g}_0 + T(t_1) \Phi(t_1) \\ &= p_h v_h(t_1) - p_{vh}(t_1) g_0 \\ &= p_h v_h(t_1) - p_{vh}(t_0) g_0 + p_h(t_1 - t_0) g_0 = 0 \end{aligned}$$

which is the last result of the lemma. □

An efficient solution algorithm is derived in the next section, based on the results of Theorem 1 and Lemma 11.

3.8 Algorithmic procedure

For most practical applications a significant altitude increase is required making the constant thrust strategy inefficient. The algorithm solving the flat Earth problem is restricted to solutions such that :

Assumption 3.

We assume that the optimal trajectory comprises at least one coast arc.

This assumption discards abnormal extremals (Lemma 2), and also normal extremals with $p_h = 0$ (Lemma 10).

Since p_h is positive (Theorem 1), the solutions sought restrict to normal extremals with $p_h > 0$.

Basically the shooting method applied to the flat Earth problem consists in solving a BVP with 6 unknowns and 6 equations.

The 6 unknowns are :

- the initial costate components $(p_h, p_{vd}, p_{vh}(t_0), p_m(t_0))$ and the non positive real number p^0 . These components are defined up to a multiplicative scalar.
- the final time t_f

The 6 equations are :

- the 3 final conditions :
$$\begin{cases} h(t_f) = h_p \\ v_d(t_f) = v_p \\ v_h(t_f) = 0 \end{cases}$$
- the 2 transversality conditions :
$$\begin{cases} p_m(t_f) = -p^0 \\ H(t_f) = 0 \end{cases}$$
- a normalization condition on the costate components

For a normal extremal, p^0 is strictly negative and a usual normalization choice is to set $p^0 = -1$. However for the present problem, even if there is no abnormal extremal, a different normalization will be chosen that takes advantage of the fact that the costate vector is defined up to a multiplicative scalar (see below). The problem is reduced in 3 successive steps as presented in the following sections.

3.8.1 Final time

From lemma 11 the final time is determined directly from p_h and $p_{vh}(t_0)$ by : $t_f = t_0 + \frac{p_{vh}(t_0)}{p_h}$, replacing the transversality condition $H(t_f) = 0$. This reduces the BVP to 5 unknowns and 5 equations.

The 5 unknowns are :

- the initial costate components $(p_h, p_{vd}, p_{vh}(t_0), p_m(t_0))$ and the non positive real number p^0 . These components are defined up to a multiplicative scalar.

The 5 equations are :

- the 3 final conditions :
$$\begin{cases} h(t_f) = h_p \\ v_d(t_f) = v_p \\ v_h(t_f) = 0 \end{cases}$$
- the transversality condition : $p_m(t_f) = -p^0$
- a normalization conditions on the costate components

with the final time t_f being determined by : $t_f = t_0 + \frac{p_{vh}(t_0)}{p_h}$

3.8.2 Mass costate

In the flat Earth problem, the mass costate $p_m(t)$ is only used to assess the switching function :

$$\Phi(t) = \frac{\|\bar{p}_v(t)\|}{m(t)} - \frac{p_m(t)}{v_e} \text{ which in turn determines the switching dates.}$$

From Theorem 1, there is at most one switching date t_1 , and from Lemma 11 the value of t_1 can be determined by the condition (58) : $p_h v_h(t_1) + g_0 p_h(t_1 - t_0) = g_0 p_{vh}(t_0)$ which makes no use of p_m . The assessment of $p_m(t)$ is therefore no longer useful and it can be put aside when solving the BVP, together with the associated transversality condition $p_m(t_f) = -p^0$. The unknown p^0 that is used only in this transversality condition is also discarded of the problem. This gives a degree of freedom for the costate normalization, that will be used in §3.8.3 to further simplify the problem. The BVP is now reduced to 3 unknowns and 3 equations.

The 3 unknowns are :

- the initial costate components $(p_h, p_{vd}, p_{vh}(t_0))$ which are defined up to a multiplicative scalar.

The 3 equations are :

- the 3 final conditions :
$$\begin{cases} h(t_f) = h_p \\ v_d(t_f) = v_p \\ v_h(t_f) = 0 \end{cases}$$

with the switching time t_1 and the final time t_f being determined respectively by :

$$\begin{cases} p_h v_h(t_1) + g_0 p_h(t_1 - t_0) = g_0 p_{vh}(t_0) \\ t_f = t_0 + \frac{p_{vh}(t_0)}{p_h} \end{cases} \quad (59)$$

Furthermore these conditions (59) yielding t_1 and t_f are equivalent to the 2 conditions :

$$\begin{cases} v_h(t_f) = 0 \\ t_f = t_0 + \frac{p_{vh}(t_0)}{p_h} \end{cases} \quad (60)$$

Indeed on the coast arc from t_1 to t_f , we have :

$$\begin{aligned} v_h(t_f) &= v_h(t_1) - g_0(t_f - t_1) \\ &= v_h(t_1) + g_0(t_1 - t_0) - g_0(t_f - t_0) \\ &= v_h(t_1) + g_0(t_1 - t_0) - g_0 \frac{p_{vh}(t_0)}{p_h} \end{aligned}$$

And : $v_h(t_f) = 0 \Leftrightarrow p_h v_h(t_1) + g_0 p_h(t_1 - t_0) = g_0 p_{vh}(t_0)$

We can therefore put aside the final condition $v_h(t_f) = 0$ in the BVP. This condition is automatically satisfied when t_1 and t_f satisfy the conditions (59).

3.8.3 Costate normalization

Form Assumption 3 the costate p_h is strictly positive. A useful normalization of the costate vector consists in choosing $p_h = 1$ so that p_h is no longer an unknown of the problem. We stress that this normalization is not the usual one (that would rather consist of normalizing the costate vector so that $p^0 = -1$). The normalization chosen here happens to be more convenient and more adapted to our specific situation.

This reduces the BVP to 2 unknowns and 2 equations.

The 2 unknowns are :

- the initial costate components $(p_{vd}, p_{vh}(t_0))$.

The 2 equations are :

- the 2 final conditions :
$$\begin{cases} h(t_f) = h_p \\ v_d(t_f) = v_p \end{cases}$$

with the switching time t_1 and the final time t_f being determined by (59), with $p_h = 1$:

$$\begin{cases} v_h(t_1) + g_0(t_1 - t_0) = g_0 p_{vh}(t_0) \\ t_f = t_0 + \frac{p_{vh}(t_0)}{p_h} \end{cases}$$

3.8.4 Reduced problem solution

The reduced BVP is solved by a shooting method which consists in a numerical integration of the trajectory from guessed values of the 2 unknowns $(p_{vd}, p_{vh}(t_0))$, and a Newton-like solver to find the unknown values satisfying the final conditions. The algorithm is the following.

Input :	Gravity constant :	g ₀
	Propulsion features :	T _{max} , v _e
	Vehicle initial mass :	m(t ₀)
	Initial conditions :	t ₀ , h(t ₀), v _d (t ₀), v _h (t ₀)
	Final conditions :	h _p , v _p

1. Initialization : Choose arbitrary values of the unknowns $p_{vd} > 0$ and $p_{vh}(t_0) > 0$

2. Integrate numerically the state equations :

$$\begin{cases} \dot{h}(t) = v_h(t) \\ \dot{v}_d(t) = \frac{T_{max}}{m(t_0) - \frac{T_{max}}{v_e}(t - t_0)} \frac{p_{vd}}{\sqrt{p_{vd}^2 + [p_{vh}(t_0) - (t - t_0)]^2}} \\ \dot{v}_h(t) = \frac{T_{max}}{m(t_0) - \frac{T_{max}}{v_e}(t - t_0)} \frac{p_{vh}(t_0) - (t - t_0)}{\sqrt{p_{vd}^2 + [p_{vh}(t_0) - (t - t_0)]^2}} - g_0 \end{cases}$$

from t_0 to t_1 satisfying : $v_h(t_1) + g_0(t_1 - t_0) = g_0 p_{vh}(t_0)$

3. Compute explicitly the final state at : $t_f = t_0 + p_{vh}(t_0)$

$$\begin{cases} h(t_f) = h(t_1) + v_h(t_1)(t_f - t_1) - \frac{1}{2}g_0(t_f - t_1)^2 \\ v_d(t_f) = v_d(t_1) \end{cases}$$

4. Solve with a Newton-like method the system :

$$\mathbf{S}(p_{vd}, p_{vh}(t_0)) = \begin{pmatrix} h(t_f) - h_p \\ v_d(t_f) - v_p \end{pmatrix} = 0$$

Output : BVP unknowns p_{vd} and $p_{vh}(t_0)$

Algorithm 1 : Flat Earth problem

Instead of a numerical integration, it is possible to compute explicit expressions of the state components $h(t)$, $v_d(t)$ and $v_h(t)$ on $[t_0, t_1]$. Nevertheless the resulting expressions do not allow an explicit computation of the switching time t_1 given by the nonlinear equation $v_h(t_1) + g_0(t_1 - t_0) = g_0 p_{vh}(t_0)$. Therefore it does not improve the algorithm neither for the execution time, nor for the robustness. The numerical integration happens to be more efficient from both points of view since it allows detecting without failure the switching date t_1 during the integration.

The free Fortran routines used for the shooting method are respectively Dop853.f for the numerical integration (8th order Runge-Kutta method), and Hybrd.f for the nonlinear system solution (Newton-like method with numerical derivatives by finite differences). The same routines will be used for the successive shooting problems along the continuation procedure described in Chapter §4. Both allow controlling the accuracy of the solution. For the application example presented in Chapter §6, the numerical integration is performed with absolute accuracies respectively of 10^{-8} m on the position, and 10^{-8} m/s on the velocity. For the zeros of the shooting function the required accuracies are respectively 10^{-4} m on the final position and 10^{-4} m/s on the final velocity.

3.8.5 Solution renormalization

The above algorithm is easy to carry out and it converges in a few iterations from scratch, i.e. for almost any initial positive values of the unknowns $(p_{vd}, p_{vh}(t_0))$. This solution will be used as the starting point for the continuation procedure described in Chapter §4. For that purpose, it is necessary to retrieve the costate components of the initial problem which we denote with an upper bar : $(\bar{p}_h, \bar{p}_{vd}, \bar{p}_{vh}(t_0), \bar{p}_m(t_0))$.

The first stage consists in assessing the missing components $(p_d, p_h, p_m(t_0))$ of the initial costate from the reduced solution $(p_{vd}, p_{vh}(t_0))$ and from the switching date t_1 .

The position costates are constant : $p_d = 0$ (free coordinate) and $p_h = 1$ (normalization choice).

The velocity costate is computed explicitly from the reduced solution $(p_{vd}, p_{vh}(t_0))$:

$$\bar{p}_v(t) = \begin{pmatrix} p_{vd} \\ p_{vh}(t_0) - (t - t_0) \end{pmatrix} \Rightarrow \|\bar{p}_v(t)\| = \sqrt{p_{vd}^2 + [p_{vh}(t_0) - (t - t_0)]^2} \quad \text{for } t \in [t_0, t_f] \quad (61)$$

The value of the initial mass costate is deduced from the Hamiltonian which is constantly null :

$$\begin{aligned} H(t_0) &= \bar{p}_r^t \bar{v}(t_0) + \bar{p}_v(t_0)^t \bar{g}_0 + T(t_0) \Phi(t_0) \\ &= p_h v_h(t_0) - p_{vh}(t_0) g_0 + T_{\max} \left[\frac{\|\bar{p}_v(t_0)\|}{m(t_0)} - \frac{p_m(t_0)}{v_e} \right] = 0 \end{aligned}$$

Replacing with $\|\bar{p}_v(t_0)\| = \sqrt{p_{vd}^2 + p_{vh}(t_0)^2}$ given by (60) and $p_h = 1$, we obtain for the initial mass costate :

$$p_m(t_0) = \frac{v_e}{T_{\max}} \left[v_h(t_0) - p_{vh}(t_0)g_0 + \frac{T_{\max}}{m(t_0)} \sqrt{p_{vd}^2 + p_{vh}(t_0)^2} \right] \quad (62)$$

The second stage consists in renormalizing the costate vector components to restore the usual normalization $p^0 = -1$.

The scalar p^0 is given by the transversality condition on the mass costate : $p_m(t_f) = -p^0$.

The value of the final mass costate is deduced from switching function which vanishes at the switching time t_1 :

$$\Phi(t_1) = \frac{\|\bar{p}_v(t_1)\|}{m(t_1)} - \frac{p_m(t_1)}{v_e} = 0 \Rightarrow p_m(t_1) = v_e \frac{\|\bar{p}_v(t_1)\|}{m(t_1)} \quad (63)$$

The mass and the mass costate are constant along the coast arc from t_1 to t_f :

$$\begin{cases} m(t_f) = m(t_1) = m(t_0) - \frac{T_{\max}}{v_e}(t_1 - t_0) \\ p_m(t_f) = p_m(t_1) \end{cases} \quad (64)$$

We obtain for the final mass costate :

$$p_m(t_f) = v_e \frac{\sqrt{p_{vd}^2 + [p_{vh}(t_0) - (t_1 - t_0)]^2}}{m(t_0) - \frac{T_{\max}}{v_e}(t_1 - t_0)} \quad (65)$$

Dividing the costate components by $-p^0 = p_m(t_f)$, we obtain the solution of the initial problem with the usual normalization $p^0 = -1$:

$$\begin{cases} \bar{p}_d = 0 \\ \bar{p}_h = \frac{1}{-p^0} = \frac{m(t_0) - \frac{T_{\max}}{v_e}(t_1 - t_0)}{v_e \sqrt{p_{vd}^2 + [p_{vh}(t_0) - (t_1 - t_0)]^2}} \\ \bar{p}_{vd} = \frac{p_{vd}}{-p^0} = p_{vd} \cdot \bar{p}_h \\ \bar{p}_{vh}(t_0) = \frac{p_{vh}(t_0)}{-p^0} = p_{vh}(t_0) \cdot \bar{p}_h \\ \bar{p}_m(t_0) = \frac{p_m(t_0)}{-p^0} = \frac{v_e}{T_{\max}} \left[\bar{p}_h v_h(t_0) - \bar{p}_{vh}(t_0)g_0 + \frac{T_{\max}}{m(t_0)} \sqrt{\bar{p}_{vd}^2 + \bar{p}_{vh}^2(t_0)} \right] \end{cases} \quad (66)$$

The final time is unchanged by the renormalization : $t_f = t_0 + p_{vh}(t_0) = t_0 + \frac{\bar{p}_{vh}(t_0)}{\bar{p}_h}$

4 Continuation from the flat Earth to the round Earth model

This chapter presents the procedure to pass from the flat Earth problem whose solution is easy to the original round Earth problem which is more difficult to solve from scratch. For that purpose the coplanar transfer problem is formulated component-wise considering successively the simplified flat Earth model of Chapter §3 and the “true” round Earth model. A coordinate change is defined in order to establish some similarities between both models and to build a modified formulation of the flat Earth problem. Continuation parameters are then defined that introduce continuously the ‘round Earth terms’ into the modified formulation.

4.1 Flat Earth model

4.1.1 Coordinates and state equations

For clarity, we recall here the notations of §3.1 for the simplified flat Earth model. The position, the velocity and the

thrust are expressed in rectangular coordinates : $\vec{r} = \begin{pmatrix} d \\ h \end{pmatrix}$, $\vec{v} = \begin{pmatrix} v_d = \dot{d} \\ v_h = \dot{h} \end{pmatrix}$, $\vec{T} = T \begin{pmatrix} u_d \\ u_h \end{pmatrix}$ with : $\begin{cases} 0 \leq T \leq T_{\max} \\ \|\vec{u}\| = \sqrt{u_d^2 + u_h^2} = 1 \end{cases}$

The state vector has 5 components : $X = \begin{pmatrix} d \\ h \\ v_d \\ v_h \\ m \end{pmatrix}$ satisfying the state equations :

$$\begin{cases} \dot{d}(t) = v_d(t) \\ \dot{h}(t) = v_h(t) \\ \dot{v}_d(t) = \frac{T(t)}{m(t)} u_d(t) \\ \dot{v}_h(t) = \frac{T(t)}{m(t)} u_h(t) - g_0 \\ \dot{m}(t) = -\frac{T(t)}{v_e} \end{cases} \quad (67)$$

with the uniform vertical gravity taken at the Earth surface : $g_0 = \frac{\mu}{r_E^2}$.

The initial and final conditions are respectively :

$$\left\{ \begin{array}{l} d(t_0) = d_0 \\ h(t_0) = h_0 \\ v_d(t_0) = v_{d0} \\ v_h(t_0) = v_{h0} \\ m(t_0) = m_0 \end{array} \right. \text{ and } \left\{ \begin{array}{l} d(t_f) \text{ free} \\ h(t_f) = h_p \\ v_d(t_f) = v_p \\ v_h(t_f) = 0 \\ m(t_f) \text{ free} \end{array} \right. \quad (68)$$

These final conditions correspond to a horizontal injection at a free distance from the initial position.

4.1.2 Application of the PMP

We denote by $(\text{OCP})_{\text{flat}}$ the optimal control problem of maximizing the final mass $m(t_f)$ for the control system (67) subject to the initial and final conditions (68), with a free final time.

Denoting $\mathbf{p} = \begin{pmatrix} p_d \\ p_h \\ p_{vd} \\ p_{vh} \\ p_m \end{pmatrix}$ the costate components, the Hamiltonian of the flat Earth problem is H_{flat} :

$$\begin{aligned} H_{\text{flat}}[\mathbf{X}(t), \mathbf{P}(t), \mathbf{U}(t)] &= \mathbf{P}(t)^t \cdot \dot{\mathbf{X}}(t) = p_d(t)\dot{d}(t) + p_h(t)\dot{h}(t) + p_{vd}(t)\dot{v}_d(t) + p_{vh}(t)\dot{v}_h(t) + p_m(t)\dot{m}(t) \\ &= p_d(t)v_d(t) + p_h(t)v_h(t) \\ &\quad + p_{vd}(t) \left[\frac{T(t)u_d(t)}{m(t)} \right] + p_{vh}(t) \left[\frac{T(t)u_h(t)}{m(t)} - g_0 \right] + p_m(t) \left[-\frac{T(t)}{v_e} \right] \end{aligned} \quad (69)$$

Since the system is autonomous and the final time is free, the Hamiltonian is constant and null along any extremal.

The costate equations are :

$$\left\{ \begin{array}{l} \dot{p}_d(t) = 0 \\ \dot{p}_h(t) = 0 \\ \dot{p}_{vd}(t) = -p_d(t) \\ \dot{p}_{vh}(t) = -p_h(t) \\ \dot{p}_m(t) = \frac{T(t)}{m^2(t)} [p_{vd}(t)u_d(t) + p_{vh}(t)u_h(t)] \end{array} \right. \quad (70)$$

4.2 Round Earth model

4.2.1 Coordinates and state equations

In view of a continuation process linking the flat Earth model to the round Earth model, the polar coordinate system is the natural choice. Indeed it works with local kinematics variables (radius vector, velocity, flight path angle) that can be easily related to flat Earth rectangular coordinates (altitude, horizontal and vertical velocity).

The Earth is modeled as a circle of center O and radius r_E . The reference inertial frame (O, \vec{i}, \vec{j}) defines the trajectory plane of the coplanar transfer. The vehicle is modeled as a material point M whose polar coordinates in (O, \vec{i}, \vec{j}) are :

$$\overrightarrow{OM} = \vec{r} = \begin{pmatrix} r \\ \varphi \end{pmatrix} \text{ with } r \text{ the radius vector and } \varphi \text{ the polar angle.}$$

In the Frénet frame $(\vec{e}_r, \vec{e}_\varphi)$ defined by : $\begin{cases} \vec{e}_r = \sin\varphi \vec{i} + \cos\varphi \vec{j} \\ \vec{e}_\varphi = \cos\varphi \vec{i} - \sin\varphi \vec{j} \end{cases}$, the vehicle position and velocity are defined

$$\text{respectively by : } \begin{cases} \vec{r} = r\vec{e}_r \\ \vec{v} = \frac{d\vec{r}}{dt} = \dot{r}\vec{e}_r + r\dot{\varphi}\vec{e}_\varphi = v_r\vec{e}_r + v_\varphi\vec{e}_\varphi \end{cases}$$

The velocity components in $(\vec{e}_r, \vec{e}_\varphi)$ are :

$$\vec{v} = \begin{pmatrix} v_r = \dot{r} = v \cos \gamma \\ v_\varphi = r\dot{\varphi} = v \sin \gamma \end{pmatrix} \text{ with } v \text{ the velocity modulus and } \gamma \text{ the local slope.}$$

The coordinate system is depicted on Figure 2. The polar angle φ is oriented clock-wise in view of the upcoming coordinate change.

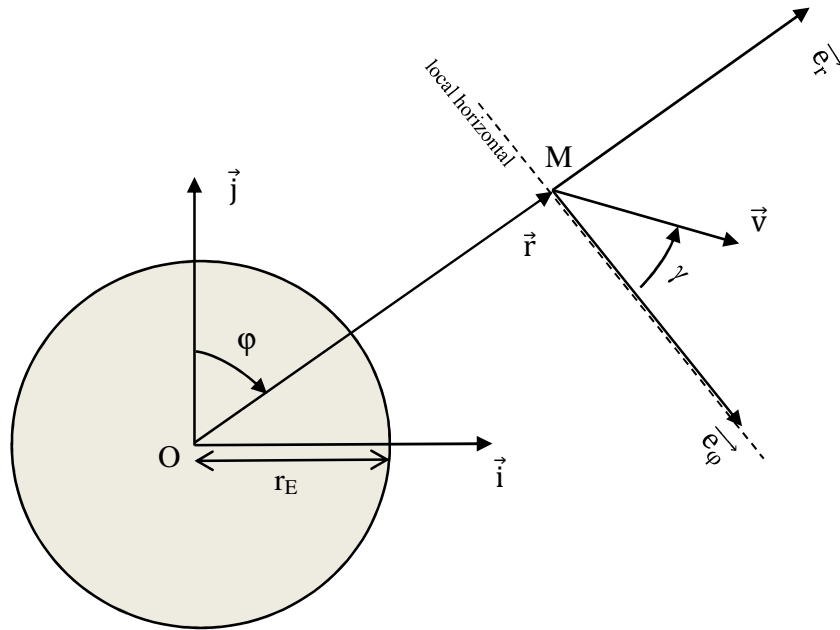


Figure 2 : Round Earth model

The Earth gravitational field is central : $\vec{g}(\vec{r}) = -g(r)\vec{e}_r$ with $g(r) = \frac{\mu}{r^2}$, with μ the Earth gravitational constant.

The thrust is expressed by: $\vec{T} = T\vec{u} = T \begin{pmatrix} u_v \\ u_\gamma \end{pmatrix}$ where T and \vec{u} represent respectively the thrust magnitude and

direction : $\begin{cases} 0 \leq T \leq T_{\max} \\ \|\vec{u}\| = \sqrt{u_v^2 + u_\gamma^2} = 1 \end{cases}$. The components u_v and u_γ are respectively along the velocity and the clock-wise normal.

The state vector has 5 components : $X(t) = \begin{pmatrix} r(t) \\ \varphi(t) \\ v(t) \\ \gamma(t) \\ m(t) \end{pmatrix}$ satisfying the state equations :

$$\begin{cases} \dot{r}(t) = v(t)\sin\gamma(t) \\ \dot{\varphi}(t) = \frac{v(t)}{r(t)}\cos\gamma(t) \\ \dot{v}(t) = -g(r(t))\sin\gamma(t) + \frac{T(t)u_v(t)}{m(t)} & \text{with } g(r) = \frac{\mu}{r^2} \\ \dot{\gamma}(t) = \left(\frac{v(t)}{r(t)} - \frac{g(r(t))}{v(t)} \right) \cos\gamma(t) + \frac{T(t)u_\gamma(t)}{m(t)v(t)} \\ \dot{m}(t) = -\frac{T(t)}{v_e} \end{cases} \quad (71)$$

The initial and final conditions are respectively :

$$\begin{cases} r(t_0) = r_0 \\ \varphi(t_0) = \varphi_0 \\ v(t_0) = v_0 \\ \gamma(t_0) = \gamma_0 \\ m(t_0) = m_0 \end{cases} \quad \text{and} \quad \begin{cases} r(t_f) = r_p \\ \varphi(t_f) \text{ free} \\ v(t_f) = v_p \\ \gamma(t_f) = 0 \\ m(t_f) \text{ free} \end{cases} \quad (72)$$

These final conditions correspond to an injection at the perigee of the targeted orbit (null flight path angle) at a free angular distance from the initial position. This means that the orientation of the final orbit is not prescribed.

The final conditions can also be defined in terms of orbit energy K_f and eccentricity e_f . Using the relationships (15) in the 2-dimensional case yields as final constraints :

$$\begin{cases} \frac{v(t_f)^2}{2} - \frac{\mu}{r(t_f)} = K_f \\ \sin^2\gamma(t_f) + \left(1 - \frac{r(t_f)v(t_f)^2}{\mu} \right)^2 \cos^2\gamma(t_f) = e_f^2 \end{cases} \quad (73)$$

In that case, the injection position on the orbit is not prescribed, nor the orbit orientation. In view of the continuation procedure between the flat and the round Earth model, we consider rather final conditions under the form (72) that apply directly on the state vector components. As mentioned in §2.3.3, the resulting BVP is reduced and thus it is easier to handle by the shooting method. Nevertheless this leads to over-constrain the final state so that an additional optimization of the free injection conditions must be processed together with the BVP solution. This is discussed further in §4.8.

4.2.2 Application of the PMP

We denote by $(\text{OCP})_{\text{round}}$ the optimal control problem of maximizing the final mass $m(t_f)$ for the control system (71) subject to the initial and final conditions (72), with a fixed final time. Indeed as explained in §2.3.2, the problem has no finite solution at free final time. The way to choose the final time value will be explained in the continuation procedure.

Denoting $\mathbf{p} = \begin{pmatrix} p_r \\ p_\varphi \\ p_v \\ p_\gamma \\ p_m \end{pmatrix}$ the costate components, the Hamiltonian for the round Earth problem is H_{round} :

$$\begin{aligned}
H_{\text{round}}[\mathbf{X}(t), \mathbf{P}(t), \mathbf{U}(t)] &= \mathbf{P}(t)^t \cdot \dot{\mathbf{X}}(t) = p_r(t)\dot{r}(t) + p_\varphi(t)\dot{\varphi}(t) + p_v(t)\dot{v}(t) + p_\gamma(t)\dot{\gamma}(t) + p_m(t)\dot{m}(t) \\
&= p_r(t)[v(t)\sin\gamma(t)] + p_\varphi(t)\left[\frac{v(t)\cos\gamma(t)}{r(t)}\right] \\
&\quad + p_v(t)\left[-g(r(t))\sin\gamma(t) + \frac{T(t)u_v(t)}{m(t)}\right] + p_\gamma(t)\left[\left(\frac{v(t)}{r(t)} - \frac{g(r(t))}{v(t)}\right)\cos\gamma(t) + \frac{T(t)u_\gamma(t)}{m(t)v(t)}\right] \\
&\quad + p_m(t)\left[-\frac{T(t)}{v_e}\right]
\end{aligned} \tag{74}$$

Since the system is autonomous, the Hamiltonian is constant along any extremal. Contrary to the flat Earth model, its value is not necessarily null since the final time is fixed.

With the derivative of the Earth central gravitational field $g(r) = \frac{\mu}{r^2}$ given by : $\frac{dg(r)}{dr} = -\frac{2\mu}{r^3} = -\frac{2g(r)}{r}$, the costate equations are :

$$\begin{cases} \dot{p}_r(t) = p_\phi(t) \frac{v(t)\cos\gamma(t)}{r(t)^2} - p_v(t) \frac{2g(r(t))\sin\gamma(t)}{r(t)} + p_\gamma(t) \left[\frac{v(t)}{r(t)} - \frac{2g(r(t))}{v(t)} \right] \frac{\cos\gamma(t)}{r(t)} \\ \dot{p}_\phi(t) = 0 \\ \dot{p}_v(t) = -p_r(t)\sin\gamma(t) - p_\phi(t) \frac{\cos\gamma(t)}{r(t)} - p_\gamma(t) \left[\left(\frac{1}{r(t)} + \frac{g(r(t))}{v(t)^2} \right) \cos\gamma(t) - \frac{T(t)u_\gamma(t)}{m(t)v(t)^2} \right] \\ \dot{p}_\gamma(t) = -p_r(t)v(t)\cos\gamma(t) + p_\phi(t) \frac{v(t)\sin\gamma(t)}{r(t)} + p_v(t)g(r(t))\cos\gamma(t) + p_\gamma(t) \left[\frac{v(t)}{r(t)} - \frac{g(r(t))}{v(t)} \right] \sin\gamma(t) \\ \dot{p}_m(t) = p_v(t) \frac{T(t)u_v(t)}{m(t)^2} + p_\gamma(t) \frac{T(t)u_\gamma(t)}{m(t)^2 v(t)} \end{cases} \quad (75)$$

4.3 Modified flat Earth model

4.3.1 Flat vs round Earth problem

Making the shooting method converge is practically difficult on the round Earth problem $(\mathbf{OCP})_{\text{round}}$, because it requires very accurate guesses for the costate initialization and for the control discontinuities. On the other hand, the solution of the flat Earth problem $(\mathbf{OCP})_{\text{flat}}$ is easily obtained by the Algorithm 1 presented in §3.8.4. A natural idea to bypass the initialization issues of $(\mathbf{OCP})_{\text{round}}$ is to retrieve the costate solution of $(\mathbf{OCP})_{\text{flat}}$ and use it as initial guess for the shooting method.

This idea does not work so directly, because the coordinate systems are different, and also because the features of the two problems are too different. In order to make the transition possible, a continuation process must be set up by introducing parameters in the dynamic model. The goal is to pass progressively from the flat Earth problem to the initial round Earth problem by a continuous variation of these parameters. The continuation process must take into account the respective features of the flat and round Earth models we would like to link :

- The coordinate system is rectangular for the flat Earth model, polar for the round Earth model. A coordinate change is therefore necessary on the round Earth model to make some similarity appear in the state equations.
- The gravity is vertical and constant g_0 for the flat Earth model, central and variable $g(r)$ for the round Earth model. The continuation process should change continuously the gravity from g_0 to $g(r)$.
- The round Earth model admits orbits (i.e. stable periodic trajectories with null thrust), namely the Keplerian orbits, whereas no orbits exist in the flat Earth model. Indeed the vertical velocity is always decreasing in the absence of thrust, due to the constant downward gravity and the fact that the altitude decrease is not compensated by the Earth curvature.

This last difference between the two models introduces major changes in the solution of the optimal control problem. In order to overcome this issue and make a continuation possible, the dynamics of the flat Earth model should be modified by additional terms, so that horizontal trajectories ('orbits') with zero control may exist. At the same time, this modified model should be equivalent, up to some change of coordinates, to the round Earth model. This modified flat Earth model is derived in the next section, by defining a change of coordinates that flattens circular orbits into horizontal trajectories, and then by computing the associated control system.

4.3.2 Coordinate change

In order to establish some similarities in the state equations of the round Earth model and the flat Earth model, a coordinate change is defined that maps the round Earth variables to "equivalent flat Earth variables". The new rectangular coordinates (d, h, v_d, v_h) are defined from the polar coordinates (r, φ, v, γ) as follows, with the goal of mapping circular orbits to horizontal trajectories :

- The downrange d is defined by the curvilinear distance at the radius r : $d = r\varphi$
- The altitude is defined by the distance to the Earth surface : $h = r - r_E$
- The horizontal and vertical velocity components are defined from the modulus and flight path angle :

$$\begin{cases} v_d = v \cos \gamma \\ v_h = v \sin \gamma \end{cases}$$

The coordinate change is depicted on Figure 3.

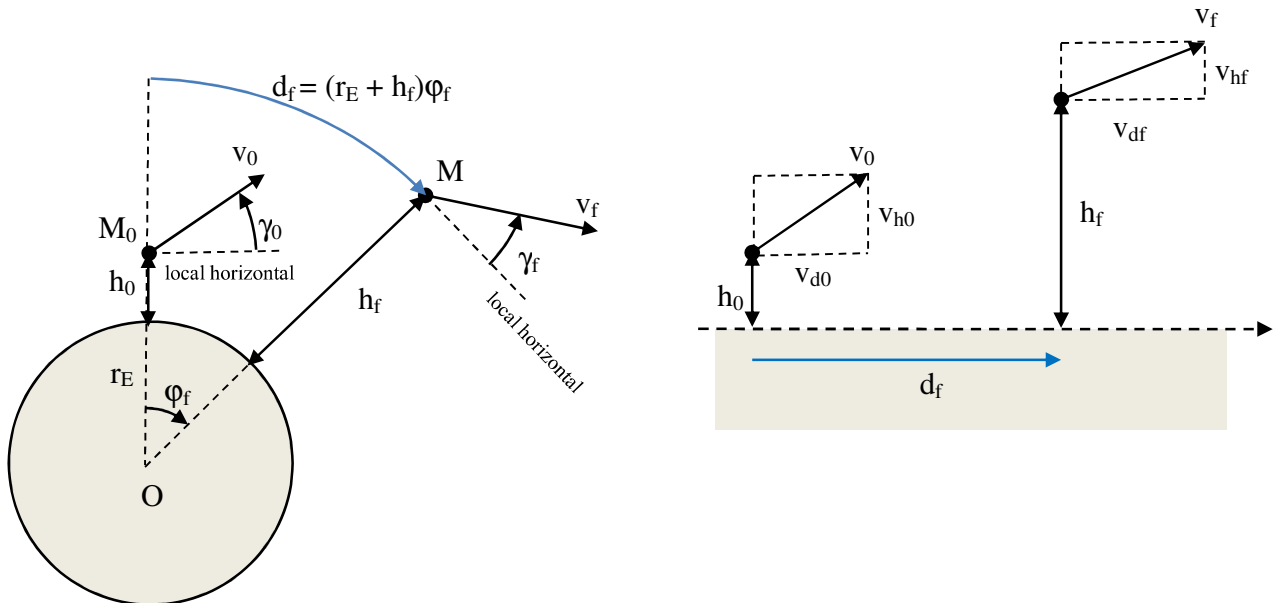


Figure 3 : Coordinate change from the round Earth to the flat Earth model

Summing up we consider the change of coordinates for the round Earth problem :

$$\begin{cases} d = r\varphi \\ h = r - r_E \\ v_d = v\cos\gamma \\ v_h = v\sin\gamma \end{cases} \Leftrightarrow \begin{cases} r = r_E + h \\ \varphi = \frac{d}{r_E + h} \\ v = \sqrt{v_d^2 + v_h^2} \\ \gamma = \text{Arctan} \frac{v_h}{v_d} \end{cases} \quad (76)$$

The coordinate change (76) defines a diffeomorphism denoted $F : F(d, h, v_d, v_h) = (r, \varphi, v, \gamma)$.

For the thrust direction $\bar{u}(t)$, the transformation from polar to rectangular coordinates is :

$$\begin{pmatrix} u_d \\ u_h \end{pmatrix} = \begin{pmatrix} \cos\gamma & -\sin\gamma \\ \sin\gamma & \cos\gamma \end{pmatrix} \begin{pmatrix} u_v \\ u_\gamma \end{pmatrix} \quad (77)$$

4.3.3 State equations

In order to write the state equations with the new coordinates, we first differentiate the new coordinates (76) :

$$\begin{cases} \dot{d}(t) = \dot{r}(t)\varphi(t) + r(t)\dot{\varphi}(t) \\ \dot{h}(t) = \dot{r}(t) \\ \dot{v}_d(t) = \dot{v}(t)\cos\gamma(t) - v(t)\dot{\gamma}(t)\sin\gamma(t) \\ \dot{v}_h(t) = \dot{v}(t)\sin\gamma(t) + v(t)\dot{\gamma}(t)\cos\gamma(t) \end{cases}$$

The polar coordinates derivatives are then replaced from the state equations (71) :

$$\Rightarrow \begin{cases} \dot{d}(t) = \varphi(t)v(t)\sin\gamma(t) + r(t)\frac{v(t)}{r(t)}\cos\gamma(t) \\ \dot{h}(t) = v(t)\sin\gamma(t) \\ \dot{v}_d(t) = \left[-g(r(t))\sin\gamma(t) + \frac{T(t)u_v(t)}{m(t)} \right] \cos\gamma(t) - \left[\left(\frac{v(t)}{r(t)} - \frac{g(r(t))}{v(t)} \right) \cos\gamma(t) + \frac{T(t)u_\gamma(t)}{m(t)v(t)} \right] v(t)\sin\gamma(t) \\ \dot{v}_h(t) = \left[-g(r(t))\sin\gamma(t) + \frac{T(t)u_v(t)}{m(t)} \right] \sin\gamma(t) + \left[\left(\frac{v(t)}{r(t)} - \frac{g(r(t))}{v(t)} \right) \cos\gamma(t) + \frac{T(t)u_\gamma(t)}{m(t)v(t)} \right] v(t)\cos\gamma(t) \end{cases}$$

Simplifying and reordering the right hand sides yields :

$$\Rightarrow \begin{cases} \dot{d}(t) = v(t)\cos\gamma(t) + \varphi(t)v(t)\sin\gamma(t) \\ \dot{h}(t) = v(t)\sin\gamma(t) \\ \dot{v}_d(t) = \frac{T(t)}{m(t)} [u_v(t)\cos\gamma(t) - u_\gamma(t)\sin\gamma(t)] - \frac{v(t)^2}{r(t)} \cos\gamma(t)\sin\gamma(t) \\ \dot{v}_h(t) = \frac{T(t)}{m(t)} [u_v(t)\sin\gamma(t) + u_\gamma(t)\cos\gamma(t)] + \frac{v(t)^2}{r(t)} \cos^2\gamma(t) - g(r(t)) \end{cases}$$

Replacing in the right hand sides by the new coordinates (76), and retrieving the mass equation unchanged, we obtain the state equations in the new coordinate system :

$$\begin{cases} \dot{d}(t) = v_d(t) + \frac{d(t)v_h(t)}{r_E + h(t)} \\ \dot{h}(t) = v_h(t) \\ \dot{v}_d(t) = \frac{T(t)}{m(t)} u_d(t) - \frac{v_d(t)v_h(t)}{r_E + h(t)} \\ \dot{v}_h(t) = \frac{T(t)}{m(t)} u_h(t) + \frac{v_d(t)^2}{r_E + h(t)} - g(r_E + h(t)) \\ \dot{m}(t) = -\frac{T(t)}{v_e} \end{cases} \quad (78)$$

We call ‘modified flat Earth model’ the formulation (78) in rectangular coordinates. It still represents the original round Earth model, up to the change of coordinates. In particular, the centrifugal term $\frac{v_d^2}{r}$ compensates the gravity in the vertical velocity equation and makes null thrust orbits possible. For example, the following initial conditions yield a circular orbit at the altitude h_0 in a central gravitational field $g(r) = \frac{\mu}{r^2}$:

$$\begin{cases} h(t_0) = h_0 = r_0 - r_E \\ v_h(t_0) = 0 \\ v_d(t_0) = r_0 g(r_0) \end{cases} \Rightarrow \begin{cases} \dot{h}(t_0) = 0 \\ \dot{v}_h(t_0) = 0 \\ \dot{v}_d(t_0) = 0 \end{cases} \quad \text{and thus :} \quad \begin{cases} h(t) = h(t_0) \\ v_h(t) = 0 \\ v_d(t) = v_d(t_0) \end{cases}, \forall t$$

For convenience, the flat Earth model formulation (67) is reproduced here below in order to compare it with the modified flat Earth formulation :

$$\begin{cases} \dot{d}(t) = v_d(t) \\ \dot{h}(t) = v_h(t) \\ \dot{v}_d(t) = \frac{T(t)}{m(t)} u_d(t) \\ \dot{v}_h(t) = \frac{T(t)}{m(t)} u_h(t) - g_0 \\ \dot{m}(t) = -\frac{T(t)}{v_e} \end{cases} \quad (79)$$

We observe two differences between (78) and (79) :

- the first one is the gravity term in the dynamics of v_h , constant in the flat Earth model, varying with the radius vector in the modified model.
- the second one is the presence of additional terms in the dynamics of d , v_d and v_h . These new terms introduce a coupling between the horizontal and vertical motion, which does not exist in the flat Earth model. They can be considered as corrective terms, making possible in particular the existence of orbits (i.e. periodic trajectories without thrust).

The modified model is equivalent to the round Earth model while establishing an interesting similarity with the flat Earth model. This similarity is used in the next section to set up a continuation procedure between the flat Earth problem and the round Earth problem.

4.4 Continuation procedure

4.4.1 Continuation parameters

To pass from the simplified flat Earth model (79) to the modified flat Earth model (78), we introduce two parameters denoted respectively λ_1 and λ_2 . The first parameter λ_1 permits to pass continuously from the constant gravity to the varying gravity, and the second parameter λ_2 introduces continuously the corrective terms in the dynamics. This defines a family of control systems with the parameters λ_1 and λ_2 varying between 0 and 1 :

$$\left\{ \begin{array}{l} \dot{d}(t) = v_d(t) + \lambda_2 \frac{d(t)v_h(t)}{r_E + h(t)} \\ \dot{h}(t) = v_h(t) \\ \dot{v}_d(t) = \frac{T(t)}{m(t)} u_d(t) - \lambda_2 \frac{v_d(t)v_h(t)}{r_E + h(t)} \\ \dot{v}_h(t) = \frac{T(t)}{m(t)} u_h(t) + \lambda_2 \frac{v_d(t)^2}{r_E + h(t)} - g(r_E + \lambda_1 h(t)) \quad \text{with} \quad g(r) = \frac{\mu}{r^2} \\ \dot{m}(t) = -\frac{T(t)}{v_e} \end{array} \right. \quad (80)$$

The system initial and final conditions are those of the flat Earth problem (68) :

$$\left\{ \begin{array}{l} d(t_0) = d_0 \\ h(t_0) = h_0 \\ v_d(t_0) = v_{d0} \\ v_h(t_0) = v_{h0} \\ m(t_0) = m_0 \end{array} \right. \text{ and } \left\{ \begin{array}{l} d(t_f) \text{ free} \\ h(t_f) = h_p \\ v_d(t_f) = v_p \\ v_h(t_f) = 0 \\ m(t_f) \text{ free} \end{array} \right. \quad (81)$$

4.4.2 Continuation path

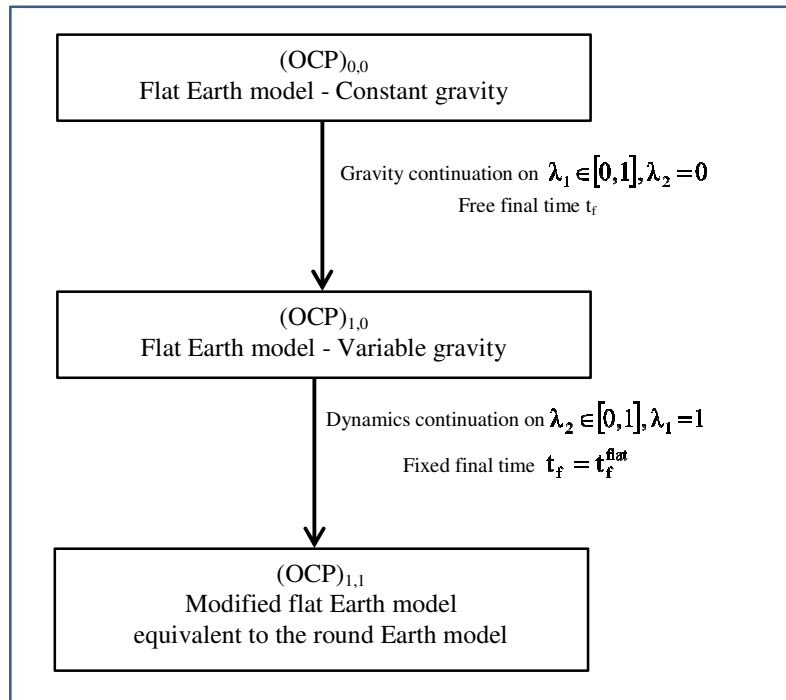
For all $(\lambda_1, \lambda_2) \in [0, 1]^2$, we denote by $(\text{OCP})_{\lambda_1, \lambda_2}$ the optimal control problem of maximizing the final mass $m(t_f)$ for the control system (68) subject to the initial and final conditions (59). The problem is at free final time for $\lambda_2 = 0$, and fixed final time $\lambda_2 \neq 0$ as explained hereafter.

For $\lambda_1 = \lambda_2 = 0$, we recover the simplified flat Earth problem $(\text{OCP})_{\text{flat}}$, and for $\lambda_1 = \lambda_2 = 1$, we recover the modified flat Earth problem which is diffeomorphic to the initial round Earth problem $(\text{OCP})_{\text{round}}$. The continuation from the flat Earth problem $(\text{OCP})_{0,0}$ to the modified flat Earth problem $(\text{OCP})_{1,1}$ proceeds in two steps :

- A first continuation (which we call the ‘gravity continuation’) is realized on the parameter λ_1 from 0 to 1, while fixing the parameter $\lambda_2 = 0$. This introduces the variable gravity in the modified model. During this continuation, the final time is free since the corresponding optimal control problems $(\text{OCP})_{\lambda_1,0}$ have a finite solution at free final time. We denote t_f^{flat} the optimal value of the final time obtained as solution of $(\text{OCP})_{1,0}$.
- A second continuation (which we call the ‘dynamics continuation’) is realized on the parameter λ_2 from 0 to 1, while fixing the parameter $\lambda_1 = 1$. This introduces the additional dynamics terms in the modified model. During this continuation, the final time is fixed to the value t_f^{flat} obtained as solution of $(\text{OCP})_{1,0}$. Indeed the targeted problem $(\text{OCP})_{1,1}$ which is equivalent to the round Earth problem, has no finite solution at free final time, making necessary to fix this value.

The choice of fixing $t_f = t_f^{\text{flat}}$ is somewhat arbitrary. The actual constraint in a practical application is rather expressed as a maximum number of thrust arcs (limited by the engine re-ignition capabilities) or a maximum number of revolutions (limited by the operational tracking capabilities). The solution obtained for $(\text{OCP})_{1,1}$ with the fixed final time $t_f = t_f^{\text{flat}}$ can be improved by a third continuation consisting in progressively increasing the value of t_f , as long as the final mass increases, and until an operational constraint (number of ignitions, or number of revolutions) is violated. This is illustrated on the application example in Chapter §6.

The continuation procedure is depicted hereafter.



Algorithm 2 : Continuation procedure from (OCP)_{0,0} to (OCP)_{1,1}

The continuation consists in solving a series of shooting problems for neighboring optimal control problems of the family $(\text{OCP})_{\lambda_1, \lambda_2}$. Each problem differs from the previous one by a small increment on the parameter λ_1 for the first continuation, or the parameter λ_2 for the second continuation. Every time a problem has been solved, its solution (i.e. the initial costate components, and the final time) is retrieved as initial guess for the next shooting problem. If the increment on the continuation parameter is sufficiently small, one can hope that this initial guess allows the convergence of the Newton method.

Different continuation algorithms exist, differing by the way the continuation parameter is updated from one shooting problem to the other (Refs^{20,21}). For the coplanar transfer problem, we will use the simplest method which is a discrete homotopy with constant increments.

The BVP associated to the problem $(\text{OCP})_{\lambda_1, \lambda_2}$ is derived in the next section by applying the PMP, in a very similar way as done for the general 3 dimensional case (Chapter §2).

4.5 Application of the PMP

4.5.1 Hamiltonian and costate equations

Denoting $\mathbf{p} = \begin{pmatrix} p_d \\ p_h \\ p_{vd} \\ p_{vh} \\ p_m \end{pmatrix}$ the costate components, the Hamiltonian of the modified flat Earth problem (80) is H_{λ_1, λ_2} :

The Hamiltonian of the modified flat Earth problem (80) is :

$$\begin{aligned}
 H_{\lambda_1, \lambda_2}[\mathbf{X}(t), \mathbf{P}(t), \mathbf{U}(t)] &= \mathbf{P}(t)' \cdot \dot{\mathbf{X}}(t) = p_d(t)\dot{d}(t) + p_h(t)\dot{h}(t) + p_{vd}(t)\dot{v}_d(t) + p_{vh}(t)\dot{v}_h(t) + p_m(t)\dot{m}(t) \\
 &= p_d(t) \left[v_d(t) + \lambda_2 \frac{d(t)v_h(t)}{r_E + h(t)} \right] + p_h(t)v_h(t) \\
 &\quad + p_{vd}(t) \left[\frac{T(t)}{m(t)} u_d(t) - \lambda_2 \frac{v_d(t)v_h(t)}{r_E + h(t)} \right] \\
 &\quad + p_{vh}(t) \left[\frac{T(t)}{m(t)} u_h(t) + \lambda_2 \frac{v_d(t)^2}{r_E + h(t)} - g(r_E + \lambda_1 h(t)) \right] \\
 &\quad + p_m(t) \left[-\frac{T(t)}{v_e} \right]
 \end{aligned} \tag{82}$$

The costate equations are :

$$\left\{ \begin{aligned}
 \dot{p}_d(t) &= -p_d(t) \frac{\lambda_2 v_h(t)}{r_E + h(t)} \\
 \dot{p}_h(t) &= p_d(t) \frac{\lambda_2 d(t)v_h(t)}{(r_E + h(t))^2} - p_{vd}(t) \frac{\lambda_2 v_d(t)v_h(t)}{(r_E + h(t))^2} + p_{vh}(t) \left[\frac{\lambda_2 v_d(t)^2}{(r_E + h(t))^2} + \lambda_1 \frac{\partial g}{\partial r}(r_E + \lambda_1 h(t)) \right] \\
 \dot{p}_{vd}(t) &= -p_d(t) + p_{vd}(t) \frac{\lambda_2 v_h(t)}{r_E + h(t)} - p_{vh}(t) \frac{2\lambda_2 v_d(t)}{r_E + h(t)} \\
 \dot{p}_{vh}(t) &= -p_d(t) \frac{\lambda_2 d(t)}{r_E + h(t)} - p_h(t) + p_{vd}(t) \frac{\lambda_2 v_d(t)}{r_E + h(t)} \\
 \dot{p}_m(t) &= \frac{T(t)}{m(t)^2} [p_{vd}(t)u_d(t) + p_{vh}(t)u_h(t)]
 \end{aligned} \right. \tag{83}$$

$$\text{with } g(r) = \frac{\mu}{r^2} \text{ and } \frac{dg(r)}{dr} = -\frac{2\mu}{r^3} = -\frac{2g(r)}{r}.$$

4.5.2 Control

Denoting the position costate, the velocity costate and the gravity by vectors respectively :

$$\bar{p}_r = \begin{pmatrix} p_d \\ p_h \end{pmatrix}, \bar{p}_v = \begin{pmatrix} p_{vd} \\ p_{vh} \end{pmatrix}, \bar{g} = \begin{pmatrix} 0 \\ -g(r_E + \lambda_1 h(t)) \end{pmatrix}, \text{ the Hamiltonian can be expressed as :}$$

$$\begin{aligned} H_{\lambda_1, \lambda_2} [X(t), P(t), U(t)] = & \bar{p}_r(t)^t \bar{v}(t) + \bar{p}_v(t)^t \bar{g}(r_E + \lambda_1 h(t)) + T(t) \left[\frac{\bar{p}_v(t)^t \bar{u}(t)}{m(t)} - \frac{p_m(t)}{v_e} \right] \\ & + \frac{\lambda_2}{r_E + h(t)} [p_d(t)d(t)v_h(t) - p_{vd}(t)v_d(t)v_h(t) + p_{vh}(t)v_d(t)^2] \end{aligned} \quad (84)$$

The expression of H_{λ_1, λ_2} is similar to the Hamiltonian of the general 3 dimensional case (17), with the additive terms introduced in the modified flat Earth model. These purely additive terms do not depend on the control, so that the results obtained for the general 3 dimensional problem in §2.5 and §2.6 apply to the modified flat Earth problem without change. For completeness, we reproduce hereafter the calculus of the optimal control obtained derived from the PMP.

The Hamiltonian maximization yields firstly the optimal thrust direction along the velocity costate :

$$\begin{aligned} \text{Max}_{U(t)} H_{\lambda_1, \lambda_2} [X(t), P(t), U(t)] & \Leftrightarrow \text{Max}_{T(t), \bar{u}(t)} T(t) \left[\frac{\bar{p}_v(t)^t \bar{u}(t)}{m(t)} - \frac{p_m(t)}{v_e} \right] \text{ s.t. } \begin{cases} 0 \leq T(t) \leq T_{\max} \\ \|\bar{u}(t)\| = 1 \end{cases} \\ & \Rightarrow \bar{u}(t) = \frac{\bar{p}_v(t)}{\|\bar{p}_v(t)\|} \text{ if } T(t) > 0 \text{ and } \|\bar{p}_v(t)\| \neq 0 \end{aligned} \quad (85)$$

For a null thrust the Hamiltonian is insensitive to the thrust direction, and from Lemma 1, the function $t \mapsto \|\bar{p}_v(t)\|$ does not vanish identically on any subinterval of $[t_0, t_f]$. The thrust direction \bar{u} is thus well defined almost everywhere by (85).

Replacing the optimal thrust direction \bar{u} in the Hamiltonian and defining the switching function $\Phi_{\lambda_1, \lambda_2}$ as :

$$\Phi_{\lambda_1, \lambda_2}(t) = \frac{\|\bar{p}_v(t)\|}{m(t)} - \frac{p_m(t)}{v_e} \quad \text{for } t \in [t_0, t_f] \quad (86)$$

the Hamiltonian can be expressed as :

$$\begin{aligned} H_{\lambda_1, \lambda_2} [X(t), P(t), U(t)] = & \bar{p}_r(t)^t \bar{v}(t) + \bar{p}_v(t)^t \bar{g}(r_E + \lambda_1 h(t)) + T(t) \Phi_{\lambda_1, \lambda_2}(t) \\ & + \frac{\lambda_2}{r_E + h(t)} [p_d(t)d(t)v_h(t) - p_{vd}(t)v_d(t)v_h(t) + p_{vh}(t)v_d(t)^2] \end{aligned} \quad (87)$$

The Hamiltonian maximization requires secondly choosing the thrust magnitude so that :

$$\begin{cases} T(t)=0 & \text{if } \Phi_{\lambda_1, \lambda_2}(t) < 0 \\ T(t)=T_{\max} & \text{if } \Phi_{\lambda_1, \lambda_2}(t) > 0 \end{cases} \quad (88)$$

4.5.3 Transversality conditions

The final constraints (81) yield for the position and velocity transversality conditions :

$$\begin{cases} d(t_f) & \text{free} \\ h(t_f) = h_p & \text{fixed} \\ v_d(t_f) = v_p & \text{fixed} \\ v_h(t_f) = 0 & \text{fixed} \end{cases} \Rightarrow \begin{cases} p_d(t_f) = 0 & \text{fixed} \\ p_h(t_f) & \text{free} \\ p_{vd}(t_f) & \text{free} \\ p_{vh}(t_f) & \text{free} \end{cases} \quad (89)$$

With the usual normalization of the state vector components with $p^0 = -1$ the transversality condition for the mass costate is :

$$p_m(t_f) = 1 \quad (90)$$

The system being autonomous, the Hamiltonian is constant and if the final time is free, the Hamiltonian is null along any extremal :

$$H_{\lambda_1, \lambda_2}(t_f) = 0 \quad (91)$$

4.5.4 Boundary value problem (BVP) and shooting method

The BVP derived from the PMP is of dimension 6. It consists in finding a zero of the shooting function $S_{\lambda_1, \lambda_2} : \mathbb{R}^6 \rightarrow \mathbb{R}^6$ is defined as :

$$S_{\lambda_1, \lambda_2} [P(t_0), t_f] = \begin{pmatrix} h(t_f) - h_p \\ v_d(t_f) - v_p \\ v_h(t_f) \\ p_d(t_f) \\ p_m(t_f) - 1 \\ H(t_f) \end{pmatrix} \quad (92)$$

If the final time t_f is fixed, the last equation is discarded and the shooting function is reduced to 5 components. Computing the shooting function values requires integrating the state and costate equations starting from the fixed initial state $X(t_0)$ and the guessed initial costate $P(t_0)$. The control along the trajectory is determined 'on line' during the trajectory by the maximization condition on the Hamiltonian : the thrust magnitude depends on the sign of the switching function, while the thrust direction is defined by the velocity costate.

4.6 Practical implementation

The trajectory is integrated numerically from the initial date t_0 to the guessed final date t_f using the free Fortran software Dop853.f which implements an 8th order Runge-Kutta method with an adaptative stepsize.

Three kinds of shooting problems must be solved at the successive stages of the continuation procedure :

- The starting point corresponds to the flat Earth problem : $(\text{OCP})_{0,0} = (\text{OCP})_{\text{flat}}$. This problem is solved from scratch by the Algorithm 1 presented in §3.8.4 without any specific initialization task.
- The first part of the continuation procedure consists in solving the equation $S_{\lambda_1,0}[P(t_0), t_f] = 0$ for a sequence of parameters λ_1 increasing from 0 to 1. These problems are of dimension 6 with a free final time.
- The second part of the continuation procedure consists in solving the equation $S_{\lambda_1, \lambda_2}[P(t_0)] = 0$ for a sequence of parameters λ_2 increasing from 0 to 1. These problems are of dimension 5 with the final time fixed at the optimal value of $(\text{OCP})_{1,0}$: $t_f = t_f^{\text{flat}}$. The final problem $(\text{OCP})_{1,1}$ is equivalent to the round Earth problem $(\text{OCP})_{\text{round}}$.

The shooting problems are solved using the free Fortran software Hybrd.f which implements a Newton-like method. The successive shooting problems prove relatively robust to the initial guess, so that quite large steps are possible on the parameters λ_1 and λ_2 from one problem to the next one. For the application example presented in §6, the steps used are respectively $\Delta\lambda_1 = 0.1$ and $\Delta\lambda_2 = 0.2$. In case of a step failure, the parameter increment is halved and the solution is restarted until the shooting method converges. The increment is then reset to its initial value to go on the continuation path.

It is possible to consider different paths in $[0,1]^2$ driving the parameters λ_1 and λ_2 from (0,0) to (1,1). The path proposed allows keeping the final time free until the intermediate problem $(\text{OCP})_{1,0}$. Since the value of the final time must be fixed in a somewhat arbitrary way, before solving the round Earth problem $(\text{OCP})_{1,1}$, it seems reasonable to fix this value as late as possible in the continuation procedure. The final time value must anyway be refined after the second continuation in order to get the maximum final mass under actual operational constraints (typically 2 engines re-ignition at most, or one revolution at most). Furthermore the numerical applications show that the first part of the continuation from $(\text{OCP})_{0,0}$ to $(\text{OCP})_{1,0}$ introducing the variable gravity in the flat Earth model is quite fast and robust. Hence considering a more direct continuation path from $(\text{OCP})_{0,0}$ to $(\text{OCP})_{1,1}$ would not yield a significant gain of execution time, while taking more risks regarding the convergence.

This algorithmic procedure provides a way of solving the upper stage coplanar transfer problem without any a priori knowledge on the optimal solution. The price to pay is that, instead of solving only one optimal control problem, one has to solve the series of $(\text{OCP})_{\lambda_1, \lambda_2}$. However the whole procedure remains time-efficient since the shooting method relies on a Newton-like method. It offers the advantage of starting from scratch and therefore of by-passing the initialization issues usually encountered when trying to solve directly the problem with the shooting method.

4.7 Round Earth problem costate

The above continuation process yields, if successful, the solution of the modified flat Earth problem $(\text{OCP})_{1,1}$. The state, the costate and the control are obtained in the modified coordinate system :

$$\mathbf{X}(t) = \begin{pmatrix} d(t) \\ h(t) \\ v_d(t) \\ v_h(t) \\ m(t) \end{pmatrix}, \quad \mathbf{P}(t) = \begin{pmatrix} p_d(t) \\ p_h(t) \\ p_{vd}(t) \\ p_{vh}(t) \\ p_m(t) \end{pmatrix}, \quad \mathbf{U}(t) = \mathbf{T}(t) \begin{pmatrix} u_d(t) \\ u_h(t) \end{pmatrix} \quad \text{for } t \in [t_0, t_f]$$

We denote with an upper bar the state, the costate, the control and the Hamiltonian corresponding to the solution of the initial round Earth problem $(\text{OCP})_{\text{round}}$ in the polar coordinate system (§4.2) :

$$\bar{\mathbf{X}}(t) = \begin{pmatrix} r(t) \\ \varphi(t) \\ v(t) \\ \gamma(t) \\ m(t) \end{pmatrix}, \quad \bar{\mathbf{P}}(t) = \begin{pmatrix} p_r(t) \\ p_\varphi(t) \\ p_v(t) \\ p_\gamma(t) \\ p_m(t) \end{pmatrix}, \quad \bar{\mathbf{U}}(t) = \mathbf{T}(t) \begin{pmatrix} u_v(t) \\ u_\gamma(t) \end{pmatrix} \quad \text{for } t \in [t_0, t_f]$$

The problems $(\text{OCP})_{1,1}$ and $(\text{OCP})_{\text{round}}$ are equivalent up to the change of coordinates (76) and (77) defining a diffeomorphism \mathbf{F} and its inverse $\mathbf{G}=\mathbf{F}^{-1}$:

$$\begin{cases} r = r_E + h \\ \varphi = \frac{d}{r_E + h} \\ v = \sqrt{v_d^2 + v_h^2} \\ \gamma = \text{Arctan} \frac{v_h}{v_d} \end{cases} \Leftrightarrow \bar{\mathbf{X}} = \mathbf{F}(\mathbf{X}) \quad (93)$$

$$\begin{cases} d = r\varphi \\ h = r - r_E \\ v_d = v \cos \gamma \\ v_h = v \sin \gamma \end{cases} \Leftrightarrow \mathbf{X} = \mathbf{G}(\bar{\mathbf{X}}) \quad (94)$$

$$\begin{pmatrix} u_d \\ u_h \end{pmatrix} = \begin{pmatrix} \cos \gamma & -\sin \gamma \\ \sin \gamma & \cos \gamma \end{pmatrix} \begin{pmatrix} u_v \\ u_\gamma \end{pmatrix} \quad (95)$$

The mass component is unchanged in the state vectors \mathbf{X} and $\bar{\mathbf{X}}$, and it is therefore not taken into account in \mathbf{F} .

The state and the control of the initial round Earth problem are retrieved by inverting the above coordinate change.

The corresponding costate transformation can be established by different ways :

- The analytical approach consists in using a generating function to derive the canonical transformation from (X, P) to (\bar{X}, \bar{P}) .
- The geometrical approach relies on the intrinsic character of the PMP .

4.7.1 Analytical approach

The canonical transformation can be guessed by considering that the Hamiltonian must be identical, up to a function of the time, for both coordinate systems. For a point transformation like (93), this is more direct than the usual approach consisting in the differentiation of a generating function²².

We denote respectively H and \bar{H} the Hamiltonian in the 2 coordinate systems :

$$\begin{cases} H[X(t), P(t), U(t)] = P^t(t) \dot{X}(t) \\ \bar{H}[\bar{X}(t), \bar{P}(t), \bar{U}(t)] = \bar{P}^t(t) \dot{\bar{X}}(t) \end{cases} \quad \text{for } t \in [t_0, t_f] \quad (96)$$

For convenience, the time variable is dropped in what follows. Since the system is autonomous, the Hamiltonian does not depend explicitly on the time. We thus impose the condition, without additive function of the time :

$$\bar{H}[\bar{X}, \bar{P}, \bar{U}] = H[X, P, U] \Leftrightarrow \bar{P}^t \dot{\bar{X}} = P^t \dot{X} \quad (97)$$

Differentiating the coordinate change, with dF denoting the differential of F , we have :

$$\bar{X} = F(X) \Rightarrow \dot{\bar{X}} = dF(X) \dot{X} \quad (98)$$

Replacing in (97) gives :

$$\begin{aligned} \bar{P}^t \dot{\bar{X}} = \bar{P}^t [dF(X) \dot{X}] &= [\bar{P}^t dF(X)] \dot{X} = [dF(X)^t \bar{P}] \dot{X} \\ &= P^t \dot{X} \end{aligned} \quad (99)$$

F being a diffeomorphism, the differential $dF(X)$ is not singular.

A sufficient condition to have a canonical transformation is to define the costate coordinate change as :

$$\bar{P} = dF(X)^{-t} P \quad (100)$$

It must now be checked that \bar{P} defined by (Eq 100) satisfies the PMP. By Cauchy uniqueness, this will ensure that \bar{P} is indeed the costate in the new coordinate system. This analytical demonstration is tedious and we will not undertake it. Indeed it can be established in a much more elegant way by the following geometrical approach.

4.7.2 Geometrical approach

We recast the Pontryagin Maximum Principle under a geometrical form in order to recall its intrinsic character.

Let M and N be smooth manifolds of respective dimensions n and m , M_0 and M_f two regular submanifolds of M and U a subset of N .

We consider the autonomous control system represented on M by a first order ODE :

$$\begin{aligned} \dot{X}(t) &= f[X(t), U(t)] \quad \text{for } t \in [t_0, t_f] \\ f &: M \times N \rightarrow TM \end{aligned} \quad (101)$$

The function f is smooth, taking its value in the tangent bundle TM . The control U is a bounded measurable function taking its value in a subset U of N .

We consider the optimal control problem at free or fixed final time :

$$\begin{aligned} \text{Max}_{U(\cdot), t_f} J(U, t_f) &= \int_{t_0}^{t_f} f^0(X(t), U(t)) dt \quad \text{s.t.} \quad \begin{cases} \dot{X}(t) = f[X(t), U(t)] & \text{for } t \in [t_0, t_f] \\ X(t_0) \in M_0 \\ X(t_f) \in M_f \end{cases} \\ J &: N \times R \rightarrow R \\ f^0 &: M \times N \rightarrow R \end{aligned} \quad (102)$$

where J is the cost function to minimize and f^0 is a smooth function.

According to the Pontryagin Maximum Principle, if $X(\cdot)$ is an optimal trajectory, there exists :

- a control $U(\cdot)$ on $[t_0, t_f]$,
- an absolutely continuous mapping $P(\cdot): t \in [t_0, t_f] \mapsto P(t) \in T_{X(t)}^*M$, named the costate vector, taking its value in the cotangent bundle T^*M at $X(t)$,
- a non positive real number p^0 , with $(P(\cdot), p^0) \neq (0, 0)$
- a Hamiltonian function

$$H[X, P, p^0, U] = \langle P, f(X, U) \rangle + p^0 f^0(X, U) \quad (103)$$

and it satisfies :

- the Hamiltonian system :

$$\begin{cases} \dot{X}(t) = \frac{\partial H}{\partial P}[X(t), P(t), p^0, U(t)] \\ \dot{P}(t) = -\frac{\partial H}{\partial X}[X(t), P(t), p^0, U(t)] \end{cases} \quad \text{for almost every } t \in [t_0, t_f] \quad (104)$$

- the maximization of the Hamiltonian :

$$H[X(t), P(t), p^0, U(t)] = \text{Max}_{W(t)} H[X(t), P(t), p^0, W(t)] \quad \text{for almost every } t \in [t_0, t_f] \quad (105)$$

- the transversality conditions at both endpoints :

$$\begin{cases} P(t_0) \perp T_{X(t_0)}M_0 \\ P(t_f) \perp T_{X(t_f)}M_f \end{cases} \quad (106)$$

Formulated in this way on a manifold, the Pontryagin Maximum Principle is intrinsic, i.e. its statement does not depend on a specific choice of coordinates²³.

Let now \bar{M} and \bar{N} be two other smooth manifolds of the same dimension n and m of respectively M and N .

We consider two diffeomorphisms $\Phi: M \rightarrow \bar{M}$ and $\Psi: N \rightarrow \bar{N}$ defining the coordinate changes :

$$\begin{cases} \bar{X} = \Phi(X) \\ \bar{U} = \Psi(U) \end{cases} \quad (107)$$

The differential $d\Phi$ defines a diffeomorphism between the tangent bundles TM and $T\bar{M}$, while the transpose of its inverse $d\Phi^{-1}$ defines a diffeomorphism between the cotangent bundles T^*M and $T^*\bar{M}$ (Refs^{22,24}). From the intrinsic character of the PMP, \dot{X} and P are respectively mapped onto $\dot{\bar{X}}$ and \bar{P} by $d\Phi$ and $d\Phi^{-1}$:

$$\begin{cases} \dot{\bar{X}} = d\Phi(X)\dot{X} \\ \bar{P} = d\Phi(X)^{-1}P \end{cases} \quad (108)$$

which is analogous to (100).

4.7.3 Round Earth costate

Instead of applying the transformation (100) with the diffeomorphism F defining the state coordinate change (93), it is easier to work with its inverse G (94). Indeed :

$$\begin{cases} \bar{X} = F(X) \\ X = G(\bar{X}) \end{cases} \Rightarrow dF(X)^{-1} = dG(\bar{X}) \quad (109)$$

so that :

$$\bar{P} = dG(\bar{X})^t P \quad (110)$$

The differential of G is :

$$G \begin{pmatrix} r \\ \varphi \\ v \\ \gamma \end{pmatrix} = \begin{pmatrix} r\varphi \\ r - r_E \\ v\cos\gamma \\ v\sin\gamma \end{pmatrix} \Rightarrow dG \begin{pmatrix} r \\ \varphi \\ v \\ \gamma \end{pmatrix} = \begin{pmatrix} \varphi & r & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & \cos\gamma & -v\sin\gamma \\ 0 & 0 & \sin\gamma & v\cos\gamma \end{pmatrix} \quad (111)$$

The costate for the initial round Earth problem $(OCP)_{\text{round}}$ is thus :

$$\begin{cases} p_r = \varphi p_d + p_h \\ p_\varphi = r p_d \\ p_v = \cos\gamma p_{vd} + \sin\gamma p_{vh} \\ p_\gamma = -v \sin\gamma p_{vd} + v \cos\gamma p_{vh} \end{cases} \quad (112)$$

4.8 Final conditions

Going back to the general 3-dimensional transfer problem, the upper stage mission is generally specified in terms of the targeted orbit parameters, yielding the conditions (15). For a coplanar transfer, these conditions reduce to (73), with an additional condition on the orbit orientation.

The continuation procedure described in §4.4 solves the coplanar transfer problem subject to the final conditions (72) :

- The values specified for the altitude and the velocity components correspond to an injection at the perigee of the targeted orbit. The final anomaly is thus fixed at zero.
- The free final downrange means that the perigee position is free in the transfer plane. The orbit orientation is not controlled and the perigee argument is thus free.

In practice, for an upper stage mission :

- Either the targeted orbit is circular. There is then no orientation constraint. The conditions (72) are equivalent to a free injection anomaly and they are representative of the actual problem.
- Or the targeted orbit is elliptical. The actual constraints are generally a fixed perigee argument and a free injection anomaly (no rendezvous constraint), and the conditions (72) are not representative of the actual problem.

Moreover the problem $(OCP)_{1,1}$ is solved at fixed final time, whereas the mission duration is rather limited by the number of engines ignition or the number of revolutions.

In order to be representative in the general transfer problem, the following improvements should be envisioned :

- The final time must be optimized, while staying compliant with the operational constraints mentioned here above. As mentioned in §4.4.2, this can be done by an additional continuation on the fixed final time value until finding a local maximum for the final mass. This method is applied on the application example of §6.
- In the case of an elliptical orbit, different approaches are possible. One method would be to add a first continuation on the constrained final downrange, starting from the initial value of $(OCP)_{1,1}$, until reaching the desired perigee position. A second additional continuation on the injection anomaly could then locate the injection position yielding a local maximum for the final mass. Another method would be to add a single continuation between the final conditions (72) expressed on the state components to the final conditions expressed on the constrained orbital parameters, with the respective changes in the transversality conditions.

In the same perspective it can be envisioned to solve the 3-dimensional problem starting from the coplanar solution, by adding a continuation on the inclination and the ascending node longitude. This has a reasonable chance to work as long as the required plane change remains small, as is generally the case for an upper stage mission.

5 Bi-impulse model

In this chapter an alternative method is proposed to initialize the costate vector for the round Earth problem. This method is more direct than the continuation from the flat Earth model and it relies on the solution of the bi-impulse transfer problem. This problem is a simplified instance of the general transfer problem presented in §2.3 with the additional assumption of an infinite thrust level. The maneuvers are modeled as velocity impulses, i.e. instantaneous velocity changes. Owing to this assumption the optimal control problem reduces to a nonlinear programming problem which can be easily solved. The Lagrange multipliers associated to the initial state provide a fairly good guess for the initial costate, since both represent the optimal cost derivatives. Starting from this initialization, a continuation on the thrust level allows solving the continuous thrust transfer problem by the shooting method.

5.1 Problem statement

The bi-impulse transfer consists of two instantaneous velocity increments delivered at optimized dates. The problem formulation is very similar to the general transfer problem of §2.3, with the difference that the continuous thrust maneuvers are replaced by velocity impulses. Each impulse is modeled by a discontinuous change of the state vector velocity components while the position components remain unchanged. Between the impulses the vehicle follows a coast arc under the influence of the Earth gravitational field.

Only two impulses are allowed corresponding to the upper stage engine re-ignition capability. The first and second impulses are denoted respectively $\Delta\vec{v}_0$ and $\Delta\vec{v}_f$. They are delivered at the upper stage injection date t_0 and at the orbit injection date t_f . Denoting t^- and t^+ the dates before and after an impulse, the mass consumed by a velocity increment $\Delta\vec{v}$ is assessed from the rocket equation, also known as Tsiolkovsky formula²⁵ :

$$\Delta v = v_e \ln \frac{m(t^-)}{m(t^+)} \quad (113)$$

Where :

- $\Delta v = \|\Delta\vec{v}\|$ is the impulse modulus
- v_e is the engine exhaust velocity
- m is the vehicle mass

The mass Δm consumed by the velocity increment $\Delta\vec{v}$ is then : $\Delta m = m(t^-) - m(t^+)$

Maximizing the final mass is equivalent to minimizing the sum of the two impulse modulus. This bi-impulse problem has been widely studied and various solution methods have been proposed depending on the initial and final constraints^{25,26,27}. Anyway there is no fully explicit solution that could be applied for the present problem and we have to turn to numerical solution methods. For that purpose the bi-impulse problem is formulated as an NLP problem in the next section.

5.1.1 Direct formulation

With the above assumptions the vehicle dynamics is modeled as follows :

- The initial state at t_0^- is denoted with a subscript 0. It is fixed, corresponding to the position, velocity and gross mass at the upper stage ignition :

$$\mathbf{X}(t_0^-) = \mathbf{X}_0 \Leftrightarrow \begin{cases} \bar{\mathbf{r}}(t_0^-) = \bar{\mathbf{r}}_0 \\ \bar{\mathbf{v}}(t_0^-) = \bar{\mathbf{v}}_0 \\ m(t_0^-) = m_0 \end{cases} \quad (114)$$

- The first impulse is delivered at t_0 . It puts the vehicle on a coast arc crossing the targeted orbit :

$$\mathbf{X}(t_0^+) = \mathbf{X}(t_0^-) + \begin{pmatrix} \bar{\mathbf{0}} \\ \Delta\bar{\mathbf{v}}_0 \\ -\Delta m_0 \end{pmatrix} \Leftrightarrow \begin{cases} \bar{\mathbf{r}}(t_0^+) = \bar{\mathbf{r}}_0 \\ \bar{\mathbf{v}}(t_0^+) = \bar{\mathbf{v}}_0 + \Delta\bar{\mathbf{v}}_0 \\ m(t_0^+) = m_0 - \Delta m_0 \end{cases} \quad (115)$$

- During the coast arc from t_0^+ to t_f^- the vehicle is only subjected to the Earth gravitational field. The vehicle dynamics is represented by the first order ODE :

$$\begin{cases} \dot{\bar{\mathbf{r}}}(t) = \bar{\mathbf{v}}(t) \\ \dot{\bar{\mathbf{v}}}(t) = \bar{\mathbf{g}}(\bar{\mathbf{r}}(t)) & \text{for } t \in [t_0^+, t_f^-] \\ \dot{m}(t) = 0 \end{cases} \Rightarrow m(t_f^-) = m(t_0^+) \quad (116)$$

- The second impulse is delivered at t_f when the vehicle crosses the targeted orbit. It corrects the velocity to put the vehicle on the targeted orbit.

$$\mathbf{X}(t_f^+) = \mathbf{X}(t_f^-) + \begin{pmatrix} \bar{\mathbf{0}} \\ \Delta\bar{\mathbf{v}}_f \\ -\Delta m_f \end{pmatrix} \Leftrightarrow \begin{cases} \bar{\mathbf{r}}(t_f^+) = \bar{\mathbf{r}}(t_f^-) \\ \bar{\mathbf{v}}(t_f^+) = \bar{\mathbf{v}}(t_f^-) + \Delta\bar{\mathbf{v}}_f \\ m(t_f^+) = m(t_f^-) - \Delta m_f \end{cases} \quad (117)$$

- The final state at t_f^+ is denoted with a subscript f :

$$\mathbf{X}_f = \mathbf{X}(t_f^+) \Leftrightarrow \begin{cases} \bar{\mathbf{r}}_f = \bar{\mathbf{r}}(t_f^+) \\ \bar{\mathbf{v}}_f = \bar{\mathbf{v}}(t_f^+) \\ m_f = m(t_f^+) \end{cases} \quad (118)$$

The final position $\bar{\mathbf{r}}_f$ and velocity $\bar{\mathbf{v}}_f$ define the injection orbit parameters (15) :

- Energy : K_f
- Angular momentum : $\bar{\mathbf{h}}_f$
- Eccentricity vector: $\bar{\mathbf{e}}_f$
- Anomaly or longitude : L_f

The final state is constrained on the targeted orbit, specified by its energy K_T , its angular momentum \vec{h}_T and its eccentricity vector \vec{e}_T . For the coplanar transfer problem, the angular momentum constraint is dropped. Nevertheless we keep it in the problem formulation since the methodology presented applies similarly in a 2-dimensional or a 3-dimensional space. This is illustrated on the application example presented in §6.

For the bi-impulse transfer problem there are thus 5 final constraints that do not depend explicitly of the time :

$$\begin{cases} K_f = K_T \\ \vec{h}_f = \vec{h}_T \\ \vec{e}_f = \vec{e}_T \end{cases} \quad (119)$$

The injection anomaly or longitude is free. The rough formulation of the bi-impulse transfer problem is :

$$\text{Max}_{\Delta\vec{v}_0, \Delta\vec{v}_f, t_f} m_f \quad \text{s.t.} \quad \begin{cases} K_f = K_T \\ \vec{h}_f = \vec{h}_T \\ \vec{e}_f = \vec{e}_T \end{cases} \quad (120)$$

In the 3-dimensional space this nonlinear programming problem has 7 variables and 5 constraints. In the 2-dimensional space the dimensions reduce to respectively 5 variables and 3 constraints.

5.1.2 Problem reduction

Although it is quite possible to tackle directly the bi-impulse transfer problem under the formulation (120), we modify it in order to reduce the NLP problem dimension and to ease the solution. The transfer problem is transformed into a rendezvous problem by :

- defining an anomaly time law $L_T(t)$ on the targeted orbit starting from an initial value L_{T0} at t_0 :

$$L_T(t_0) = L_{T0} \quad (121)$$

- adding a constraint on the injection anomaly L_f :

$$L_f = L_T(t_f) \quad (122)$$

In order to make the rendezvous problem equivalent to the transfer problem, the initial value L_{T0} is free and it must be chosen to maximize the cost function m_f . This amounts in fact to free the injection anomaly, as in the initial transfer problem.

With this additional anomaly constraint the final position \vec{r}_f and velocity \vec{v}_f become completely determined for a given injection date t_f :

$$\begin{cases} \mathbf{K}_f = \mathbf{K}_T \\ \vec{h}_f = \vec{h}_T \\ \vec{e}_f = \vec{e}_T \\ \mathbf{L}_f = \mathbf{L}_T(t_f) \end{cases} \Leftrightarrow \begin{cases} \vec{r}_f = \vec{r}_T(t_f) \\ \vec{v}_f = \vec{v}_T(t_f) \end{cases} \quad (123)$$

The coast arc is therefore constrained by the initial position \vec{r}_0 , the final position \vec{r}_f and the coast duration $\Delta t = t_f - t_0$. In that way the transfer is formulated as a Lambert problem that can be tackled efficiently by many numerical methods¹⁷. Solving this Lambert problem for a given final date t_f , we obtain the velocities at the coast arc endpoints, respectively $\vec{v}(t_0^+)$ and $\vec{v}(t_f^-)$. The corresponding impulses are then directly computed as the velocity differences :

$$\begin{cases} \Delta \vec{v}_0 = \vec{v}(t_0^+) - \vec{v}_0 \\ \Delta \vec{v}_f = \vec{v}_T - \vec{v}(t_f^-) \end{cases} \quad (124)$$

With the related mass evolution derived from the Tsiolkovsky formula (113) :

$$\begin{cases} m(t_0^+) = m_0 \cdot e^{-\frac{\Delta v_0}{v_c}} \\ m_f = m(t_f^-) \cdot e^{-\frac{\Delta v_f}{v_c}} \end{cases} \Rightarrow m_f = m_0 \cdot e^{-\frac{\Delta v_0 + \Delta v_f}{v_c}} \quad (125)$$

The Lambert transfer is depicted on the Figure 4.

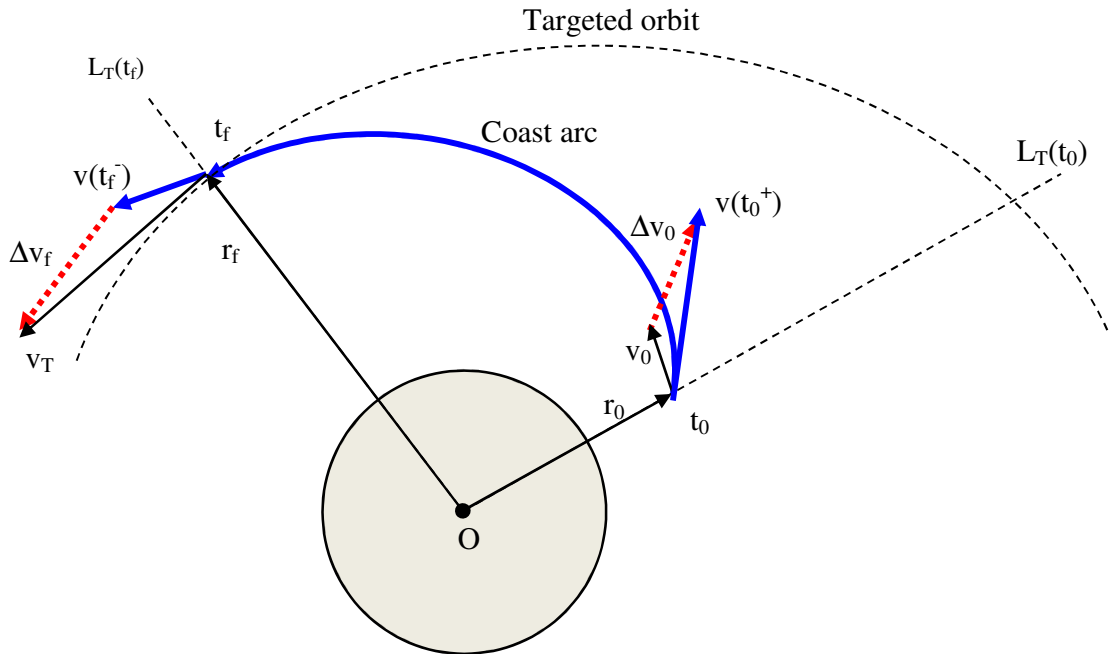


Figure 4 : Lambert transfer

The formulation of the bi-impulse transfer problem as a rendezvous problem allows the following reductions :

- The impulses $\Delta\bar{v}_0$ and $\Delta\bar{v}_f$ are assessed from the final date t_f by solving numerically the associated Lambert problem. They are no longer unknowns of the problem and they can be discarded from the problem variables.
- The final constraints (123) are automatically met, for the position $\bar{r}_f = \bar{r}_T(t_f)$ by solving the Lambert problem, and for the velocity $\bar{v}_f = \bar{v}_T(t_f)$ by assessing the impulses from (124). They are also discarded from the problem formulation.
- On the other hand, the initial anomaly L_{T_0} on the targeted orbit is added to the problem variables in order to make the rendezvous problem equivalent to the transfer problem.

The bi-impulse transfer problem (120) is therefore reformulated as an unconstrained problem with 2 variables :

$$\text{Max}_{L_{T_0}, t_f} m_f \quad (126)$$

This unconstrained problem is quite easily solved by any NLP solver, yielding the optimal impulsive maneuvers $\Delta\bar{v}_0$, $\Delta\bar{v}_f$ and the transfer duration t_f . For the numerical application, an internal NLP solver developed at Astrium for space mission analyses has been used. Since there are only two unknowns it is even possible to solve this unconstrained problem by a two levels sweeping method (e.g. with a golden search section), provided that the search bounds have been correctly chosen. The same solution methodology is applicable if some initial orbital parameters (including the anomaly) or some final orbital parameters are free, by adding these parameters to the optimization variables L_{T_0} and t_f in (126).

5.2 Initial costate assessment

The solution of the bi-impulse transfer problem can be used to provide an initial costate guess for the continuous thrust transfer problem.

From the PMP, the costate vector represents the sensitivity of the cost function to the state along an extremal :

$$P(t) = \frac{\partial J[X(t_f), t_f]}{\partial X(t)} \quad \text{for } t \in [t_0, t_f] \quad (127)$$

The initial costate can thus be assessed from the derivatives of the cost function wrt to initial state :

$$P(t_0) = \frac{\partial J[X(t_f), t_f]}{\partial X(t_0)} = \frac{\partial m_f}{\partial X_0} \quad (128)$$

These derivatives may be assessed either by finite differences, or by Lagrange multipliers. An analytical assessment is also possible under simplifying assumptions.

5.2.1 Finite differences

The optimization problem is solved applying successively a small perturbation δX_i on each component X_{0i} of the initial state vector (position \vec{r}_0 , velocity \vec{v}_0 , mass m_0). This requires solving one optimization problem per state component. The additional computational cost is small : indeed the unperturbed solution provides a very good initialization for each perturbed problem, so that the convergence is immediate.

5.2.2 Lagrange multipliers

We consider the following constrained optimization problem :

$$\text{Min}_x f(x) \quad \text{s.t.} \quad g(x) = c \quad (129)$$

The optimal solution is denoted with a superscript * :

$$\text{Solution : } \begin{cases} x = x^* \\ f(x^*) = f^* \end{cases} \quad (130)$$

The Lagrange multipliers λ of the constraints g represent the optimal cost sensitivity to the constraint levels, up to the sign²⁸ :

$$\lambda = - \frac{\partial f^*}{\partial c} \quad (131)$$

Most NLP solvers provide the Lagrange multipliers at the end of the optimization. In order to assess the initial costate from the Lagrange multipliers, the bi-impulse transfer problem (126) is reformulated as :

$$\text{Max}_{X(t_0^-), t_1, t_f} m_f \quad \text{s.t.} \quad X(t_0^-) = X_0 \quad (132)$$

This problem is strictly equivalent to the unconstrained problem (126), since the variables $X(t_0^-)$ corresponding to a free initial state are in fact constrained to their fixed values X_0 . The Lagrange multipliers associated to the constraints $X(t_0^-) = X_0$ are then :

$$\lambda = - \frac{\partial m_f}{\partial X_0} \quad (133)$$

By this way, a single optimization run yields the initial costate $P(t_0)$ up to the sign.

5.2.3 Analytical assessment

The derivatives the cost function (125) are given by :

$$\begin{aligned}
m_f = m_0 e^{-\frac{\Delta v_0 + \Delta v_f}{v_e}} &\Rightarrow \begin{cases} \frac{\partial m_f}{\partial m_0} = e^{-\frac{\Delta v_0 + \Delta v_f}{v_e}} \\ \frac{\partial m_f}{\partial x} = m_0 e^{-\frac{\Delta v_0 + \Delta v_f}{v_e}} \left(\frac{-1}{v_e} \right) \left(\frac{\partial \Delta v_0}{\partial x} + \frac{\partial \Delta v_f}{\partial x} \right) \end{cases} \\
&\Rightarrow \begin{cases} \frac{\partial m_f}{\partial m_0} = \frac{m_f}{m_0} \\ \frac{\partial m_f}{\partial x} = -\frac{m_f}{v_e} \left(\frac{\partial \Delta v_0}{\partial x} + \frac{\partial \Delta v_f}{\partial x} \right) \end{cases}
\end{aligned} \tag{134}$$

where x denotes a component of the initial state other than the mass.

An analytical assessment of the initial costate is possible under the simplifying assumption of a Hohmann-like transfer. In many practical cases the upper stage is injected in the vicinity of the apogee of its initial osculating orbit, with a flight path angle near to zero. For a circular targeted orbit, the optimal bi-impulse transfer consists then in a half elliptical orbit with tangential boosts at the endpoints (Figure 5).

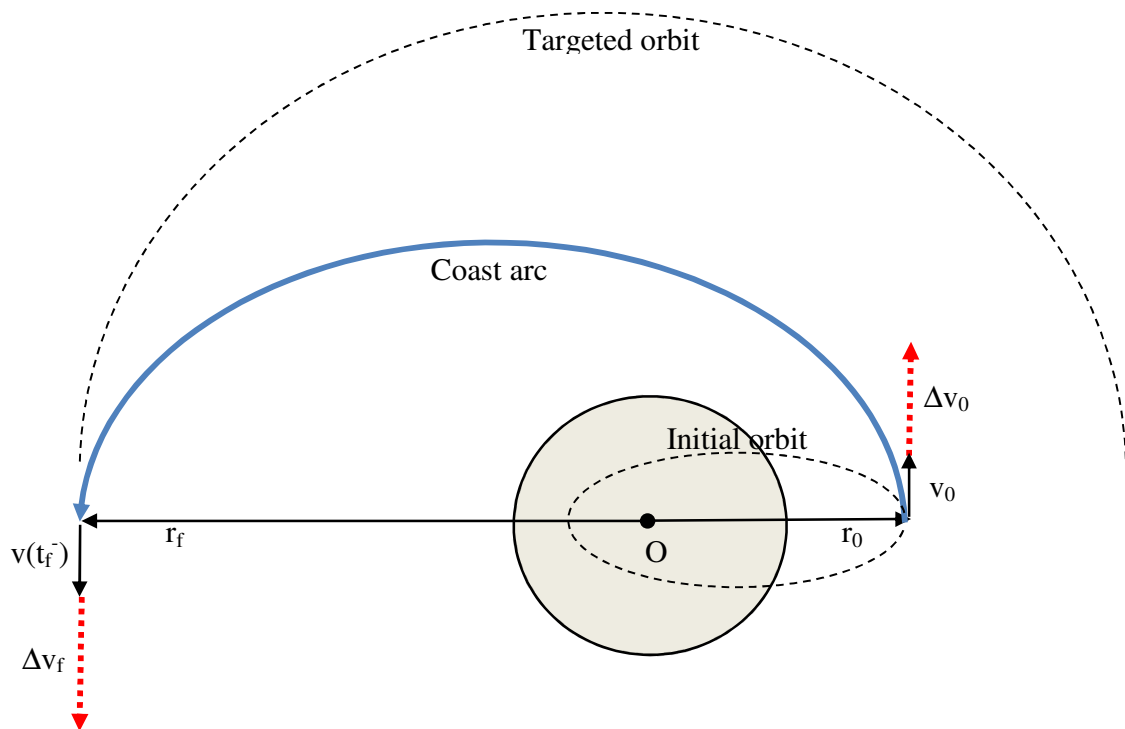


Figure 5 : Hohmann-like transfer

The coast arc semi-major axis is :

$$a = \frac{r_0 + r_f}{2} \quad (135)$$

The velocities at the coast arc endpoints are given by :

$$\begin{cases} v(t_0^+) = \sqrt{\mu \left(\frac{2}{r_0} - \frac{1}{a} \right)} \\ v(t_f^-) = \sqrt{\mu \left(\frac{2}{r_f} - \frac{1}{a} \right)} \end{cases} \quad (136)$$

The tangential initial and final impulses are assessed by :

$$\begin{cases} \Delta v_0 = v(t_0^+) - v_0 = \sqrt{\mu \left(\frac{2}{r_0} - \frac{1}{a} \right)} - v_0 \\ \Delta v_f = v_T - v(t_f^-) = v_T - \sqrt{\mu \left(\frac{2}{r_f} - \frac{1}{a} \right)} \end{cases} \quad (137)$$

The impulses depend on the initial radius vector r_0 and the initial velocity modulus v_0 . For this specific transfer, we obtain thus an analytical assessment for the following cost derivatives :

$$\begin{cases} \frac{\partial m_f}{\partial r_0} = \frac{m_f}{v_e} \left(\frac{\mu}{r_0^2 v(t_0^+)} - \frac{\mu}{4a^2} \left(\frac{1}{v(t_0^+)} - \frac{1}{v(t_f^-)} \right) \right) \\ \frac{\partial m_f}{\partial v_0} = \frac{m_f}{v_e} \\ \frac{\partial m_f}{\partial m_0} = \frac{m_f}{m_0} \end{cases} \quad (138)$$

with m_f , a , $v(t_0^+)$ and $v(t_f^-)$ given respectively by (134), (135) and (136).

In a practical case, these analytical derivatives provide a good initialization for the initial costate components $P_r(t_0)$, $P_v(t_0)$ and $P_m(t_0)$ as long as the initial state is close to the apogee of the initial osculating orbit, and the targeted orbit is close to circular. The flight path angle costate $P_\gamma(t_0)$ must nevertheless be assessed by numerical derivation.

5.3 Duration assessment

The boost durations for the continuous thrust transfer problem can be assessed from the bi-impulse transfer solution. For a finite thrust level T , the propellant flow rate q is given by :

$$T = qv_e = - \frac{dm}{dt} v_e \quad (139)$$

Assuming a constant flow rate, the boost duration can be assessed from the Tsiolkovsky formula :

- Impulse : $\Delta v = v_e \ln \frac{m(t^-)}{m(t^+)}$
- Mass consumption : $\Delta m_c = m(t^-) - m(t^+) = m(t^-) \left[1 - e^{-\frac{\Delta v}{v_e}} \right]$
- Boost duration : $\Delta t_c = \frac{\Delta m_c}{q}$

$$\Rightarrow \Delta t_c = \frac{m(t^-) v_e}{T} \left[1 - e^{-\frac{\Delta v}{v_e}} \right] \quad (140)$$

The coast arc duration for the continuous thrust transfer problem is initialized directly with the bi-impulse coast duration.

5.4 Thrust level continuation

The method presented above yields an initial guess for the costate vector and for the sequence durations, derived from the bi-impulse transfer solution. This initial guess allows us to solve the continuous thrust transfer by the shooting method, provided that the thrust level is sufficiently high. Indeed the bi-impulse model is the limit of the continuous thrust model with an infinite thrust level and null boost durations.

To pass from the bi-impulse model to the continuous thrust model, we introduce a parameter denoted α . This parameter is a multiplicative factor on the thrust level. It permits us to pass continuously from a very high thrust level T_∞ to the actual vehicle thrust level T , via intermediate values T_α :

$$T_\alpha = (1 - \alpha)T_\infty + \alpha T \quad (141)$$

This defines a family of control systems with the parameter α varying between 0 and 1 :

$$\dot{X}(t) = f[X(t), U(t)] \Leftrightarrow \begin{cases} \dot{\bar{r}}(t) = \bar{v}(t) \\ \dot{\bar{v}}(t) = \frac{T_\alpha(t)}{m(t)} \bar{u}(t) + \bar{g}(\bar{r}(t)) \\ \dot{m}(t) = -\frac{T_\alpha(t)}{v_e} \end{cases} \quad \text{for } t \in [t_0, t_f] \quad (142)$$

The “infinite” thrust level T_∞ must be chosen so that the boost durations remain small compared to the total transfer time. In that case, the initial costate derived from the bi-impulse solution is sufficiently accurate to allow the shooting method to converge. The continuation procedure from $\alpha=0$ to $\alpha=1$ allows solving the continuous thrust transfer problem starting from the bi-impulse initialization.

In practice, it is even possible to bypass the thrust level continuation and try to solve directly the problem for $\alpha=1$ from the bi-impulse initialization. In case of failure, the continuation is started from the “infinite” thrust level T_∞ .

6 Application examples

This chapter illustrates the algorithmic procedures on practical application examples. Two cases are considered with different vehicle features, and different initial and final conditions. The targeted orbits for the first and second case are respectively sun-synchronous (SSO) and geostationary (GEO). An additional plane change is also considered for the first case in order to illustrate the bi-impulse procedure on a 3-dimensional transfer.

6.1 Example data

6.1.1 Case 1 : Sun-synchronous orbit

The polar coordinates at the upper stage ignition are :

- Altitude : $h_0 = 200 \text{ km} \Rightarrow r_0 = r_E + h_0 = 6578 \text{ km}$
- Longitude (or polar angle) : $\varphi_0 = 0 \text{ deg}$
- Absolute velocity modulus : $v_0 = 5500 \text{ m/s}$
- Flight path angle : $\gamma_0 = 2 \text{ deg}$

The initial apogee and perigee altitudes are respectively 204 km and -4194 km. The initial anomaly is 178 deg.

The upper stage uses a cryogenic engine. The mass and propulsion features are :

- Initial gross mass : $m_0 = 40\,000 \text{ kg}$
- Vacuum thrust : $T = 180 \text{ kN}$
- Vacuum specific impulse : $I_{sv} = 450 \text{ s} \Rightarrow v_e = 4413 \text{ m/s}$

The targeted orbit is circular at the altitude of 800 km. The injection conditions in polar coordinates are :

- Altitude : $h_f = 800 \text{ km} \Rightarrow r_f = r_E + h_f = 7178 \text{ km}$
- Longitude : $\varphi_f \text{ free} \Rightarrow \text{corresponding to a free injection anomaly}$
- Absolute velocity modulus : $v_f = 7500 \text{ m/s} \left(v_f \approx \sqrt{\frac{\mu}{r_f}} \right) \approx \text{orbital velocity at } r_f$
- Flight path angle : $\gamma_f = 0 \text{ deg}$

The polar coordinates are transformed into the modified coordinates through the coordinate changes (76) :

	Initial conditions	Final conditions
Downrange d (km)	0	free
Altitude h (km)	200	800
Horizontal velocity v_d (m/s)	5496.6	7500
Vertical velocity v_h (m/s)	191.9	0

Table 3 : Initial and final conditions (SSO)

For the bi-impulse procedure, a 3-dimensional transfer is also considered with the same data by requiring an additional plane change of 5 deg.

6.1.2 Case 2 : Geostationary orbit

The polar coordinates at the upper stage ignition are :

- Altitude : $h_0 = 4000 \text{ km} \Rightarrow r_0 = r_E + h_0 = 10378 \text{ km}$
- Longitude (or polar angle) : $\varphi_0 = 0 \text{ deg}$
- Absolute velocity modulus : $v_0 = 7700 \text{ m/s}$
- Flight path angle : $\gamma_0 = 30 \text{ deg}$

The initial apogee and perigee altitudes are respectively 31987 km and 745 km. The initial anomaly is 77 deg.

The upper stage uses storable propellants. The mass and propulsion features are :

- Initial gross mass : $m_0 = 10\,000 \text{ kg}$
- Vacuum thrust : $T = 65 \text{ kN}$
- Vacuum specific impulse : $I_{sv} = 350 \text{ s} \Rightarrow v_e = 3432 \text{ m/s}$

The targeted orbit is circular at the altitude of 35786 km. The injection conditions in polar coordinates are :

- Altitude : $h_f = 35786 \text{ km} \Rightarrow r_f = r_E + h_f = 42164 \text{ km}$
- Longitude : $\varphi_f \text{ free} \Rightarrow \text{corresponding to a free injection anomaly}$
- Absolute velocity modulus : $v_f = 3075 \text{ m/s} \left(v_f \approx \sqrt{\frac{\mu}{r_f}} \right) \approx \text{orbital velocity at } r_f$
- Flight path angle : $\gamma_f = 0 \text{ deg}$

The polar coordinates are transformed into the modified coordinates through the coordinate changes (76) :

	Initial conditions	Final conditions
Downrange d (km)	0	free
Altitude h (km)	4000	35786
Horizontal velocity v_d (m/s)	6668.4	3075
Vertical velocity v_h (m/s)	3850.0	0

Table 4 : Initial and final conditions (GEO)

6.2 Continuation procedure from the flat Earth model

The continuation procedure from the flat Earth model is illustrated on the first test case (§6.1.1) with a sun-synchronous targeted orbit.

6.2.1 Initialization

The continuation procedure consists in solving a sequence of shooting problems starting with the Algorithm 1 presented in §3.8.4. With the normalization choice $p_h=1$, the initial problem is reduced to 2 unknowns p_{vd} and $p_{vh}(t_0)$. The algorithm converges instantaneously from scratch, provided that positive values are chosen as initial guess of p_{vd} and $p_{vh}(t_0)$, for example $p_{vd} = p_{vh}(t_0) = 100$. It requires 41 trajectory simulations. The solution of the initial problem is then renormalized so that $p_m(t_f) = -p^0 = 1$:

(OCP) _{0,0}	Final time (s)	Final mass (kg)	Hamiltonian	Costate $p_h(t_0)$	Costate $p_{vd}(t_0)$	Costate $p_{vh}(t_0)$	Costate $p_m(t_0)$
Initialization	1433.3	1676.2	0	1.000	96.349	1433.314	NA
Renormalization	1433.3	1676.2	0	0.756	72.808	1083.108	-0.137

NA = Not Assessed

Table 5 : Initialization

6.2.2 Gravity continuation

This solution is used as starting point for the first continuation, which introduces the variable gravity into the flat Earth model. The initial step increment is $\Delta\lambda_1=0.1$. The continuation is achieved in 15 steps, including some restarts with halved steps, and it requires 953 trajectory simulations. The solution is presented in the Table 6.

(OCP) _{1,0}	Final time (s)	Final mass (kg)	Hamiltonian	Costate $p_h(t_0)$	Costate $p_{vd}(t_0)$	Costate $p_{vh}(t_0)$	Costate $p_m(t_0)$
Gravity homotopy	1483.2	1504.8	0	3.850	69.874	2198.207	-0.236

Table 6 : Gravity continuation

We can notice that the performance is lowered when introducing the variable gravity, passing from $m_f = 1676.2$ kg to $m_f = 1504.8$ kg. This could seem counterintuitive since the variable gravity is always lower than the constant gravity taken at the Earth surface. In fact the gravity helps to flatten the trajectory by reducing the vertical velocity ('gravity turn'). It has thus a positive influence on the performance when a horizontal injection is targeted.

6.2.3 Dynamics continuation

At this step, we switch from a free final time to a fixed final time in order to have a well posed problem. The final time is fixed at the optimal value yielded by the first continuation $t_f = t_f^{\text{flat}} = 1483.2$ s, so that the solution is

unchanged between the free and the fixed final time problem. This means that the shooting function has from now on one less unknown and one less constraint to satisfy at the endpoint of the extremal flow.

This solution is used as starting point for the second continuation, which introduces the additional terms making the flat Earth model equivalent to the round Earth model. The initial step increment is $\Delta\lambda_2=0.2$. The continuation is achieved in 5 steps requiring 741 trajectory simulations. The solution is presented in the Table 7.

Compared to the first continuation, we observe that the number of simulations is not proportional to the number of homotopy steps. In fact larger steps may require more iterations of the Newton method because the initial point is farther from the solution. Different step sizes have therefore been tried in order to minimize the execution time, which is roughly proportional to the number of simulations. The choices $\Delta\lambda_1 = 0.1$, $\Delta\lambda_2 = 0.2$ prove a good compromise between time efficiency and robustness.

$(OCP)_{1,1}$	Final time (s)	Final mass (kg)	Hamiltonian	Costate $p_h(t_0)$	Costate $p_{vd}(t_0)$	Costate $p_{vh}(t_0)$	Costate $p_m(t_0)$
Dynamics homotopy	1483.2	18915.4	3.149	12.211	6821.530	5225.629	0.310

Table 7 : Dynamics continuation

We can notice that the performance for the round Earth model is far better than the one obtained for the flat Earth model. This could be expected because increasing the altitude is less costly in the round Earth model than in the flat Earth model. Indeed the Earth surface curvature makes the altitude naturally increase on a free motion (i.e. without external forces) as pictured on the Figure 6.

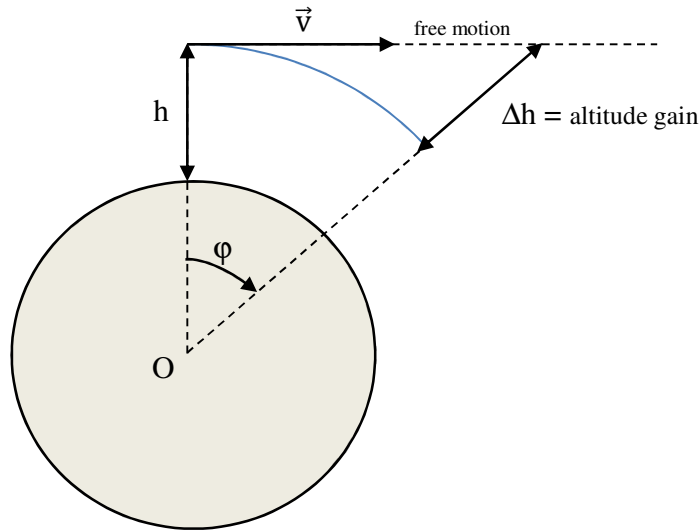


Figure 6 : Altitude increase in the round Earth model

This centrifugal effect is expressed in polar coordinates by the centrifugal term $\frac{v(t)}{r(t)} \cos \gamma(t)$ in the flight path angle equation (71). It allows increasing the apogee up to 800 km (targeted altitude) much more easily after the first boost and the coast arc. The durations of the first boost are indeed :

- $\Delta t_1 = 943.8\text{s}$ for $(\text{OCP})_{1,0}$
- $\Delta t_1 = 481.3\text{s}$ for $(\text{OCP})_{1,1}$

Most of the performance gain is thus realized on this first boost. For $(\text{OCP})_{1,1}$, the circularization at the altitude of 800 km is achieved by the second boost with a duration $\Delta t_2 = 35.6\text{s}$.

6.2.4 Continuation path

The Figure 7 shows the zero paths of the shooting function respectively for the first and the second homotopy.

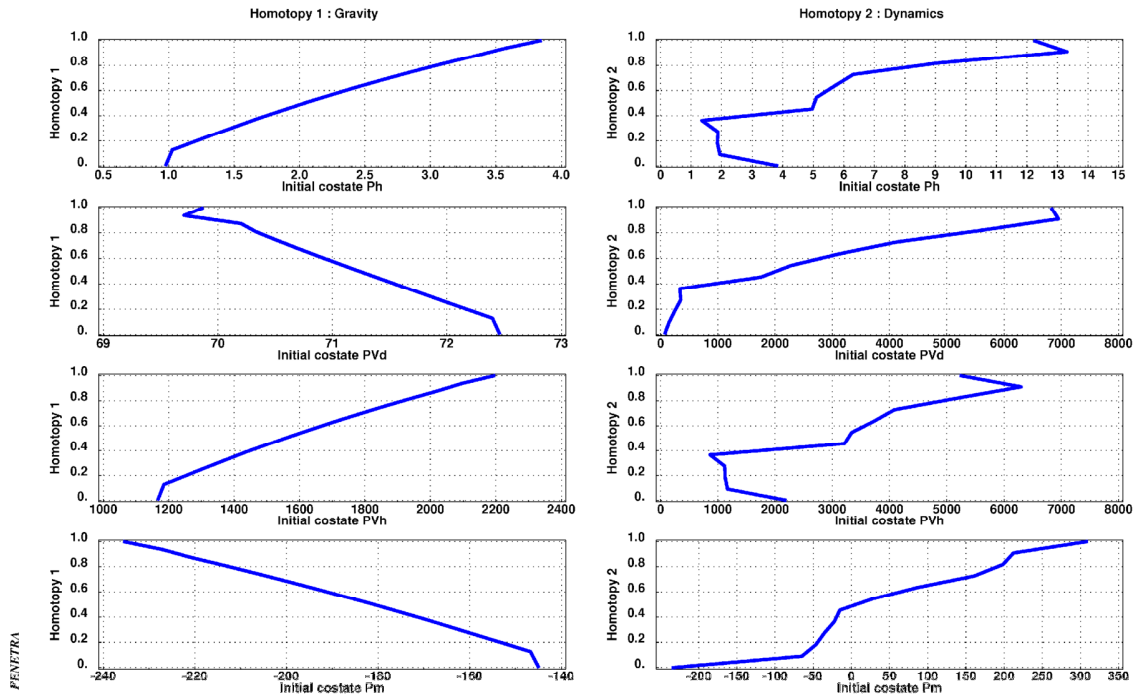


Figure 7 : Continuation path

It can be noticed that the second homotopy is not smooth around some values of the homotopy parameter λ_2 , namely $\lambda_2 \approx 0.4$, and $\lambda_2 \approx 0.8$. These values correspond to changes in the optimal thrust strategy :

- $T_{\max} - 0$ for $0 < \lambda_2 < 0.41$,
- $T_{\max} - 0 - T_{\max}$ for $0.41 < \lambda_2 < 0.78$,
- $T_{\max} - 0$ for $0.78 < \lambda_2 < 0.86$,
- $T_{\max} - 0 - T_{\max}$ for $0.86 < \lambda_2 < 1$.

When the optimal thrust strategy changes, the zero path of the shooting function makes corners. It can be observed numerically, by reducing the homotopy steps, that the path remains continuous but it is no longer differentiable. This phenomenon is due to the occurrence in the continuation process of a switching date (i.e. a zero of the switching function) coinciding with the final date. In that case, the switching function is still continuous but not differentiable (see ref¹³ for more details). This occurs 3 times for λ_2 taking the approximate values 0.41, 0.78 and 0.86.

The Figure 8 shows the evolution of the commutation dates and the final mass respectively for the first and the second homotopy.

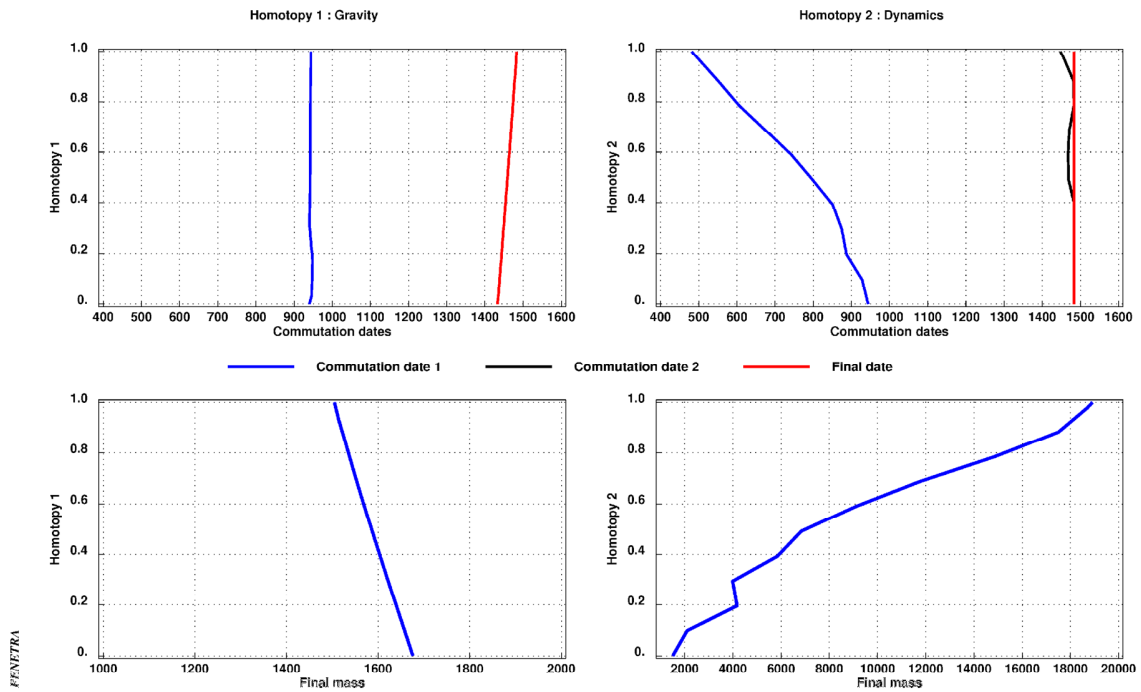


Figure 8 : Commutation dates and final mass

6.2.5 Trajectory variables

The following Figure 9 and Figure 10 compare the trajectory and control strategy obtained respectively :

- for $(\text{OCP})_{1,0}$ at the end of the first homotopy,
- for $(\text{OCP})_{1,1}$ at the end of the second homotopy.

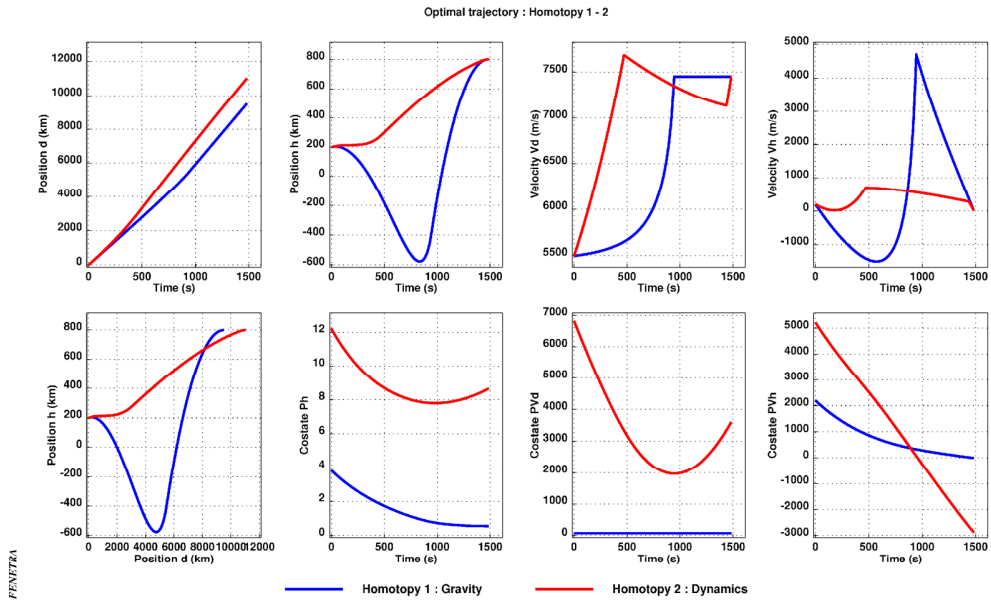


Figure 9 : Trajectory variables

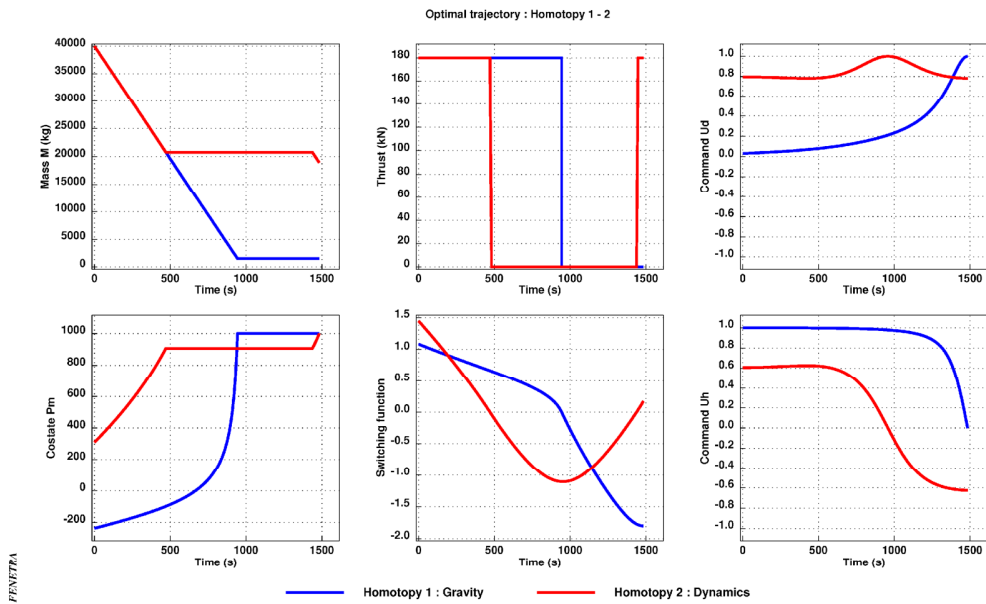


Figure 10 : Switching function - Control

It can be noticed that the altitude becomes negative on the optimal trajectory of $(OCP)_{1,0}$. This is not physically feasible. Nevertheless it corresponds only to an intermediate fictitious problem, and the minimum altitude condition is to be checked only on the solution of $(OCP)_{1,1}$ corresponding to the real problem with the round Earth model. For the present application example, the altitude is monotonously increasing and the minimum altitude constraint is naturally satisfied.

It may happen that the solution of $(OCP)_{1,1}$ violates this constraint that is not explicitly taken into account in the solution process. For a low initial acceleration level, the optimal trajectory may indeed begin with a diving leg to increase the velocity. In order to yield a feasible solution whatever the data, the present algorithm must be enhanced to cope with the minimum altitude constraint.

6.2.6 Final time

The solution obtained at the end of the second homotopy corresponds to a fixed final time, whose value has been fixed at the optimal value $t_f = t_f^{flat}$ yielded by the first homotopy. As mentioned in §4.8, it is possible to improve the performance by an additional continuation which consists in solving a sequence of fixed final time problems where the final time value varies.

The step on the final time can be taken up to $\Delta t_f = 100$ s. Each step requires on average 25 trajectory simulations. The curves on the Figure 11 show the evolution of the final mass, the Hamiltonian, the minimum altitude and the final longitude with the final time.

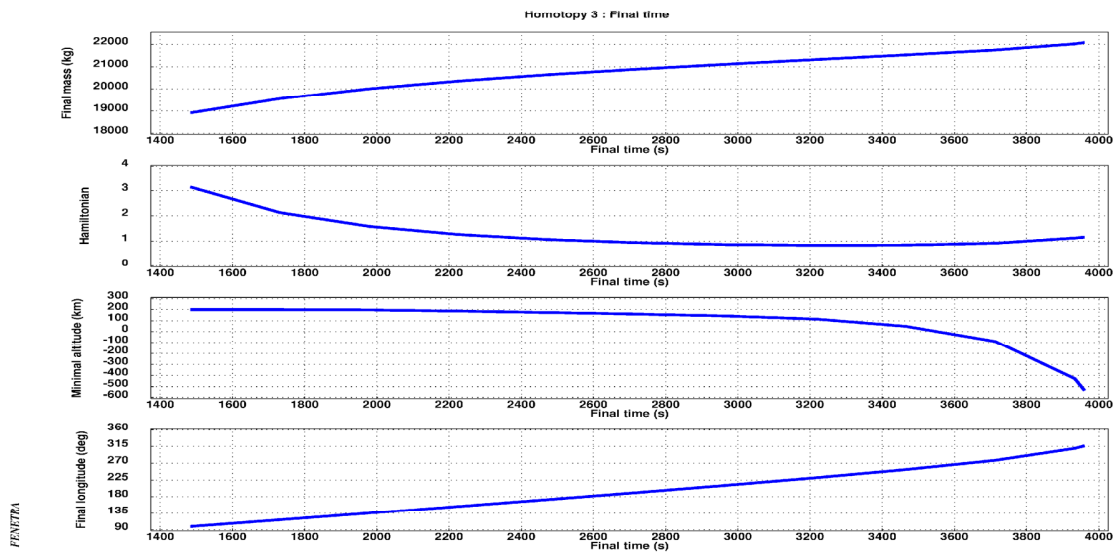


Figure 11 : Final time continuation

The shooting method converges without difficulty until $t_f \approx 3900$ s, but numerical issues arise after that date. This can be explained by looking at the respective values of the final longitude and of the minimum altitude along the trajectory (Figure 11). Increase the final time results in increase the final longitude. When more than a half revolution is allowed, the optimal coast arc consists in passing as close as possible to the Earth center in order to benefit from the gravity acceleration on the downward trajectory leg. Such solutions are not physically feasible, and also induce numerical difficulties in the shooting methods since the gravity acceleration tends to very high values. Moreover the modified coordinate system (78) is not suited to such trajectories making more than a half revolution. In order to improve the numerical integration accuracy and to enhance the convergence of the shooting method, the state and the costate must be moved either to rectangular or to orbital coordinates to pursue the continuation on the final time.

Increasing the final time value yields a significant performance improvement :

- With the modified coordinates, the maximum final mass is 22082 kg for a final time of 3956 s, but the corresponding trajectory crosses the Earth surface with a minimal altitude of -518 km. This solution corresponds to a local maximum. It can be observed that the Hamiltonian passes by a positive local minimum for a lower final time value, around 3300 s.
- Moving to rectangular coordinates, the solution is further improved yielding a final mass of 22103 kg for a final time of 4121 s. The minimal altitude decreases to -700 km.

The best acceptable solution staying above the atmospheric upper limit (conventionally taken at the altitude of 120 km) yields a final mass of 21294 kg for a final time of about 3200 s. The corresponding trajectory makes more than a half revolution (230 deg), which is a usual feature for such upper stages missions. It must be noticed that this solution is not the optimal solution with the minimum altitude constraint, since the constraint is not explicitly taken into account in the solution. It is only the best acceptable solution found along the continuation procedure.

The 2 solutions (with and without the altitude limitation) are presented in the Table 8.

Final time (s)	Final mass (kg)	Final longitude (deg)	Minimal altitude (km)	Hamiltonian	Costate $p_h(t_0)$	Costate $p_{vd}(t_0)$	Costate $p_{vh}(t_0)$	Costate $p_m(t_0)$
3183.2	21294.1	230	120	0.834	6.500	6098.6	1912.1	0.475
4121.4	22103.7	298	-700	1.155	5.905	4995.7	172.5	0.553

Table 8 : Final time continuation

6.2.7 Results recap

For this application example, the computation accuracies are the following :

- the numerical integration is performed with absolute accuracies respectively of 10^{-6} m on the position, and 10^{-9} m/s on the velocity,
- the zeros of the shooting function are required with respective accuracies of 10^{-1} m on the final position and 10^{-4} m/s on the final velocity.

The Table 9 summarizes the successive steps of the continuation procedure.

	Final time (s)	Final mass (kg)	Hamiltonian	Costate $p_h(t_0)$	Costate $p_{vd}(t_0)$	Costate $p_{vh}(t_0)$	Costate $p_m(t_0)$
Initialization	1433.3	1676.2	0	1.000	96.349	1433.314	NA
Renormalization	1433.3	1676.2	0	0.756	72.808	1083.108	-0.137
Gravity homotopy	1483.2	1504.8	0	3.850	69.874	2198.207	-0.236
Dynamics homotopy	1483.2	18915.4	3.149	12.211	6821.530	5225.629	0.310
Final time homotopy	3183.2	21294.1	0.834	6.500	6098.6	1912.1	0.475

Table 9 : Results recap

The total number of trajectory simulations is about 2300 :

- 41 for the initialization (OCP)_{0,0}, solved instantaneously
- 953 for the gravity continuation until (OCP)_{1,0}, achieved in 5s
- 741 for the dynamics continuation until (OCP)_{1,1}, achieved in 4s
- 593 for the final time continuation, achieved in 3s.

The computation is performed on a Sun Solaris OS 5.10 workstation. With an average of 200 simulations per second, the convergence is achieved in about 12 s, without specific code or algorithm optimization. If necessary, the procedure can be significantly sped up by code improvement, especially regarding the time per simulation.

The continuation procedure from the flat Earth model fails on the second test case (§6.1.2) with a geostationary targeted orbit. The altitude difference between the initial and the final orbit is indeed much higher than for the sun-synchronous case. Even if the flat Earth problem theoretically admits a solution whatever the initial and the final conditions, this solution corresponds in this case to a vanishing final mass and an infinite acceleration level. The trajectory integration becomes prone to numerical errors, and specific algorithm enhancements would be necessary to allow the shooting method to converge. Furthermore the difference in terms of performance and of command law between the flat Earth and the round Earth problem makes the continuation procedure questionable in such cases.

6.3 Continuation procedure from the impulsive model

The continuation procedure from the impulsive model is detailed on the first test case (coplanar sun-synchronous targeted orbit). The procedure is then applied successively on the second test case (coplanar geostationary targeted orbit), and on the first case with an additional plane change in order to illustrate the method applicability to a 3-dimensional transfer.

6.3.1 Initialization

Solving the bi-impulse transfer problem (132) yields a final mass of 22099 kg for a final time of 2270 s. The variation of the final mass in function of the transfer time is plotted on the Figure 12.

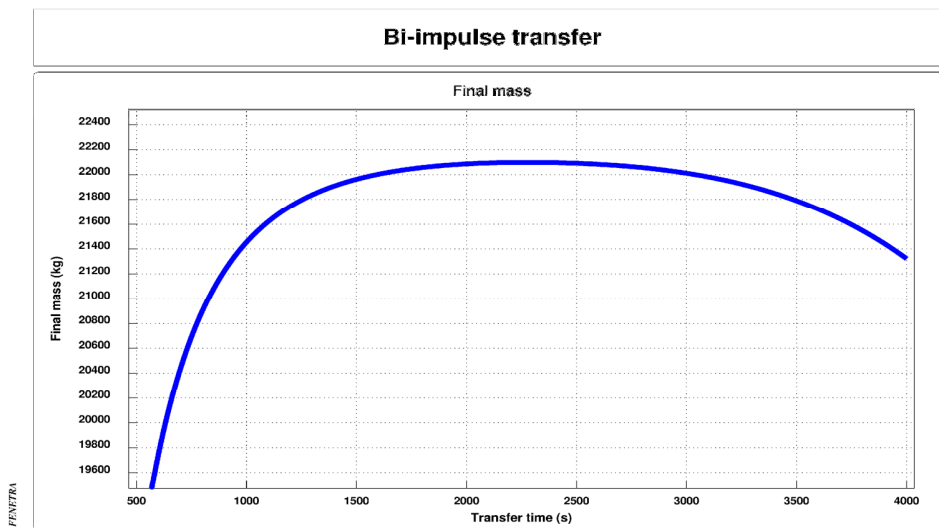


Figure 12 : Bi-impulse transfer performance vs transfer time (SSO)

The curve is relatively flat between 1500 s and 3000 s. This can be explained by the fact that the initial state is close to the apogee of the initial orbit, whereas the final orbit is circular. The optimal solution is a Hohmann-like transfer which is not very sensitive to the injection position and therefore to the transfer time.

The initial state is given in spherical coordinates. The initial costate assessed from the Lagrange multipliers is therefore in the same coordinate system :

$$\begin{cases} p_{\varphi}(t_0) = -\lambda_{\varphi 0} = 0 \\ p_r(t_0) = -\lambda_{r0} = 5.9227 \\ p_v(t_0) = -\lambda_{v0} = 5006.8 \\ p_{\gamma}(t_0) = -\lambda_{\gamma 0} = -9.2033 \\ p_m(t_0) = -\lambda_{m0} = 0.55247 \end{cases}$$

The multiplier $\lambda_{\varphi 0}$ is null since the final orbit is circular and the injection anomaly is free. Change the initial polar angle φ_0 results in a rotation of the whole trajectory without cost change.

For the present example, the analytical assessment (138) with the simplifying assumption of a Hohmann-like transfer gives for the initial costate components P_r , P_v and P_m :

$$\begin{cases} v(t_0^+) = 7952.3 \text{ m/s} \\ v(t_f^-) = 7287.6 \text{ m/s} \\ m_f = 21868.6 \text{ kg} \end{cases} \Rightarrow \begin{cases} p_r(t_0) = 5.8602 \\ p_v(t_0) = 4955.5 \\ p_m(t_0) = 0.54671 \end{cases}$$

The agreement with the numerical solution of the nonlinear problem is fairly good in the present case. The flight path angle costate must nevertheless be still assessed by a numerical method.

6.3.2 Thrust level continuation

The continuation procedure consists in solving a sequence of shooting problems with decreasing thrust levels

- The first shooting problem is initialized with the costate and durations derived from the bi-impulse solution. The thrust level T_{∞} is fixed to a sufficiently high value so that the continuous thrust problem is close to the impulsive problem.
- Each successive shooting problem is initialized with the previous solution, while decreasing the thrust level (parameter α).

The Table 10 presents the intermediate solutions issued by the shooting method on the continuation path for respective thrust levels of 10 and 3 times the actual level of 180 kN. For each thrust level the Table gives :

- the boost and coast durations,
- the initial costate in spherical coordinates (the unities are kg, km and rad)

	Bi-impulse	Thrust level $\times 10$	Thrust level $\times 3$	Thrust level $\times 1$
Thrust level (kN)	Infinite	1800	540	180
Boost 1 duration (s)	0	41.8	139.0	380.8
Coast duration (s)	2269.6	2559.2	3219.5	3682.7
Boost 2 duration (s)	0	2.1	7.3	57.9
Final time (s)	2269.6	2603.1	3365.8	4121.4
Final mass (kg)	22098.96	22098.73	22098.85	22103.65
P_r (kg/km)	5.9227	5.9220	5.9211	5.9049
P_v (kg/(km/s))	5006.8	5006.8	5005.9	4998.7
P_γ (kg/rad)	-9.2033	-9.7160	-10.078	-10.921
P_m (kg/kg)	0.55247	0.55247	0.55249	0.55287

Table 10 : Thrust level continuation (SSO 2-dimensional transfer)

It can be observed that the final mass and the initial costate derived from the bi-impulse solution are very close to the continuous thrust solution with the nominal thrust level, although the durations are quite different. The bi-impulse initialization is in fact sufficiently accurate to bypass the continuation procedure, and the shooting method converges directly for $\alpha=1$. For the present example, this method is therefore very efficient and much more direct than the flat Earth continuation procedure. This has to be confirmed by more exhaustive tests cases.

The solution is identical to the one obtained by the flat Earth continuation procedure (Table 8) without minimal altitude constraint. Indeed applying the coordinate change (112) on the costate components we obtain :

$$\begin{cases} p_\phi(t_0) = 0 \\ p_r(t_0) = 5.9049 \\ p_v(t_0) = 4998.7 \\ p_\gamma(t_0) = -10.921 \end{cases} \Rightarrow \begin{cases} p_d(t_0) = 0 \\ p_h(t_0) = 5.9049 \\ p_{vd}(t_0) = 4995.7 \\ p_{vh}(t_0) = 172.5 \end{cases}$$

The trajectories corresponding to the successive thrust levels are plotted on the Figure 13. As mentioned in §6.2.6, it would be necessary to take into account a constraint on the minimal altitude in order to obtain realistic trajectories.

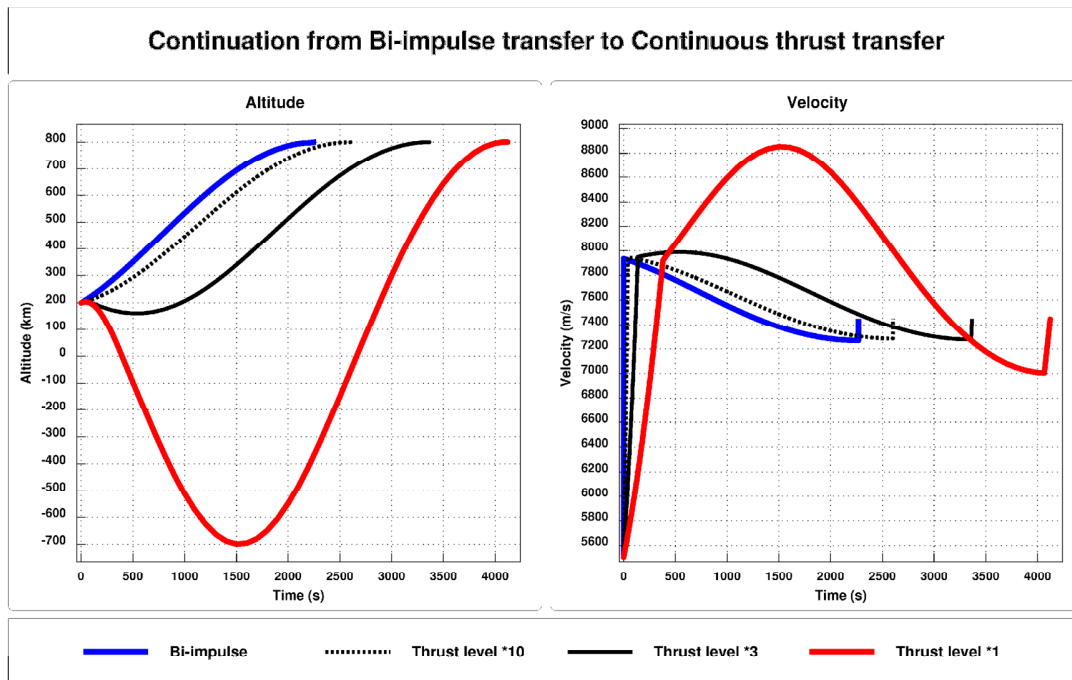


Figure 13 : Optimal trajectories vs thrust level (SSO)

6.3.3 Solution analysis

It could seem surprising to obtain a better performance with the finite thrust level of 180 kN than with the impulse modeling. This performance improvement can be explained by assessing the velocity losses during the thrust maneuvers, for the different thrust levels.

The Table 11 presents for each boost and for the different thrust levels :

- the impulsive velocity assessed by the Tsiolkovsky formula (113),
- the gravity and incidence losses.

		Bi-impulse	Thrust level × 10	Thrust level × 3	Thrust level × 1
	Thrust level (kN)	Infinite	1800	540	180
Boost 1	Duration (s)	0	41.8	139.0	380.8
	Impulse (m/s)	2435.0	2451.4	2442.9	2169.2
	Gravity loss (m/s)	0	8.0	-5.2	-252.0
	Incidence loss (m/s)	0	0.2	0.1	2.1
Boost 2	Duration (s)	0	2.1	7.3	57.9
	Impulse (m/s)	183.4	167.1	175.5	448.3
	Gravity loss (m/s)	0	0.0	0.0	0.6
	Incidence loss (m/s)	0	0.0	0.0	0.0

Table 11 : Velocity losses (SSO 2-dimensional transfer)

The velocity losses arise when the velocity increments are not delivered instantaneously²⁵. They represent the additional variation of the vehicle absolute velocity modulus due to the gravity and the incidence during a continuous thrust maneuver from t_{ini} to t_{fin} .

- The gravity loss is defined by : $\Delta V_{grav} = \int_{t_{ini}}^{t_{fin}} g \sin \gamma dt$ where γ is the flight path angle. It is positive on an upwards leg, negative on a downward leg.
- The incidence loss is defined by : $\Delta V_{inci} = \int_{t_{ini}}^{t_{fin}} \frac{T}{m} (1 - \cos i) dt$ where i is the angle between the thrust and the velocity direction. It is always positive.

The incidence loss is an indicator of the command law optimality, and it should be as near to zero as possible. A significant incidence loss on the optimal trajectory may be caused either by path constraints or by an insufficient thrust level.

By definition, the velocity losses are null with the impulsive modeling. The first boost is oriented upwards in order to raise the apogee so that the coast arc crosses the targeted orbit.

The high thrust solution (10×T) is close to the impulsive solution, with short boost durations. The first boost is delivered in 42 s and it incurs a 8 m/s penalty due to the gravity. This results in a slight cost decrease as expected.

Decreasing the thrust level further (3×T, 1×T) becomes favorable since it gives enough time to perform an initial downward leg during the first boost. On this downward leg the gravity acts positively regarding the velocity increase. The command law remains optimal as indicated by the incidence loss close to zero.

For the present example, an unrealistic trajectory is obtained since no minimal altitude constraint is imposed. This path constraint would limit the downward maneuver and downgrade the performance, on the one hand by reducing the gravity velocity gain, on the other hand by inducing higher incidence losses.

6.3.4 Geostationary case

On the geostationary case (§6.1.2) the bi-impulse procedure still yields a sufficiently good costate initialization, allowing the convergence of the shooting method. The costate guess is compared to the shooting method solution in the Table 12.

	Bi-impulse	Thrust level $\times 1$
Thrust level (kN)	Infinite	65
Boost 1 duration (s)	0	13.6
Coast duration (s)	29425.1	17934.0
Boost 2 duration (s)	0	173.6
Final time (s)	29425.	18121.2
Final mass (kg)	5561.9	6454.1
P_r (kg/km)	0.8879	0.9818
P_v (kg/(km/s))	1606.4	1874.2
P_γ (kg/rad)	-28.5817	-25.9051
P_m (kg/kg)	0.55619	0.64541

Table 12 : Initial costate guess (GEO)

6.4 3-dimensional transfer

The bi-impulse problem formulation (132) and the solution method (§5.1.2) apply identically to the 3-dimensional case. Indeed the Lambert transfer occurs in the plane defined by the initial and the final positions, whatever the initial and final orbital planes. The plane changes are performed at the endpoints by the impulse assessments (124) after solving the Lambert problem.

The initial costate guess is derived from the Lagrange multipliers associated to the 7 components of the initial state. The costate components correspond to the initial state in spherical coordinates, denoting :

- ψ the angle between the initial position and the targeted orbital plane (latitude)
- χ the angle between the initial velocity and the targeted orbital plane (azimuth)

The first test case (SSO) is retrieved, with an additional plane change requirement of 5 deg between the initial and the final orbital plane, assumed to be equatorial (inclination = 0 deg). Two cases are considered by moving, either the initial velocity orientation (angle $\chi = 5$ deg), or the initial position (angle $\psi = 5$ deg) out of the targeted orbital plane.

A continuation on the thrust level is then started considering an initial high level and progressively reducing it until the actual one. Opposite to the 2-dimensional case, some costate components are highly sensitive to the thrust level. In order to make the shooting method converge from the bi-impulse solution, it is necessary to set the initial thrust level to 40 times the actual one.

6.4.1 Out of plane initial velocity

The initial conditions for this 3-dimensional test case are identical to §6.1.1 with :

- $\psi = 0$ deg : the initial position is in the targeted orbital plane.
- $\chi = 5$ deg : the initial velocity makes an angle of 5 deg with the targeted orbital plane.

This corresponds to an initial orbit with an inclination of 5 deg and a perigee argument of -178 deg. The other orbital parameters are unchanged (apogee 204 km, perigee -4194 km, anomaly 178 deg).

The Table 13 presents the intermediate solutions issued by the shooting method on the continuation path for respective thrust levels corresponding to 40 and 10 times the actual level of 180 kN.

	Bi-impulse	Thrust level $\times 40$	Thrust level $\times 10$	Thrust level $\times 1$
Thrust level (kN)	Infinite	7200	1800	180
Boost 1 duration (s)	0	10.6	42.6	384.3
Coast duration (s)	2296.8	2450.2	2533.8	3675.6
Boost 2 duration (s)	0	0.5	2.2	64.0
Final time (s)	2296.8	2461.3	2578.4	4123.9
Final mass (kg)	21765.07	21781.82	21780.91	21716.7
P_r (kg/km)	5.7918	5.7329	5.7378	5.7701
P_ϕ (kg/rad)	0	0	0	0
P_ψ (kg/rad)	111.207	107.234	-61.044	-70.085
P_v (kg/(km/s))	4739.0	4752.4	4767.3	4647.2
P_γ (kg/rad)	-17.218	-16.706	-6.750	-16.197
P_χ (kg/rad)	124.627	125.609	124.670	149.761
P_m (kg/kg)	0.54413	0.54466	0.54431	0.54360

Table 13 : Thrust level continuation (out of plane velocity)

With 40 times the nominal thrust level, the shooting method solution is very close to the bi-impulse initialization. When the thrust level is reduced, the costate component P_ψ related to the out of plane position evolves drastically and it even changes sign, making the continuation procedure more sensitive than in the 2-dimensional case. In this 3-dimensional case the shooting method does not converge starting with 10 times the nominal thrust level and the continuation procedure is necessary to come back to the initial problem.

The other costate components and the final mass remain correctly estimated by the bi-impulse modeling.

The Table 14 presents for each boost and for the different thrust levels the impulsive velocity and the gravity and incidence losses.

		Bi-impulse	Thrust level $\times 40$	Thrust level $\times 10$	Thrust level $\times 1$
	Thrust level (kN)	Infinite	7200	1800	180
Boost 1	Impulse (m/s)	2504.0	2509.7	2510.6	2194.7
	Gravity loss (m/s)	0	2.6	8.5	-272.8
	Incidence loss (m/s)	0	63.6	59.9	66.2
Boost 2	Impulse (m/s)	181.6	173.8	172.0	500.7
	Gravity loss (m/s)	0	0.0	0.0	0.5
	Incidence loss (m/s)	0	1.0	4.1	12.3

Table 14 : Velocity losses (out of plane velocity)

Compared to the 2-dimensional case, the plane change requires out of plane thrusting mainly during the first boost. This induces an incidence loss and the final mass is therefore 387 kg lower than in the coplanar case. The sensitivity of the performance to the plane change is revealed by the high values of the costate components P_ψ and P_χ .

6.4.2 Out of plane initial position

The initial conditions for this 3-dimensional test case are identical to §6.1.1 with :

- $\psi = 5$ deg : the initial position is 5 deg outwards the targeted orbital plane.
- $\chi = 0$ deg : the initial velocity is parallel to the targeted orbital plane.

This corresponds to an initial orbit with an inclination of 5 deg and a perigee argument of -88 deg. The other orbital parameters are unchanged (apogee 204 km, perigee -4194 km, anomaly 178 deg).

The Table 15 presents the intermediate solutions issued by the shooting method on the continuation path for respective thrust levels corresponding to 40 and 10 times the actual level of 180 kN.

	Bi-impulse	Thrust level $\times 40$	Thrust level $\times 10$	Thrust level $\times 1$
Thrust level (kN)	Infinite	7200	1800	180
Boost 1 duration (s)	0	10.4	41.2	377.5
Coast duration (s)	1425.5	3885.0	3862.0	3515.1
Boost 2 duration (s)	0	2.1	8.4	99.8
Final time (s)	1425.5	3897.4	3911.6	3992.4
Final mass (kg)	20092.70	19639.36	19752.17	20531.1
P_r (kg/km)	4.9839	5.0327	4.9305	4.2308
P_ϕ (kg/rad)	0	0	0	0
P_ψ (kg/rad)	-501.398	-511.566	-510.445	-475.626
P_v (kg/(km/s))	4454.0	4134.1	4117.5	4305.8
P_γ (kg/rad)	-90.027	-150.313	-139.529	-47.431
P_λ (kg/rad)	8.729	5.055	8.0875	25.6049
P_m (kg/kg)	0.50232	0.49196	0.49741	0.52351

Table 15 : Thrust level continuation (out of plane position)

Compared to the previous case of an out of plane velocity (§6.4.1), the costate component P_ψ related to the out of plane position is this time well guessed and stable. But the other costate components and the final mass estimated by the bi-impulse modeling are farther from the shooting method solution even for the high thrust level of 40 times the real one.

The optimal solution, particularly the costate component P_γ related to the initial flight path angle, proves very sensitive to the thrust level, making the whole procedure less efficient than in the previous case. This sensitivity is intrinsic to the problem instance. It appears indeed when trying to assess the derivatives of the nonlinear problem (132) by finite differences using different increments (§5.2.1). In such a case, care must be taken on these derivatives assessment whatever the numerical method envisioned (finite differences or Lagrange multipliers).

The Table 16 presents for each boost and for the different thrust levels the impulsive velocity and the gravity and incidence losses.

		Bi-impulse	Thrust level $\times 40$	Thrust level $\times 10$	Thrust level $\times 1$
	Thrust level (kN)	Infinite	7200	1800	180
Boost 1	Impulse (m/s)	2303.2	2434.8	2405.1	2144.7
	Gravity loss (m/s)	0	-2.3	-10.7	-281.6
	Incidence loss (m/s)	0	108.5	90.0	25.0
Boost 2	Impulse (m/s)	735.2	706.0	708.9	798.5
	Gravity loss (m/s)	0	0.0	0.0	9.0
	Incidence loss (m/s)	0	402.5	395.3	301.3

Table 16 : Velocity losses (out of plane position)

Compared with the previous case the incidence losses are much higher during the second boost, in order to perform the plane change when the vehicle crosses the targeted orbit. The bi-impulse solution is consequently less representative of the continuous thrust solution, even with a high thrust level. This may explain the less good accuracy of the initial costate guess. In view of a systematic solution process, algorithmic enhancements would be necessary both for the initial assessment and for the continuation procedure.

7 Conclusion

Various methods exist to tackle the problem of the minimal consumption orbit transfer for a launcher upper stage. Direct methods are easier to handle at the initialization stage, but an accurate convergence is time consuming due to the large size of the NLP problem resulting from the discretization. On the other hand indirect methods are based on the Newton method to solve the shooting problem derived from the PMP, and thus benefit from its properties : fast convergence and high accuracy. Their major issue lies in the costate initialization which has to be quite accurate to allow the convergence of the Newton method. Various initialization algorithms have been proposed for indirect methods, applied either to nearly circular transfers¹⁰ or to low-thrust orbit transfers¹³.

Two alternative methods are proposed here in order to overcome the initialization issue and solve the problem of the coplanar orbit transfer at minimum consumption without any a priori knowledge on the optimal solution. They are based on model simplifications followed by continuation procedures to come back to the real problem. These methods constitute interesting complements to the existing ones for high-thrust orbit transfers starting from elliptical initial conditions. They are thus well suited to the solution of an upper stage flight.

7.1 Methods proposed

The methods proposed rely on model simplifications that allow a significant problem reduction. The two simplified models envisioned are respectively :

- The flat Earth model
- The impulsive model

In each case, an efficient algorithm can be set up that solves the simplified problem without specific initialization task. The model is then parameterized in order to perform a continuation procedure coming back to the real model :

- The flat Earth model is modified by adding terms that make it diffeomorphic up to a coordinate change to the ‘real’ round Earth model. The continuation procedure introduces continuously these additive terms in the model.
- The impulsive model is approximated by a very high (“infinite”) thrust level. The continuation procedure decreases progressively the thrust level until recovering the vehicle nominal level.

From the algorithmic point of view, the procedures consist in solving a series of shooting problems, starting from the simplified problem, whose solution can be found automatically, and ending up with the real problem. These procedures are time-efficient and provide a way for bypassing the usual initialization issues of the shooting method when applied directly to the real problem. Regarding the practical application, these algorithms are well-suited for a launcher upper stage using a high-thrust engine. Although no exhaustive tests were performed, the algorithms seem quite robust and efficient.

7.2 Current status

7.2.1 Flat Earth model

Convergence issues may occur when solving the flat Earth problem. The performance with the flat Earth model is indeed far worse than with the round Earth model, since the free altitude gain due to the Earth curvature is cancelled (see Figure 6). This is illustrated clearly by the application example of §6. If the final orbit is too high with respect to the initial orbit, the final mass may tend to vanish, causing numerical issues in the shooting method. A possible way to overcome this difficulty would be to perform the whole continuation with a lowered gravity field, and once the round Earth model completed to continuously increase this gravity field until recovering the true value.

Another difficulty may be encountered at the initialization step if the flat Earth problem solution consists of a single thrust arc. Algorithm 1 should be modified to cope with this case, for example by stopping the numerical integration as soon as one at least of the final conditions is met (orbit altitude or horizontal velocity). In that case, the solution yielded by the Algorithm 1 will not be optimal for the Earth problem, but it can still be used as starting point for the continuation procedure.

7.2.2 Impulsive model

The continuation procedure starting from the bi-impulse model seems more promising in view of an automatic solver for the following reasons :

- The modified coordinates used for the flat Earth procedure become unsuited to long trajectories making more than a half Earth revolution. On the opposite no specific coordinate system is required for the bi-impulse procedure. The dynamics equations remain the same along the continuation.
- The impulsive problem is “closer” to the real one than the flat Earth problem whose solution has only one boost. This may question the systematic success of the continuation from the flat to the round Earth model. Also less continuation stages are required starting from the bi-impulse solution, and one single shooting problem may even be sufficient as in the coplanar example. This results in computation time gain.
- The extension of the flat Earth procedure to 3-dimensional transfers or to elliptical final conditions requires further algorithmic developments and additional continuation stages. On the opposite, the bi-impulse procedure directly applies whatever the initial and final conditions provided that the thrust level continuation is enhanced to account for stiff costate changes, as illustrated by the example of §6.

7.3 Perspectives

Many questions remain open and from this point of view the present work should be considered as preliminary. A first question is to investigate whether the procedures proposed are systematically efficient, for any possible coplanar orbit transfer. Up to now no exhaustive tests were performed. It is very probable that some difficulties will be met, as in any continuation process, due to the intricate topology of the space of possible continuation paths, this space being not always arc-wise connected. Other perspectives are to enhance the procedures when some orbital

parameters are free, including the initial anomaly, in order not to over-constrain the problem, and to extend systematically the methods to the three-dimensional case, whatever the initial and final orbits. The continuation from the impulsive model seems particularly promising in view of solving this general orbit transfer.

Finally it should be possible to improve the zero path following, for example by differential homotopy methods, with the goal of reducing significantly the computation time in view of an industrial tool.

References

1. B. Bonnard, L. Faubourg, E. Trélat, *Mécanique céleste et contrôle des véhicules spatiaux*, Springer, Collection Mathématiques et Applications, 2006, 276 pages. E. Minieka (1974).
2. J.P. Marec, *Optimal Space trajectories*, Elsevier (1979).
3. D.S. Naidu, J.L. Hibey, C. Charalambous, *Fuel-Optimal Trajectories for Aeroassisted Coplanar Orbital Transfer Problem*, IEEE Transactions on Aerospace and Electronic Systems **26** no.2 (March 1990) 374-381.
4. L.W. Neustadt, *A general theory of minimum-fuel space trajectories*, SIAM Journal on Control **3**, no. 2 (1965), 317--356.
5. N.X. Vinh, S.H. Huo, *Optimal time-free nodal transfers between elliptical orbits*, Acta Astronautica **17** (1988), no. 8, 875--880.
6. G. Leitmann, *On a class of variational problems in rocket flight*, Journal of Aerospace Sci. **26**, no. 9, (Sep. 1959), 586-591.
7. J.T. Betts, *Practical methods for optimal control and estimation using nonlinear programming*, Second edition, Advances in Design and Control, 19, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2010.
8. L. Pontryagin, V. Boltyanskii, R. Gramkrelidze, E. Mischenko, *The mathematical theory of optimal processes*, Wiley Interscience, 1962.
9. N. Bérend, F. Bonnans, M. Haddou, J. Laurent-Varin, C. Talbot, *An Interior-Point Approach to Trajectory Optimization*, Journal of Guidance, Control and Dynamics **30** (2007), no. 5, 1228--1238.
10. P. Augros, R. Delage, L. Perrot, *Computation of optimal coplanar orbit transfers*, AIAA 1999.
11. J. Gergaud, T. Haberkorn, *Orbital transfer : some links between the low-thrust and the impulse cases*, Acta Astronautica, **60**, no. 6-9 (2007), 649-657.
12. H.J. Oberle, K. Taubert. (1997), *Existence and multiple solutions of the minimum-fuel orbit transfer problem*, J. Optim Theory Appl. **95** , 243-262.
13. J. Gergaud, T. Haberkorn, P. Martinon, *Low thrust minimum fuel orbital transfer : an homotopic approach*, J. Guidance Cont. Dyn. **27**, 6 (2004), 1046-1060.
14. P. Martinon, J. Gergaud, *Using switching detection and variational equations for the shooting method*, Optimal Cont. Appl. Methods **28**, no. 2 (2007), 95--116.
15. J.D. Thorne, C.D. Hall, *Minimum-Time Continuous Thrust Orbit Transfers*, Journal of the Astronautical Sciences **47** (1997), no. 4, 411-432.
16. T. Dargent, V. Martinot, *An integrated tool for low thrust optimal control orbit transfers in interplanetary trajectories*, Proceedings of the 18th International Symposium on Space Flight Dynamics (ESA SP-548), 2004, 6 pages.
17. R.H. Battin, *An introduction to the Mathematics and Methods of Astrodynamics*, Revised Edition, AIAA Education Series, 1999.
18. B. Bonnard, J.-B. Caillau, E. Trélat, *Geometric optimal control of elliptic Keplerian orbits*, Discrete Cont. Dyn. Syst. **5** (2005), no. 4, 929--956.
19. J.-B. Caillau, J. Noailles, *Coplanar control of a satellite around the Earth*, ESAIM Cont. Optim. Calc. Var. **6** (2001), 239--258.
20. E. L. Allgower, K. Georg, *Numerical Continuation Method. An Introduction*, Springer-Verlag, Berlin (1990).
21. J. Stoer, R. Bulirsch, *Introduction to Numerical Analysis*, Springer-Verlag, Berlin-Heidelberg-New York, 1983.
22. B. Abdesselam, *Introduction à la mécanique analytique*, Ellipses, 2007.
23. A. Agrachev, Y. Sachkov, *Control theory from the geometric viewpoint*, Encyclopaedia of Mathematical Sciences **87**, Control Theory and Optimization, II, Springer-Verlag, Berlin, 2004.
24. B. Schutz, *Geometrical methods of mathematical physics*, Cambridge University Press, 1980.
25. V. Chobotov, *Orbital mechanics - Third Edition*, AIAA Education Series (2002).
26. R.A. Broucke, A.F.B.A. Prado, *Optimal N-impulse transfer between coplanar orbits*, AAS/AIAA Astrodynamics Specialist Conference, Victoria, Canada, 1994.
27. D.F. Lawden, *Optimal transfer between coplanar elliptical orbits*, J. of Guidance Control and Dynamics, vol. 15, no. 3, pp. 788-791, 1991.
28. M. Minoux, *Programmation mathématique - Théorie et algorithmes – 2ème Edition*, Editions Tec&Doc – Lavoisier (2008).

Part 2

Multiple Space Debris Collecting Mission

Debris Selection and Trajectory Optimization

Abstract This document investigates the cost requirement for a Space Debris Collecting mission aiming at removing heavy debris from low Earth orbits. The problem mixes combinatorial optimization to select the debris among a list of candidates and continuous optimization to define the orbital manoeuvres. The solution methodology proceeds in two steps : firstly a specific transfer strategy with impulsive manoeuvres is defined so that the problem becomes of finite dimension, secondly the problem is linearized around an initial reference solution. A Branch and Bound algorithm is then applied iteratively to optimize simultaneously the debris selection and the orbital manoeuvres, yielding a new reference solution. The method is exemplified on a representative application case.

Keywords

Space debris, Orbital mechanics, Branch and Bound, Linear Programming

Acronyms

PMP Pontryagin Maximum Principle
NLP Non-Linear Programming
MINLP Mixed Integer Non Linear Programming
MILP Mixed Integer Linear Programming
LEO : Low Earth Orbit
SSO Sun-Synchronous Orbit
GTO : Geostationary Earth Orbit
PEO : Polar Earth Orbit
SDC : Space Debris Collecting
TSP : Travelling Salesman Problem
LP : Linear Programming
BB : Branch and Bound
AN : Ascending Node
RAAN : Right Ascension of the Ascending Node

Summary

1	Introduction	113
1.1	Space debris	113
1.2	Mission definition	113
1.3	Global optimization problem	113
1.4	Solution methodology for the SDC problem	116
2	Problem formulation	118
2.1	Transfer problem	118
2.2	Path problem	119
2.3	Space Debris Collecting problem	122
3	Transfer strategy	125
3.1	Debris orbits	125
3.2	Direct strategy	128
3.3	Drift strategy	128
3.4	Duration constraint	130
3.5	Thrust level	130
3.6	Selected transfer strategy	131
3.7	Problem formulation with the selected transfer strategy	133
3.8	Further simplifications	134
4	Practical solution	138
4.1	Problem linearization	138
4.2	Transfer cost and duration linearization	138
4.3	Variable product linearization	141
4.4	Linearized problem	141
4.5	Initialization	142
4.6	Iteration process	144
4.7	Post-optimization	146
4.8	Algorithms and software	146
4.9	Assumptions and simplifications recap	149
4.10	Current status and further work	150
5	Application example	152
5.1	Debris orbits	152
5.2	Initialization	153
5.3	Iterations	154

5.4 Optimal solution	154
5.5 Computation time	158
6 Conclusion.....	159

List of Figures

Figure 1 : Solution methodology	117
Figure 2 : The Travelling Salesman Problem	119
Figure 3 : Examples of non feasible paths.....	121
Figure 4 : Trajectories between debris.....	122
Figure 5 : Orbital parameters	125
Figure 6 : Orbit precession due to the Earth flattening.....	126
Figure 7 : Sun-Synchronous Orbit.....	127
Figure 8 : Forwards or backwards correction	129
Figure 9 : Transfer manoeuvres and successive orbits	132
Figure 10 : Simultaneous inclination and shape change	133
Figure 11 : Elliptical vs. circular drift orbit	135
Figure 12 : Optimal inclination change opportunity.....	136
Figure 13 : Linearization of the duration function.....	139
Figure 14 : Duration function approximation with two half segments	140
Figure 15 : Iteration process	144
Figure 16 : Depth and Breath search strategies	148
Figure 17 : Optimal path orbits.....	156
Figure 18 : RAAN correction	157
Figure 19 : Mission cost before and after optimization	157

List of Tables

Table 1 : Embedded optimization problems	114
Table 2 : RAAN precession rates for near SSO debris	127
Table 3 : Cost and duration contributors	150
Table 4 : List of 11 candidate debris.....	152
Table 5 : Initial path.....	153
Table 6 : Iterations.....	154
Table 7 : Optimal path.....	155

Variable nomenclature

Vehicle			
X(t)	State vector (position + velocity + mass)	Real	7
m(t)	Mass	Real	1
Y(t)	State vector (position + velocity)	Real	6
U(t)	Command vector	Real	3

Mission			
N	Number of candidate debris	Integer	1
n	Number of debris to deorbit	Integer	1
T_{deorb}	Deorbitation operations duration	Real	1
T_{min}	Transfer minimum duration	Real	1
T_{max}	Mission maximum duration	Real	1
C	Mission cost	Real	1
T	Mission duration	Real	1
C_L	Mission linearized cost	Real	1
T_L	Mission linearized duration	Real	1

Orbits			
R_T	Earth equatorial radius (= 6378137 m)	Real	1
μ	Earth gravitational constant (= $3.986 \cdot 10^{14} \text{ m}^3/\text{s}^2$)	Real	1
J_2	First zonal coefficient (= $1.086 \cdot 10^{-3}$)	Real	1
C_{J_2}	$C_{J_2} = \frac{3}{2} J_2 \sqrt{\mu} R_T^2 = 1.31810^{18}$	Real	1
a	Semi major axis	Real	1
e	Eccentricity	Real	1
I	Inclination	Real	1
Ω	Right ascension of the ascending node	Real	1
ω	Argument of the perigee	Real	1
v	True anomaly	Real	1
L	Longitude from the ascending node	Real	1
n	Mean motion	Real	1

$\dot{\Omega}$	RAAN precession rate	Real	1
H	Altitude	Real	1
V	Speed	Real	1
ΔV	Impulsive speed	Real	1
ΔV_P	Perigee impulsive speed (Hohmann transfer)	Real	1
ΔV_A	Apogee impulsive speed (Hohmann transfer)	Real	1

Debris k, k=1 to N			
S_k	Debris selection	Binary	1
X_k	Number of arriving edges	Binary	1
Y_k	Number of departing edges	Binary	1
Z_k	Product of x_k and y_k	Binary	1
t_k^a	Arrival date	Real	1
t_k^d	Departure date	Real	1
C_k	Operations cost	Real	1
T_k	Operations duration	Real	1
a_k	Semi major axis	Real	1
$Y_k(t)$	State vector (position + velocity)	Real	6
e_k	Eccentricity	Real	1
I_k	Inclination	Real	1
Ω_k	Right ascension of the ascending node	Real	1
$\dot{\Omega}_k$	RAAN precession rate	Real	1

Transfer i-j, i,j=1 to N, i≠j			
S_{ij}	Transfer selection	Binary	1
C_{ij}	Transfer cost	Real	1
T_{ij}	Transfer duration	Real	1
t_i	Departure date	Real	1
t_j	Arrival date	Real	1

C_{Lij}	Transfer linearized cost	Real	1
T_{Lij}	Transfer linearized duration	Real	1
a_{ij}	Drift orbit semi major axis	Real	1
e_{ij}	Drift orbit eccentricity	Real	1
I_{ij}	Drift orbit inclination	Real	1
$\dot{\Omega}_{ij}$	Drift orbit RAAN precession rate	Real	1
ΔV_{ij}	Transfer impulsive speed	Real	1
α_{ij}	Semi major axis difference wrt reference	Real	1
τ_i	Departure date difference wrt reference	Real	1
p_{ij}	Product of s_{ij} and a_{ij}	Real	1
q_{ij}	Product of s_{ij} and t_i	Real	1
a_{min}, a_{min}	Bounds on semi major axis	Real	1
$\alpha_{min}, \alpha_{min}$	Bounds on semi major axis difference	Real	1
τ_{min}, τ_{min}	Bounds on departure date difference	Real	1

1 Introduction

1.1 Space debris

A drastic growth of the space debris population in Low Earth Orbits under 2000 km is foreseen in the next decades with high collision risks for future space flights^{1,2}. A particularly crowded region is the vicinity of the Sun-Synchronous Orbits and Polar Orbits in the range 600-1200 km altitude and 80-105 deg inclinations with a density peak around the altitudes of 800-900 km. These orbits are well adapted to Earth observation and are therefore intensively used. The most efficient ways to mitigate this scenario are to avoid the creation of new debris by appropriate spacecraft conception and to actively remove the existing debris^{3,4}. Several studies have led to the conclusion that removing 5 heavy LEO debris per year, like spent satellites or launchers upper stages, is mandatory to stabilize the debris population⁵. This document deals with such a Space Debris Collecting mission that would meet this requirement.

1.2 Mission definition

The Space Debris Collecting mission aims at deorbiting 5 heavy debris per year. The debris must be selected in a list of N candidates, so that the required propellant consumption for the mission is minimized. The deorbitation aims at clearing the LEO region (altitude below 2000 km) : this can be achieved by either making the debris re-enter the atmosphere (preferred solution if not too costly regarding the SDC vehicle size and the fall-out operations) or re-orbiting the debris at a higher altitude³. For the SSO debris considered in this document, only the reentry solution is envisioned. After the debris capture the deorbitation manoeuvre is either performed by the vehicle, or by the debris itself with a deorbitation device (booster, tether, ...) supplied by the vehicle. The mission starts from the first debris selected on the path assuming that the launcher has brought the SDC vehicle to this first debris. The cost and duration to reach this first debris are therefore not counted into the mission cost and duration.

1.3 Global optimization problem

The global SDC problem is a nonlinear time dependent graph problem mixing :

- several continuous transfer problems consisting in optimizing the trajectory from one debris to the other.
- a combinatorial path problem consisting in selecting the debris and the collecting order.

Table 1 gives an overview of these embedded problem features.

	Transfer problem Optimize the trajectories	Path problem Select the debris and the order
Category	Functional optimization / Optimal control	Graphs / NP problem (TSP like)
Complexity	Infinite dimension	Exponential
Algorithms	Nonlinear programming	Integer programming
Problem nature	Nonlinear time dependent graph problem	

Table 1 : Embedded optimization problems

The transfer problem and the path problem are both challenging, even considered separately, and it is therefore not reasonable to tackle the global problem directly. We first review the available numerical methods for functional optimization (find the optimal control law of a dynamical system) and combinatorial optimization (find the optimal path in a graph) with their related issues. A solution methodology is then set up taking into account the SDC problem specificities.

1.3.1 Functional optimization

Functional optimization deals with infinite dimension problems, one at least of the unknowns being a function. The nomenclature ‘optimal control problem’ refers to a functional optimization problem where the unknown function is the command law of a controlled dynamical system. For such problems, even simple, there are generally no analytic solutions and numerical solutions must be found with iterative algorithms. The numerical methods can be divided into two main categories^{6,7} :

- Direct methods discretize the command law with time steps. The functional problem is transformed into a large size finite dimension problem for which Nonlinear Programming (NLP) algorithms can be applied. The direct methods are relatively easy to initialize, but the convergence is generally slow and the results only approximate the solution to the functional problem.
- Indirect methods try to solve the infinite dimension problem using the Pontryagin Maximum Principle (PMP). A costate vector of the same dimension as the state vector is introduced, which obeys the Euler-Lagrange equations. The problem reduces to finding the initial values of the costate vector in order to match the optimality necessary conditions. A shooting method with a correct initialization can be used to solve these nonlinear equations. The indirect methods are very difficult to initialize, but the convergence is generally quick and accurate.

1.3.2 Combinatorial optimization

Combinatorial optimization deals with problems in which all or part of the unknowns are integer variables. The classification of combinatorial problems lies on the existence or not of polynomial-time algorithms, on one hand to

solve the problem, on the other hand to check a solution⁸. The P-class includes problems with a known polynomial-time solution algorithm (the same algorithm can indeed be used to check the solution). The NP-class includes problems with a known polynomial-time checking algorithm, but no known polynomial-time solution algorithm. The set of NP-complete problems denoted NPC is formed by all NP problems that can be transformed to each other by a polynomial-time algorithm. If one day a polynomial-time solution algorithm was found for one problem of NPC, then any problem of NPC could also be solved polynomially and we would have NP=P.

The TSP and all its variants are NPC. For these hard combinatorial problems, the solution methods fall into three main categories^{8,9} :

- Explicit enumeration of all the possible combinations ensures to find the exact solution i.e. the global optimum. The total number of arrangements of n debris amongst N is equal to $A_N^n = N! / (N - n)!$. Taking into account the constraints leads to a formulation of the problem with large numbers of binary variables. With m binary variables, the total number of combinations amounts to 2^m . The explicit enumeration is practically restricted only to small size problems with a few tens of variables. For m=50, the enumeration would take one year with a computer assessing one billion combinations per second.
- Implicit enumeration (Branch and Bound, Branch and Cut) consists in exploring the tree of all possible combinations with branches cut-off during the exploration^{10,11}. The cut-off is based on an assessment of the best potential solution contained in a branch, and the comparison with a reference admissible solution. Like explicit enumeration, implicit enumeration yields the exact solution. The method efficiency is highly dependent on the branching strategy. Also the computation time grows exponentially with the problem size. These methods are therefore applicable to medium size problems (a few hundred variables).
- Approximate solutions can be found by stochastic programming methods like genetic algorithms, tabu search, simulated annealing...^{9,12}. These methods limit a priori the number of combinations assessed and thus the computation time. The combinations space is explored using an oriented random strategy. If the best solution found is judged unsatisfying, the search may be resumed with new tunings in order to improve the exploration strategy. Opposite to continuous problems there exists generally no sufficient conditions for discrete problems that guarantee the local optimality of the solution or its difference with the true global optimum. These methods are the only ones applicable for large size problems (more than thousands of variables).

1.3.3 Mixed integer nonlinear programming

The global SDC problem comes in the field of mixed integer nonlinear programming (MINLP). MINLP deals with finite dimension optimization problems mixing integer and real variables with nonlinear cost and constraints. A survey of existing MINLP solvers is presented in Ref¹³ and free MINLP software packages are available at the link¹⁴. Most solution algorithms for nonlinear problems consist in a local linearization around an initial reference point yielding a mixed integer linear programming problem (MILP). The solution of the MILP subproblem is carried out generally by a Branch and Bound algorithm. The procedure is then iterated after updating the reference point.

A major issue for MINLP problems is the convergence toward the global optimum. In the case of convex cost and constraint functions, the convergence toward the global optimum can be guaranteed, for example by procedures of outer convexification and step bounding. Such procedures implemented in several existing softwares consist in adding successive tangential linearizations as inequality constraints to the MILP subproblems while using nonlinear relaxations to bound the step¹⁴.

1.4 Solution methodology for the SDC problem

The SDC problem features are :

- The medium size : typically 5 debris must be selected amongst 10. The number of variables and related constraints required to formulate the path problem ranges between 100 and 1000. This medium size favours the choice of an implicit enumeration method like Branch and Bound in order to get the exact solution. For larger numbers of debris, it would be necessary to turn towards stochastic methods.
- The difficulty due to the infinite dimension embedded control problems, which are themselves intrinsically hard in the general case. The cost of going from any debris to the other depends on the starting date and on the allocated duration. The mission overall duration being constrained to less than one year, there is a strong coupling between the optimal control laws of the successive transfers from one debris to the other.

In view of a Branch and Bound algorithm, a linear formulation of the problem is desirable, in order to ensure the required solution robustness and efficiency. The simplification steps leading to a practical formulation that can be solved by a Branch and Bound algorithm are :

- to define a transfer strategy that restricts the optimal control problem to a reduced size NLP problem
- to formulate the problem with respect to the specific transfer strategy selected (transfer variables and constraints)
- to linearize the resulting formulation around an initial reference solution.

Due to the linearization, an iterative process must be set up in order to update the linearization around the new solution, until the solution stabilizes. The solution methodology depicted on Figure 1 follows the successive steps :

- Write the problem formulation (Chapter 2)
- Simplify the transfer problem considering the mission specificities (Chapter 3)
- Linearize the formulation around an initial pre-optimized solution and then iterate on the local solution until stabilization (Chapter 4)
- Issue the optimal path and reoptimize the controls and dates along the selected path

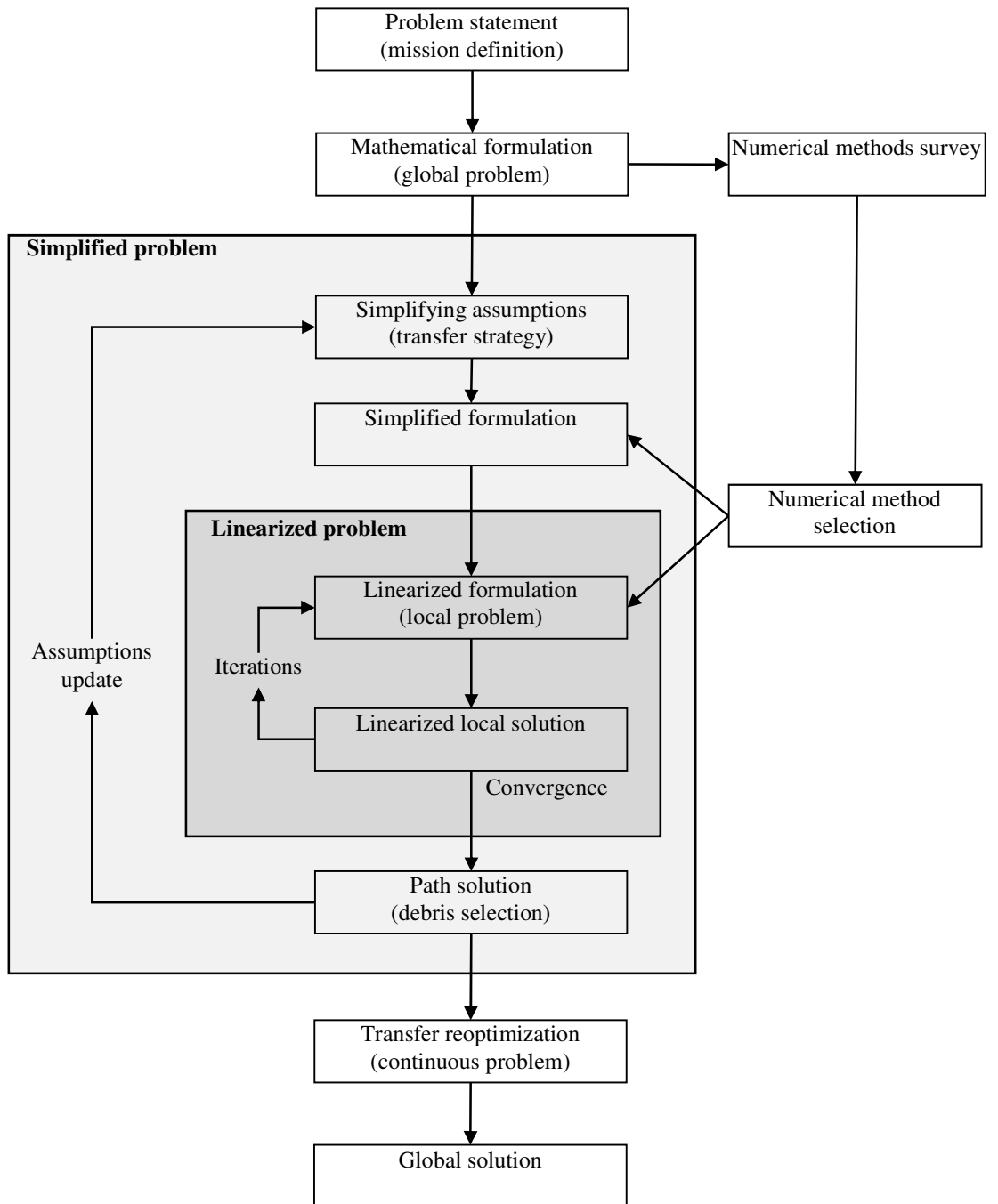


Figure 1 : Solution methodology

2 Problem formulation

This chapter presents the mathematical formulation of the two embedded optimization problems and gives an overview of the available numerical methods.

2.1 Transfer problem

The transfer problem consists in going from a debris i to another debris j . The SDC vehicle is represented by the 7-

$$\text{state vector } X(t) = \begin{pmatrix} \vec{r}(t) = \text{position} \\ \vec{v}(t) = \text{velocity} \\ m(t) = \text{mass} \end{pmatrix} = \begin{pmatrix} Y(t) \\ m(t) \end{pmatrix}.$$

To ease the subsequent formulation of the optimal control problem, we introduce the 6-vector $Y(t)$ representing the 6 first components of $X(t)$, i.e. the 3 position and the 3 velocity components or equivalently the 6 orbital parameters. The 6-state vector of the debris k at the date t is similarly denoted $Y_k(t)$.

The SDC vehicle trajectory is controlled by the 3-command vector $U(t)$ representing the vehicle thrust. Denoting the thrust magnitude by $u(t) = \|U(t)\|$, with $0 \leq u \leq u_{\max}$, the thrust direction by $\vec{d}(t)$ and the burned propellant exhaust velocity by v_e , the vehicle dynamics is represented by the first order ODE :

$$\dot{X}(t) = f[X(t), U(t), t] \Leftrightarrow \begin{cases} \dot{\vec{r}} = \vec{v} \\ \dot{\vec{v}} = \vec{g} + \frac{u\vec{d}}{m} \\ \dot{m} = -\frac{u}{v_e} \end{cases} \quad (1)$$

We denote t_i^d and t_j^a the respective dates of departure from the debris i and arrival to the debris j . The vehicle state must coincide :

- with the debris i state at the transfer beginning : $Y(t_i^d) = Y_i(t_i^d)$
- with the debris j state at the transfer end $Y(t_j^a) = Y_j(t_j^a)$.

The propellant consumption C_{ij} and duration T_{ij} required for this transfer are :

$$\begin{cases} C_{ij}(t_i^d, t_j^a, X(t_i^d), X(t_j^a), [U_{ij}(t), t_i^d \leq t \leq t_j^a]) = \int_{t_i^d}^{t_j^a} \frac{u_{ij}(t)}{v_e} dt \\ T_{ij} = t_j^a - t_i^d \end{cases} \quad (2)$$

The optimal control problem of the transfer from a debris i to a debris j is then formulated as :

$$\text{Min}_{t_i^d, t_j^a, U_{ij}(t)} C_{ij}(t_i^d, t_j^a, X(t_i^d), X(t_j^a), [U_{ij}(t), t_i^d \leq t \leq t_j^a]) \text{ s.t. } \begin{cases} Y(t_i^d) = Y_i(t_i^d) \\ Y(t_j^a) = Y_j(t_j^a) \\ \dot{X}(t) = f[X(t), U_{ij}(t), t] \text{ for } t_i^d \leq t \leq t_j^a \\ T_{ij} = t_j^a - t_i^d \leq T_{ij\max} \end{cases} \quad (3)$$

The upper bound $T_{ij\max}$ on the transfer duration is necessary in order to have a well posed problem^{15,16}. The unknowns are the initial date t_i^d , the final date t_j^a , and the command law $[U_{ij}(t), t_i^d \leq t \leq t_j^a]$ which is a function of the time.

2.2 Path problem

2.2.1 Travelling Salesman Problem and variants

The archetype of path problems is the Travelling Salesman Problem (TSP). The salesman has to visit successively N towns. Every town is linked to any other with the distances being fixed. The goal is to visit each town once and only once while minimizing the total distance covered. The distance (or cost) from any town i to any other town j is C_{ij} and the associated duration is T_{ij} (Figure 2).

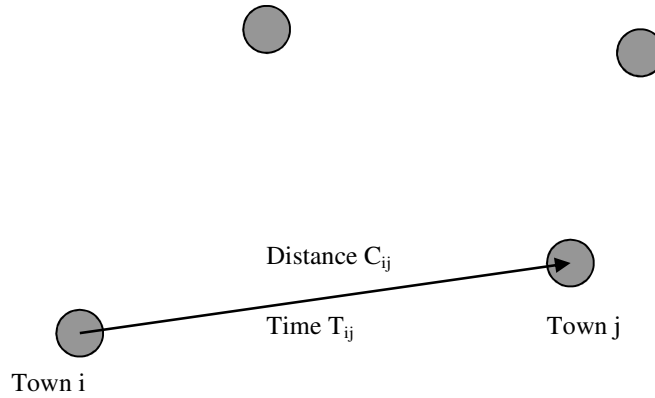


Figure 2 : The Travelling Salesman Problem

In order to formulate the SDC path problem, we start from the classical TSP with the additional features :

- Only n debris among the N must be visited. This TSP variant is the “ n -shortest path problem”^{17,18,19,20}.
- The costs C_{ij} and durations T_{ij} are time-dependent.
- There is a global duration constraint on the solution : $T \leq T_{\max}$

- Processing a debris (capture, deorbitation, release) requires a minimum duration T_{deorb} . It corresponds to a minimum waiting phase once a debris has been reached before starting the next transfer.

2.2.2 Path Problem Formulation

In terms of graphs theory, the debris are represented by nodes and the transfers are represented by directed edges. To write down the mathematical formulation of the path problem, we introduce the following variables and constraints²¹. By convention, variables with one index are related to nodes whereas variables with two indexes are related to edges.

- s_{ij} , $i, j=1$ to N with $i \neq j$, is a binary selection variable for the edge $(i-j)$
- x_k , $k=1$ to N , is the number of edges arriving to the node k : $x_k = \sum_i s_{ik} \leq 1$
- y_k , $k=1$ to N , is the number of edges departing from the node k : $y_k = \sum_j s_{kj} \leq 1$
- z_k , $k=1$ to N , is the product of x_k and y_k : $z_k = x_k \cdot y_k$
 $z_k = 1$ if and only if there is one edge arriving and one edge departing from the debris k .
- s_k , $k=1$ to N , is a binary selection variable for the node k : $s_k = x_k + y_k - z_k = \text{Min} \left(1, \sum_i s_{ik} + \sum_j s_{kj} \right)$

Remark 1

Although the variables x_k , y_k , z_k and s_k are redundant with the variables s_{ij} , they are kept in the problem formulation together with the related constraints in view of the further problem linearization (Chapter 4).

- t_k^a and t_k^d , $k=1$ to N are the dates of arrival to and departure from the debris k

The mission starts at $t_0=0$ and ends at $t < T_{\text{max}}$ (1 year) : $0 < t_k^a$, $t_k^d < T_{\text{max}}$. We introduce also a lower bound T_{min} (for example 1 day) on the transfer durations. This bound discards fuel-expensive transfers that would anyway be rejected when optimizing the path.

The path feasibility is ensured from the following set of constraints :

- From : $\sum_k x_k = \sum_{i,k} s_{ik} = n - 1$ (number of "arrival" debris)
- and $\sum_k z_k = n - 2$ (number of "mid path" debris)

we deduce that there is only one debris which is an arrival without departure (end of the mission).

- From : $\sum_k y_k = \sum_{k,j} s_{kj} = n - 1$ (number of "departure" debris)
- and $\sum_k z_k = n - 2$ (number of "mid path" debris)

we deduce that there is only one debris which is a departure without arrival (beginning of the mission).

- From : $t_k^d - t_k^a \geq T_{\text{deorb}}$ (debris operations)
and $t_i^d - t_j^a + T_{\text{max}} \cdot s_{ij} \leq T_{\text{max}} - T_{\text{min}}$ (increasing dates along the selected path)

we ensure that the required operation durations are satisfied and that the path has no loop (Figure 3 left).

Indeed for a selected edge ($s_{ij}=1$) the constraint becomes $t_i^d + T_{\text{min}} \leq t_j^a$ with a minimum transfer duration T_{min} .

Adding $\sum_{i,j} s_{ij} = n - 1$ (number of edges)

and $\sum_k z_k = n - 2$ (number of “mid path” debris)

we ensure that the path is made of $n-1$ edges and that it is connex (Figure 3 right).

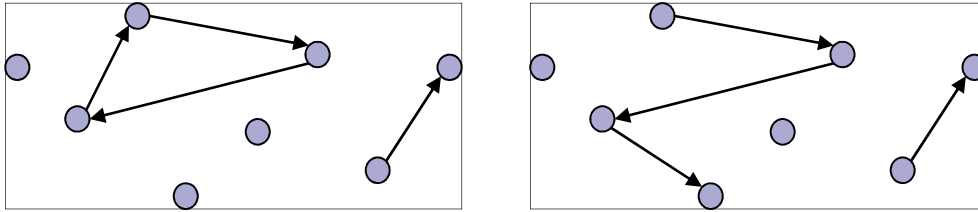


Figure 3 : Examples of non feasible paths

The path total cost C and duration T are $\begin{cases} C = \sum_{i,j} s_{ij} C_{ij} + \sum_k s_k C_k \\ T = \sum_{i,j} s_{ij} T_{ij} + \sum_k s_k T_k \end{cases}$ with the respective contributions :

- C_{ij} and $T_{ij} = t_j^a - t_i^d$ for the transfer (i-j)
- C_k and $T_k = t_k^d - t_k^a$ for the debris k operations

The cost C depends on the debris selection order because of the constraints applied to the variables s_{ij} and s_k .

2.3 Space Debris Collecting problem

The SDC problem is a nonlinear time dependent variant of the classical TSP. The main differences with the TSP come from the embedded transfer problems :

- The town locations are replaced by the debris locations on their orbits (therefore varying with the time)
- There is an infinite number of possible trajectories to go from any debris i to any other debris j depending on the dates t_i^d, t_j^a and on the command law $[U_{ij}(t), t_i^d \leq t \leq t_j^a]$. Each possible trajectory requires a propellant consumption (or cost) $C_{ij}(t_i^d, t_j^a, X(t_i^d), X(t_j^a), [U_{ij}(t), t_i^d \leq t \leq t_j^a])$ and a duration $T_{ij} = t_j^a - t_i^d$.

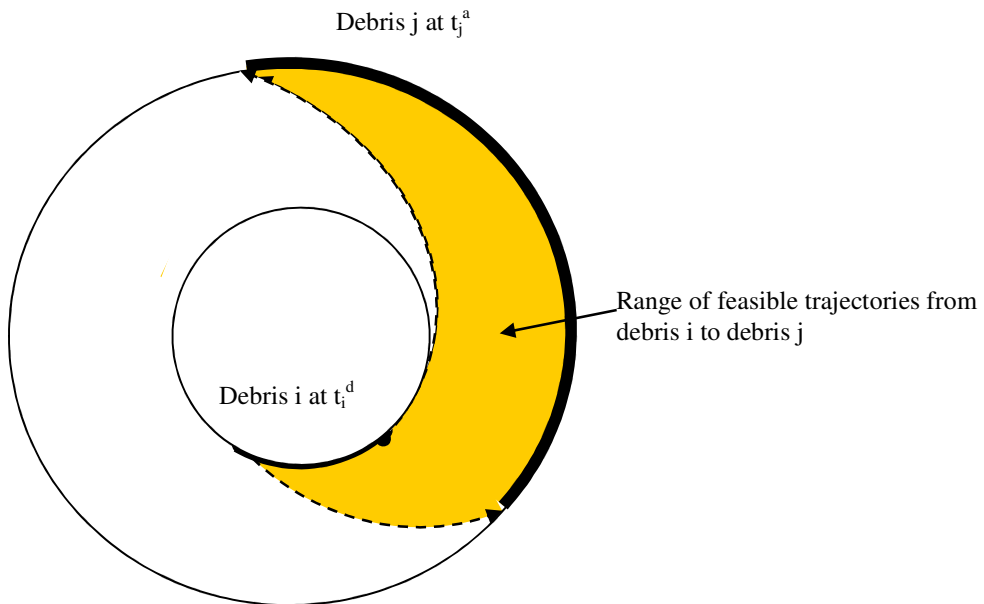


Figure 4 : Trajectories between debris

The SDC problem formulation mixes binary variables s_{ij} , x_k , y_k , z_k , s_k , real variables $t_k^d, t_k^a, X(t_k^d), X(t_k^a)$ and functions $[U_{ij}(t), t_i^d \leq t \leq t_j^a]$:

$$\begin{aligned} \text{Min}_{s_{ij}, x_k, y_k, z_k, s_k, t_k^d, t_k^a, X(t_k^d), X(t_k^a), U_{ij}(t)} C = \sum_{i,j} s_{ij} C_{ij} + \sum_k s_k C_k \quad \text{s.t.} \quad & \begin{cases} S = \sum_{i,j} s_{ij} = n - 1 \\ T = \sum_{i,j} s_{ij} T_{ij} + \sum_k s_k T_k \leq T_{\max} \\ x_k = \sum_i s_{ik} \\ y_k = \sum_j s_{kj} \\ z_k = x_k \cdot y_k \\ s_k = x_k + y_k - z_k & 1 \leq i, j, k \leq N \\ \sum_k z_k = n - 2 & i \neq j \\ t_k^d - t_k^a \geq T_{\text{deorb}} \\ t_i^d - t_j^a + T_{\max} \cdot s_{ij} \leq T_{\max} - T_{\min} \\ Y(t_i^d) = Y_i(t_i^d) \\ Y(t_j^a) = Y_j(t_j^a) \\ \dot{X}(t) = f[X(t), U_{ij}(t), t], \quad \text{for } t_i^d \leq t \leq t_j^a \end{cases} \quad (4) \end{aligned}$$

- S is the number of transfers : $S = \sum_{i,j} s_{ij}$
- C is the mission cost : $C = \sum_{i,j} s_{ij} C_{ij} + \sum_k s_k C_k$
- T is the mission duration : $T = \sum_{i,j} s_{ij} T_{ij} + \sum_k s_k T_k$
- Y is the location state vector : $X(t) = \begin{pmatrix} Y(t) \\ m(t) \end{pmatrix}$
- C_{ij} is the cost of the transfer (i-j) : $C_{ij}(t_i^d, t_j^a, X(t_i^d), X(t_j^a), [U_{ij}(t), t_i^d \leq t \leq t_j^a]) = \int_{t_i^d}^{t_j^a} \frac{u_{ij}(t)}{v_e} dt$
- C_k is the cost of the debris k operations : $C_k(t_k^a, t_k^d, Y_k(t_k^a), Y_k(t_k^d))$
- T_{ij} is the duration of the transfer (i-j) : $T_{ij} = t_j^a - t_i^d$
- T_k is the duration of the debris k operations : $T_k = t_k^d - t_k^a$
- N and n are respectively the total number of debris and the number of debris to deorbit

The mission is assessed from the first selected debris, assuming that a launcher has realized the prior rendezvous manoeuvres. If not, the same formulation applies by adding a fictitious debris numbered 0 corresponding to the SDC vehicle injection orbit. This fictitious debris is fixed as the start of the path by fixing the related variables ($x_0=0$, $y_0=1$, $z_0=0$ and $s_0=1$), and the transfer from this debris 0 to the first selected debris is then taken into account.

3 Transfer strategy

The optimal transfer strategy depends on the initial and final orbits, the mission constraints and the vehicle capabilities. In the general case, finding the optimal control law may be a challenging task. For the SDC mission it is nevertheless possible to define a specific transfer strategy that takes advantages of the mission particularities, regarding the debris orbits and the mission duration.

3.1 Debris orbits

For each debris, the orbital parameters (Figure 5) are given at the mission starting date t_0 . The semi-major axis a and the eccentricity e define the orbit shape, the inclination I and the right ascension of the ascending node (RAAN) Ω define the orbit plane, the argument of the perigee ω and the true anomaly v define the location on the orbit.

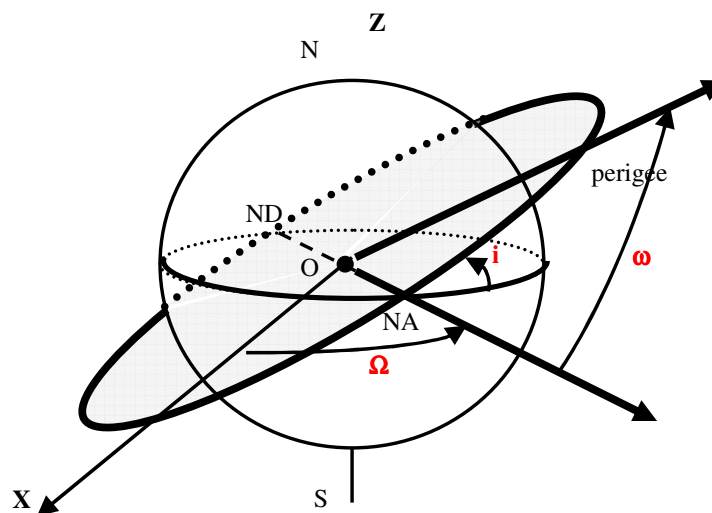


Figure 5 : Orbital parameters

The Space Debris Collecting mission aims at Low Earth Orbit (LEO) or Sun-Synchronous Orbit (SSO) old satellites. During their operational life the orbital parameters have been controlled to keep precise values matching the mission purposes, e.g. observation or telecommunication. For example typical SSO are circular at altitudes ranging from 600 to 1000 km, and inclination ranging from 96 to 100 deg. Similarly the launcher upper stages used to bring the satellites on their operational orbit have been left in the vicinity of these orbits.

After their end of life, the satellites have been left uncontrolled and subjected to perturbations (Earth gravitational perturbations, Sun and Moon attraction, atmospheric drag, solar pressure radiation, geomagnetic field, etc.). For nearly circular LEO or SSO, the average effect of the perturbations on the orbit semi-major axis, the eccentricity and the inclination is very small : this is confirmed both by theory, numerical simulations and by observation²². These debris are thus still moving on nearly circular orbits, at very close altitudes and inclinations.

In the altitude range [600 km – 1000 km] the main orbital perturbation is due to the Earth flattening, which adds the first zonal term J_2 to the Earth gravitational potential. The Earth equatorial bulge creates a torque on the debris orbit. The debris rotating on its orbit behaves as a gyroscope : the angular momentum rotates around the Earth polar axis causing a precession of the orbital plane and a secular drift of the node along the Equator (Figure 6).

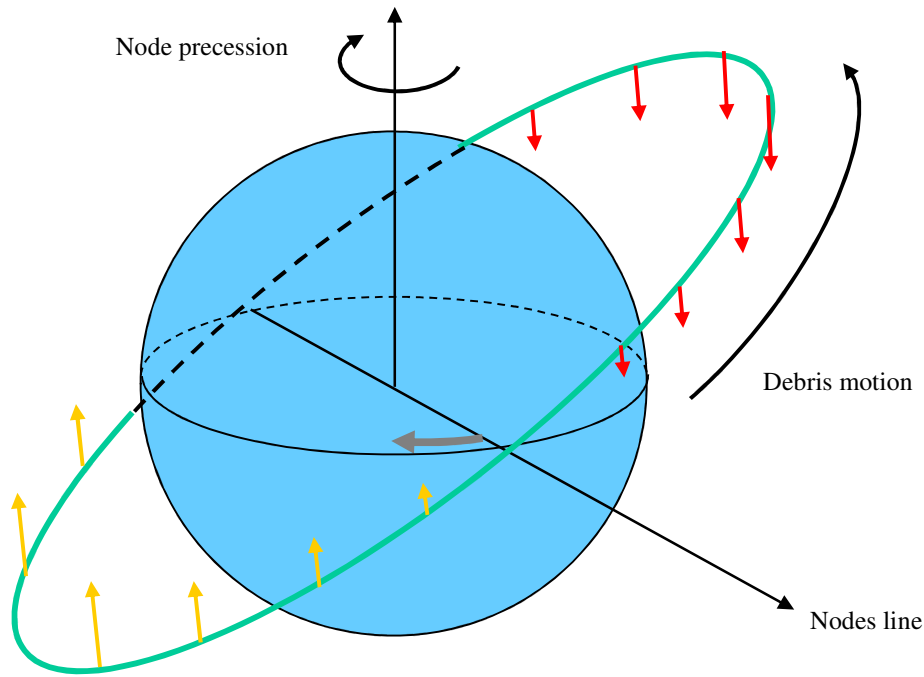


Figure 6 : Orbit precession due to the Earth flattening

The secular precession rate of the RAAN is expressed as²³ :

$$\dot{\Omega} = \frac{d\Omega}{dt} = -\frac{3}{2} J_2 n \left(\frac{R_T}{a} \right)^2 \frac{1}{(1-e^2)^2} \cos I = -C_{J_2} \frac{\cos I}{a^2 (1-e^2)^2} \quad (5)$$

The Earth constants given by the WGS84 model²³ are the gravitational constant ($\mu=3.986 \cdot 10^{14} \text{ m}^3/\text{s}^2$), the equatorial radius ($R_T=6378137\text{m}$), the first zonal coefficient ($J_2=1.086 \cdot 10^{-3}$, $C_{J_2} = \frac{3}{2} J_2 \sqrt{\mu} R_T^2$). $n = \sqrt{\frac{\mu}{a^3}}$ is the mean motion.

The other parameters a , e and I are not subjected to secular effects from the J_2 perturbation and they can be considered as constant throughout the SDC mission. The RAAN precession rate (Equation 5) is therefore constant : $\dot{\Omega}(t) = \dot{\Omega}(t_0) = \dot{\Omega}_0$ and the RAAN is a linear function of the time : $\Omega(t) = \Omega(t_0) + \dot{\Omega}_0 \times (t - t_0)$

For a Sun-Synchronous Orbit the precession rate is adjusted to match the Earth revolution rate around the Sun (360 deg in 1 year, i.e. 0.986 deg/day). The orbital plane makes a complete revolution around the polar axis in one year keeping a constant angle with the Sun direction, as illustrated on

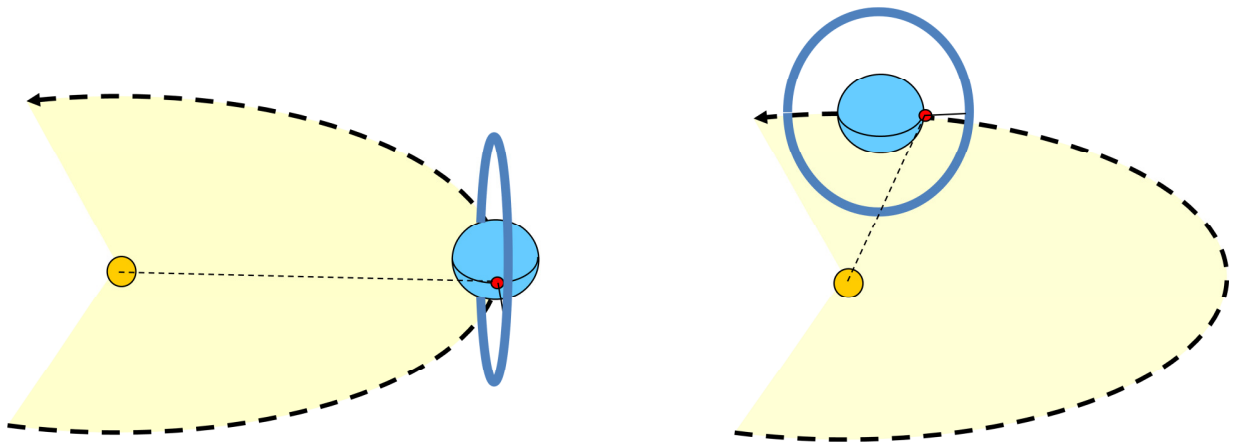


Figure 7 : Sun-Synchronous Orbit

Table 2 gives the range of values of the RAAN precession rates for near SSO debris.

RAAN precession rate		Semi major axis	
		7000 km	7200 km
Inclination	98 deg	1.002 deg/day	0.908 deg/day
	99 deg	1.126 deg/day	1.020 deg/day

Table 2 : RAAN precession rates for near SSO debris

For nearly circular orbits, the location on the orbit can be defined by the longitude from the ascending node $L = \omega + v$ replacing ω and v . In summary, the orbital parameters are split into :

- 3 constants a, e, I with $e \approx 0$,
- 1 slow variable $\Omega(t) = \Omega(t_0) + \dot{\Omega}_0 \times (t - t_0)$,
- 1 fast variable $L(t) = L(t_0) + n \times (t - t_0)$.

The next sections discuss the performances of a direct vs a drift strategy, the influence of the duration constraint and of the thrust level. A specific drift strategy is then selected in §3.6, assuming that the mission is performed by a single vehicle and considering the consumption as representative of the mission cost.

3.2 Direct strategy

The debris orbits are quasi-circular at close altitudes ($700 \text{ km} < H < 900 \text{ km}$) and inclinations ($98 \text{ deg} < I < 99 \text{ deg}$) with $a = R_T + H$ for a circular orbit.

To gain some insight into the features of an optimal transfer strategy, we first assess the impulsive costs for direct orbital parameter changes on a typical circular SSO ($H=800 \text{ km}$, $I=98.6 \text{ deg}$) :

- $\Delta H = \pm 100 \text{ km} \Rightarrow \Delta V = 50 \text{ m/s}$ (altitude correction by a Hohmann transfer)
- $\Delta I = \pm 1 \text{ deg} \Rightarrow \Delta V = 130 \text{ m/s}$ (inclination correction by a single impulse at the node)
- $\Delta \Omega = \pm 1 \text{ deg} \Rightarrow \Delta V = 130 \text{ m/s}$ (RAAN correction by a single impulse at the plane intersection²³)
- $\Delta \Omega = \pm 10 \text{ deg} \Rightarrow \Delta V = 1300 \text{ m/s}$

The anomaly correction for the rendezvous with the targeted debris is assessed considering a Lambert manoeuvre²³ :

- $\Delta L = \pm 10 \text{ deg} \Rightarrow \Delta V = 150 \text{ m/s}$ (anomaly correction)
- $\Delta L = -90 \text{ deg} \Rightarrow \Delta V = 1010 \text{ m/s}$
- $\Delta L = +90 \text{ deg} \Rightarrow \Delta V = 1640 \text{ m/s}$

All the debris altitudes and inclinations being very close, the changes of H and I to go from any debris to any other can be performed directly at moderate costs, typically less than 200 m/s , whatever the date. On the other hand, the RAAN difference changes with the time since each debris has its own precession rate. The required RAAN corrections may therefore take large values depending on the SDC mission starting date. As illustrated by the above numerical example, the impulsive cost for a direct change becomes prohibitive as soon as the required correction exceeds a few degrees. The same remark applies to the anomaly correction if the manoeuvre date cannot be chosen freely. A direct change strategy would therefore lead to prohibitive costs and an oversized SDC vehicle.

3.3 Drift strategy

An alternative strategy consists in waiting until the J_2 precession has nullified the RAAN difference. For a chaser and a target of respective orbital parameters $(a_1, e_1, I_1, \Omega_1)$ and $(a_2, e_2, I_2, \Omega_2)$ at an initial date t_0 , the RAAN evolutions are :

- $\Omega_1(t) = \Omega_1(t_0) + \dot{\Omega}_1 \times (t - t_0)$ with $\dot{\Omega}_1, \dot{\Omega}_2$ given by Equation 5.

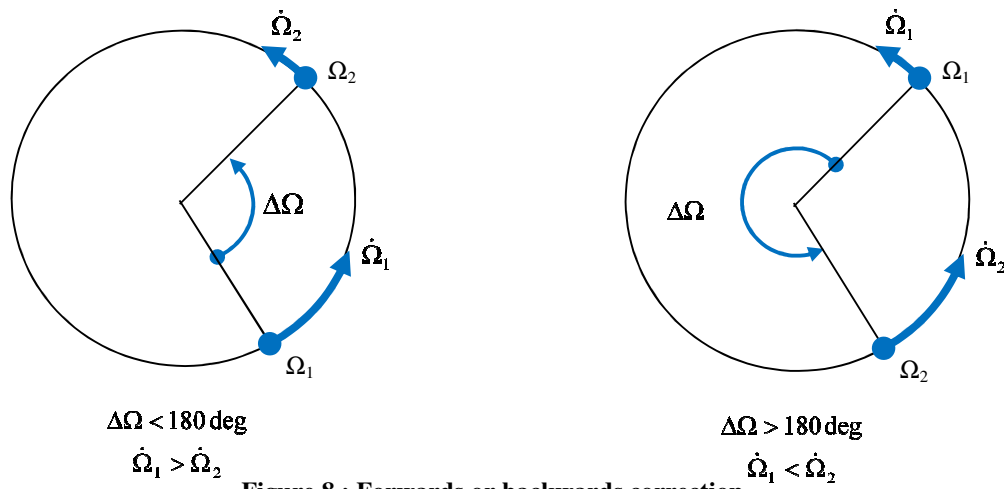
- $\Omega_2(t) = \Omega_2(t_0) + \dot{\Omega}_2 \times (t - t_0)$

The RAAN correction is completed when $\Omega_1(t) = \Omega_2(t)$ and the required duration is: $\Delta T = -\frac{\Omega_2 - \Omega_1}{\dot{\Omega}_2 - \dot{\Omega}_1} = -\frac{\Delta\Omega}{\Delta\dot{\Omega}}$.

The value of $\Delta\Omega$ is to take modulo 360 deg in order to match the sign of $\Delta\dot{\Omega}$ and yield a positive duration.

This purely waiting strategy is not time efficient for 2 reasons :

- The debris orbits are very close in terms of altitudes and inclinations, and the precession rates difference $\Delta\dot{\Omega}$ is therefore rather small (typically $\Delta\dot{\Omega} \approx \pm 0.1$ deg/day from Table 2).
- The sign of $\Delta\dot{\Omega}$ may not match the optimal correction sense. This is illustrated on Figure 8 in the case of positive precession rate (retrograde orbits like SSO).



If the chaser is initially backwards ($\Delta\Omega < 180 \text{ deg}$), it should try to overtake the target ($\dot{\Omega}_1 > \dot{\Omega}_2 \Rightarrow \Delta\dot{\Omega} < 0$) and if it is initially forwards ($\Delta\Omega > 180 \text{ deg}$), it should try to wait for it ($\dot{\Omega}_1 < \dot{\Omega}_2 \Rightarrow \Delta\dot{\Omega} > 0$).

In the worst cases (small $\Delta\dot{\Omega}$, wrong sense) waiting for a natural correction would take hundreds of days. Similarly a waiting strategy for the anomaly correction is not time efficient, since the respective mean motions of the debris can be very close.

In order to control the waiting duration, a more effective strategy consists in transferring the vehicle on an intermediate drift orbit where the RAAN correction will be speeded up. The drift orbit parameters are chosen to control both the sense of the correction (depending on whether the drift altitude is lower or higher than the target) and the correction speed (depending on the difference of the precession rates). A compromise has to be made between the cost of the additional transfer manoeuvres and the durations allocated to achieve the RAAN correction.

By the same way, a significant difference between the mean motions on the drift orbit and on the targeted debris orbit ensures that the anomaly phasing can be quickly (i.e. within a few revolutions) achieved whatever the initial anomaly difference, making the rendezvous cost negligible, once the RAAN correction is achieved.

3.4 Duration constraint

The duration constraint is necessary to have a well posed problem else the optimal solution is in infinite time^{6,7}.

Qualitatively we can distinguish :

- A weak duration constraint if it leaves enough time to make the drift strategy possible. The RAAN correction using the J_2 precession may then span on several months, depending on the initial state. In that case the rendezvous cost can be considered as negligible with respect to the orbit change manoeuvres.
- A strong duration constraint if there is not enough time to use efficiently the J_2 precession for RAAN correction. In that case, the direct change strategy must be considered, taking directly into account the rendezvous constraint at the expense of higher mission costs. For example reference²⁴ presents a Lambert based strategy in order to deorbit 3 LEO debris within a few days.

3.5 Thrust level

The term ‘thrust level’ is used, although it is in fact the acceleration level that reflects the orbit change capability of the vehicle. The way the transfer is optimized depends highly on the thrust level and on the transfer strategy :

- In the case of a high thrust engine, the manoeuvre durations can be considered as negligible with respect to the coast phase durations, making the impulsive modeling adequate. It is then possible to analyze the transfer in a “patched-conics” manner, and to define a specific strategy based on Hohmann transfers (for the orbit changes) and Lambert transfers (for the rendezvous).
- In the case of a low thrust engine, the manoeuvre durations are no longer negligible. A global optimization is required taking into account the coupling between the successive phases of the transfer. In the case of the drift strategy, approximate solutions can be derived from the impulsive solution by taking into account the cost and duration penalty of low thrust manoeuvres. This approach is realistic provided that the manoeuvre durations remain compliant with the drift strategy. If this assumption is not satisfied, it is necessary to directly tackle the global transfer optimization problem through optimal control methods.

Remark 2

The available acceleration level of the vehicle grows throughout the mission, since the engine thrust level remains constant while the vehicle gross mass decreases. A transfer between two debris can therefore be achieved at lower propellant consumption if it occurs at the mission end rather than at the beginning. The path optimization should then take in account this acceleration level increase in the case of non-impulsive manoeuvres.

Remark 3

Similarly the propellant consumption for the deorbitation manoeuvre, if it is performed by the vehicle, depends on the debris mass. The debris selection should then take into account their masses.

For the upcoming definition of the transfer strategy, it is assumed that the vehicle is equipped with a high-thrust engine and that the targeted debris are of similar size and mass in order to be compliant with the vehicle catching system. With these assumptions the path optimization no longer depends on the acceleration level variation nor on the debris masses.

3.6 Selected transfer strategy

The SDC mission starts from the first selected debris and requires four successive transfers that must be realized within one year. This leaves time to perform the ascending node changes using the J_2 precession (weak duration constraint). The drift transfer strategy is therefore adequate for the SDC mission in order to minimize the overall consumption.

Remark 4 : The real cost of the mission does in fact not only depend on the propellant consumption. The mission duration is also a major cost driver, for the vehicle design (thermal systems, on board power, ...) and for the operations (ground tracking, manoeuvre planning, collisions avoidance, ...). In terms of global cost, the propellant savings with the drift strategy may be offset by the cost induced by the mission duration. The trade-off between the direct strategy and the drift strategy must therefore be investigated before fixing the vehicle definition and the mission strategy.

Assuming that the SDC vehicle is equipped with a high thrust engine, the impulsive approximation can be considered as valid. Extension to low thrust engines is the subject of future work.

For a transfer from a debris 1 to a debris 2, we define a specific strategy composed of 3 phases and 4 manoeuvres :

- A first Hohmann transfer starting at the date t_1 to go from the debris 1 orbit to the intermediate drift orbit with 2 apsidal manoeuvres ΔV_{P1} (perigee) and ΔV_{A1} (apogee)
- A waiting phase of duration ΔT_{12} on the drift orbit until the RAAN correction is completed
- A second Hohmann transfer starting at the date t_2 to go from the intermediate drift orbit to the debris 2 orbit with 2 apsidal manoeuvres ΔV_{A2} (apogee) and ΔV_{P2} (perigee)

The parameters of the initial, drift and final orbits are denoted with the respective subscripts 1, d and 2.

The transfer strategy is depicted on Figure 9 in the case of a drift orbit higher than the debris orbits. In the problem solution there will be no a priori assumption regarding the relative altitudes of the 3 orbits. The drift orbit can be either below, or between or over the initial and final debris orbits.

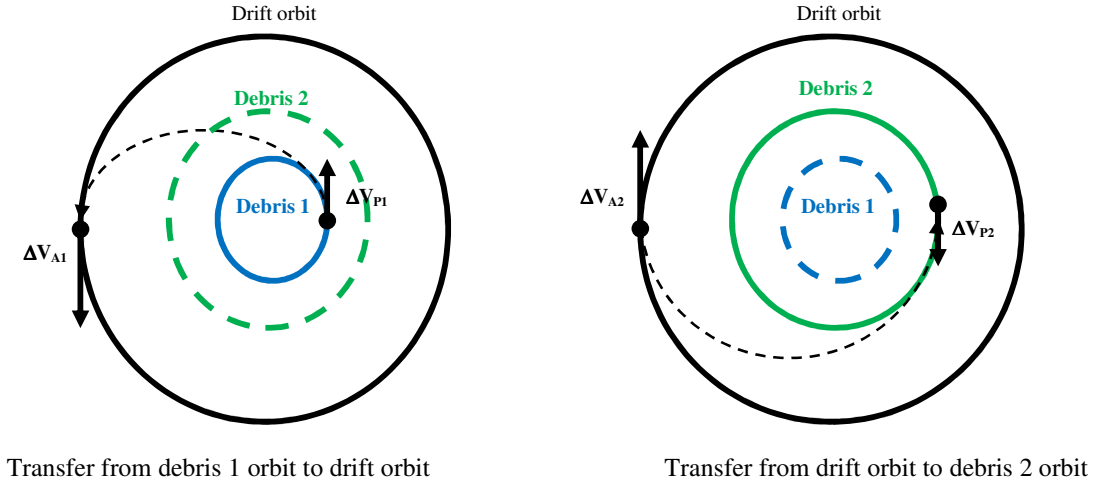


Figure 9 : Transfer manoeuvres and successive orbits

The Hohmann transfer durations are considered as negligible (typically a half revolution, i.e. about 1h) with respect to the drift duration (several weeks). The RAAN evolution of the debris and of the SDC vehicle along their respective orbits since the mission starting date t_0 are :

- Debris 1 : $\Omega_1(t_1) = \Omega_1(t_0) + \dot{\Omega}_1 \times (t_1 - t_0)$
- Debris 2 : $\Omega_2(t_2) = \Omega_2(t_0) + \dot{\Omega}_2 \times (t_2 - t_0)$
- Vehicle : $\Omega(t_1) = \Omega_1(t_1)$ (start at t_1 from the debris 1 orbit)
 $\Omega(t_2) = \Omega_2(t_2)$ (arrival at t_2 to the debris 2 orbit)
 $\Omega(t_2) = \Omega(t_1) + \dot{\Omega}_d \times (t_2 - t_1)$ (coast since t_1 until t_2 on the drift orbit)

The duration required to complete the RAAN correction is then given by :

$$T_{12} = t_2 - t_1 = \frac{\Omega_2(t_0) - \Omega_1(t_0) + (\dot{\Omega}_2 - \dot{\Omega}_1) \times (t_1 - t_0)}{\dot{\Omega}_d - \dot{\Omega}_2} \quad (6)$$

The anomaly rendezvous constraint is not considered since it can be satisfied with a negligible additional cost and duration by selecting the appropriate date of second transfer once the RAAN correction is achieved.

The overall cost of the transfer is the sum of the 4 manoeuvres : $\Delta V = \Delta V_{P1} + \Delta V_{A1} + \Delta V_{A2} + \Delta V_{P2}$

Each impulsive manoeuvre is assessed as the difference of the velocity vectors before and after the orbit change.

The velocity modulus depends on the semi-major axis a and the current radius vector r^{23} : $V = \sqrt{\mu \left(\frac{2}{r} - \frac{1}{a} \right)}$

In the case an inclination change is performed simultaneously with a shape change, the impulsive manoeuvre is assessed as the norm of the vector difference (Figure 10).

$$\Delta V = \|\vec{V}_a - \vec{V}_b\| = \sqrt{V_a^2 + V_b^2 - 2V_a V_b \cos(i_b - i_a)} \quad (7)$$

where \vec{V}_b (resp \vec{V}_a) and i_b (resp i_a) are the velocity and inclination before (resp after) the manoeuvre.

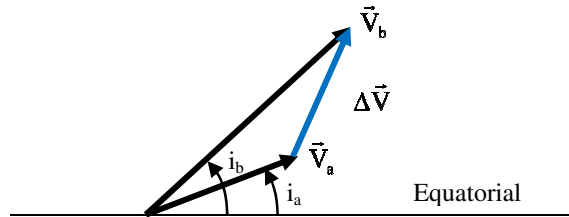


Figure 10 : Simultaneous inclination and shape change

3.7 Problem formulation with the selected transfer strategy

With the selected strategy, the control problem (Equation 3) is simplified as follows :

- The command law $[U_{ij}(t), t_i^d \leq t \leq t_j^a]$ for each transfer (i-j) is replaced by the drift orbit parameters a_{ij}, e_{ij}, I_{ij} . There are no longer function unknowns and the problem becomes a finite dimension NLP problem.
- The dynamical constraint $\dot{X}(t) = f[X(t), U_{ij}(t), t], t_i^d \leq t \leq t_j^a$ disappears since the trajectories are modeled by keplerian arcs (Hohmann transfers) with secular J_2 RAAN precession (drift orbits).
- The initial and final state constraints on $Y(t_i^d)$ and $Y(t_j^a)$ are discarded since they are directly taken into account in the transfer modeling (the anomaly rendezvous constraint is neglected).
- The transfer costs C_{ij} are measured by the required impulsive velocities ΔV_{ij} , depending only on the parameters of the initial, drift and final orbit. The mass is also discarded from the problem formulation.
- The transfer durations T_{ij} depend on the parameters of the initial, drift and final orbit, and also on the date of the transfer beginning (Equation 6).

With these simplifications the SDC problem (Equation 4) becomes a mixed integer-real finite dimension problem, formulated as :

$$\begin{array}{l}
\text{Min}_{s_{ij}, x_k, y_k, z_k, s_k, t_k^a, t_k^d, a_{ij}, e_{ij}, I_{ij}} C \quad \text{s.t.} \quad \left\{ \begin{array}{l}
S = n - 1 \\
T \leq T_{\max} \\
x_k = \sum_i s_{ik} \\
y_k = \sum_j s_{kj} \\
z_k = x_k \cdot y_k \\
s_k = x_k + y_k - z_k \\
\sum_k z_k = n - 2 \\
t_k^d - t_k^a \geq T_{\text{deorb}} \\
t_i^d - t_j^a + T_{\max} \cdot s_{ij} \leq T_{\max} - T_{\min}
\end{array} \right. \quad (8)
\end{array}$$

- S is the number of transfers : $S = \sum_{i,j} s_{ij}$
- C is the mission cost : $C = \sum_{i,j} s_{ij} C_{ij} + \sum_k s_k C_k$
- T is the mission duration : $T = \sum_{i,j} s_{ij} T_{ij} + \sum_k s_k T_k$

The cost C_{ij} and duration T_{ij} of the transfer (i-j) depend on the parameters of the initial orbit (debris i), the final orbit (debris j) and the drift orbit. The duration depends also on the date of the transfer beginning t_i^d :

$$\begin{cases}
C_{ij} = \Delta V_{ij} \\
T_{ij} = t_j^a - t_i^d = \frac{\Omega_j(t_0) - \Omega_i(t_0) + (\dot{\Omega}_j - \dot{\Omega}_i) \times (t_i^d - t_0)}{\dot{\Omega}_{ij} - \dot{\Omega}_j}
\end{cases} \quad (9)$$

3.8 Further simplifications

The problem dimension can be further reduced regarding the choice of the drift orbit eccentricity and inclination.

3.8.1 Drift orbit eccentricity

We examine two extreme scenarios regarding the choice of the drift orbit eccentricity :

- The first scenario uses an elliptical drift orbit having the same perigee (or apogee) as the initial debris orbit. This scenario requires only three manoeuvres (one for the first transfer, two for the second transfer).
- The second scenario uses a circular drift orbit. This scenario requires four manoeuvres (two for the first transfer, two for the second transfer) as depicted on Figure 9.

The comparison is led on a representative numerical example with the initial and final debris on the same SSO (800 km , 98.6 deg). We assume also that the drift orbit altitude is comprised between 600 km and 1000 km.

The plots of Figure 11 show for the two scenarios the manoeuvre costs depending on the drift orbit altitude (or perigee or apogee in the elliptical scenario) and the precession duration required for a one degree RAAN correction.

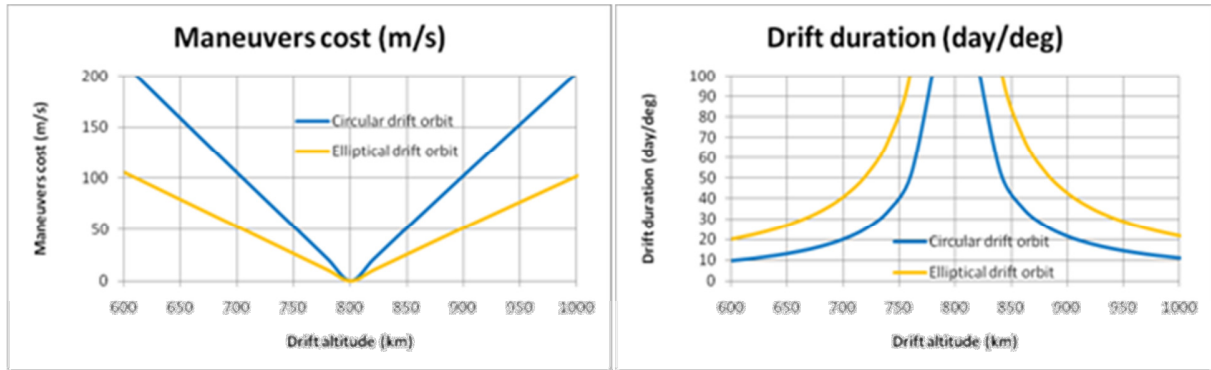


Figure 11 : Elliptical vs. circular drift orbit

The fastest RAAN correction rate that can be hoped with the elliptical scenario is about 0.05 deg/day, with a related manoeuvre cost of 100 m/s. This means that a single RAAN correction of 10 deg could not be achieved in less than 200 days. This strategy is therefore not compliant with the SDC mission global duration of one year.

On the other hand the circular drift orbit scenario allows a doubling of the RAAN correction rate, but at the expenses of higher impulsive costs. This numerical example shows that considering elliptical drift orbits can help reduce the mission cost, but makes the solution more sensitive to the duration constraint.

We will therefore consider only circular drift orbits and fix the drift eccentricities e_{ij} at zero. This assumption is the most conservative with respect to the duration constraint, since it maximizes the RAAN correction rates of each transfer. It can later be relaxed when post-optimizing the trajectory (once the path is selected) in order to reduce the mission cost.

3.8.2 Drift orbit inclination

Regarding the drift orbit inclination a similar analysis can be made as for the eccentricity. The respective inclinations on the initial and final debris orbits are I_1 and I_2 . In order to minimize the overall transfer cost, the inclination change has to be performed simultaneously with one of the four manoeuvres of the transfer. From the impulsive formula (Equation 7), the minimum cost is achieved for the minimum velocity modulus product $V_a V_b$. Assuming identical altitudes $H_1=H_2$ for the initial and the final debris with respective inclinations I_1 and I_2 , we can see on Figure 12 that there are two cost-equivalent opportunities at the apogee manoeuvres for the inclination change :

- If the drift orbit is above the debris orbits, these opportunities correspond to the 2nd and 3rd manoeuvres

- If the drift orbit is under the debris orbits, these opportunities correspond to the 1st and 4th manoeuvres

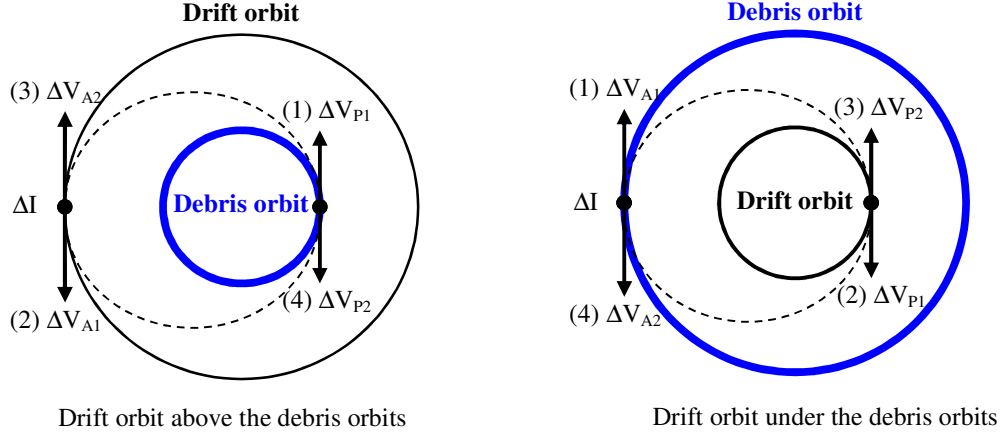


Figure 12 : Optimal inclination change opportunity

It is thus always possible to perform the inclination change either before or after the drift phase, without cost penalty. The criterion of choice is then to speed up the RAAN correction by maximizing the precession rate difference between the drift orbit and the final debris orbit.

For retrograde orbits ($I > 90$ deg), the precession rate (Equation 5) is positive : $\dot{\Omega} = -C_{J2} \cos I a^{-\frac{7}{2}} (1-e^2)^{-2} > 0$

and its variation with the semi-major axis and inclination are : $\frac{\partial \dot{\Omega}}{\partial a} < 0$, $\frac{\partial \dot{\Omega}}{\partial I} > 0$

The optimal drift orbit inclination with respect to the debris altitudes and inclinations is :

$$\begin{cases} I_{\text{drift}} = \text{Min}(I_1, I_2) & \text{if } H_{\text{drift}} > H_2 \\ I_{\text{drift}} = \text{Max}(I_1, I_2) & \text{if } H_{\text{drift}} < H_2 \end{cases}$$

We will therefore fix the drift inclinations I_{ij} at their optimal values with respect to the RAAN correction duration. In practice this assumption is quasi-optimal in terms of cost as long as the debris move at close altitudes. If not, the inclination change opportunities are not cost-equivalent and the drift inclinations I_{ij} should be kept free in the problem.

3.8.3 Debris operations

In addition to the previous assumptions regarding the drift orbit parameters, we assume that :

- the costs C_k for the debris operations (rendezvous, deorbitation, release) depend only on the debris orbits,
- the durations T_k for the debris operations are fixed ($T_k = T_{\text{deorb}}$) and identical for all the debris,
- the vehicle leaves each debris as soon as the deorbiting operations are completed : $t_k^d = t_k^a + T_{\text{deorb}}$,

- the minimum transfer duration T_{\min} is fixed to zero Indeed the constraints on the dates become

$$t_i^a - t_j^a + T_{\max} \cdot s_{ij} \leq T_{\max} - (T_{\min} + T_{\text{deorb}}) \text{ so that } T_{\min} \text{ is redundant with } T_{\text{deorb}} \text{ in the problem formulation.}$$

Delaying the departure is a loss of time : the next drift orbit must be reached as soon as possible to speed up the RAAN correction. A drift orbit identical to the initial orbit is anyway a possible solution that is not prevented by the above assumption. The arrival date on the debris k will consequently be noted without superscript : $t_k^a = t_k$ and the departure date is given by : $t_k^d = t_k + T_{\text{deorb}}$.

3.8.4 Problem simplified formulation

With these simplifying assumptions, the SDC problem formulation (Equation 8) becomes :

$$\text{Min}_{s_{ij}, x_k, y_k, z_k, s_k, t_k, a_{ij}} C = \sum_{i,j} s_{ij} C_{ij}(a_{ij}) + \sum_k s_k C_k \quad \text{s.t.} \quad \begin{cases} S = \sum_{i,j} s_{ij} = n - 1 \\ T = \sum_{i,j} s_{ij} T_{ij}(a_{ij}, t_i) + n T_{\text{deorb}} \leq T_{\max} \\ x_k = \sum_i s_{ik} \\ y_k = \sum_j s_{kj} \\ z_k = x_k \cdot y_k \\ s_k = x_k + y_k - z_k \\ \sum_k z_k = n - 2 \\ t_i - t_j + T_{\max} \cdot s_{ij} \leq T_{\max} - T_{\text{deorb}} \end{cases} \quad \begin{matrix} 1 \leq i, j, k \leq N \\ i \neq j \end{matrix} \quad (10)$$

with the cost C_{ij} and duration T_{ij} of the transfer (i-j) defined by Equation (9).

The numbers of variables and constraints are :

- $N^2 + 3N$ binary variables : $N(N-1)$ for the s_{ij} and $4N$ for the x_k, y_k, z_k, s_k
- N^2 real variables : $N(N-1)$ for the a_{ij} and N for the t_k
- $N^2 + 3N + 3$ constraints

This is a MINLP problem where the nonlinearities come from the cost functions $C_{ij}(a_{ij})$, the duration functions $T_{ij}(a_{ij}, t_i)$, their products with the edge selection variables s_{ij} and the relationships between the binary variables x_k, y_k and z_k . These last nonlinearities are easily removed by a standard transformation recalled in §4.3. The practical solution procedure of the MINLP problem is presented in the next chapter.

4 Practical solution

The above simplifications have removed the functional unknowns (control laws) and led to a finite dimension MINLP problem. But the problem is still nonlinear, and therefore not adapted to a Branch and Bound approach. Indeed the tree exploration requires solving repeatedly instances of the relaxed problem, with a guarantee of convergence. It is very difficult to ensure the required solution robustness for general NLP problems unless the convexity of the cost and constraint functions can be established. In the absence of convexity properties such problems present generally local minima making algorithms sensitive to the initialization. Also it is nearly impossible to find algorithmic settings robust whatever the problem.

In order to get the required convergence guarantee and time efficiency, the most effective approach consists in linearizing the problem. The advantage is the possibility to use reliable linear programming methods. The drawback is that the solution found is only valid in the vicinity of the starting point. Therefore the linearization and solution steps must be repeated within an iterative process until the solution stabilizes.

Another issue that must be discussed is how to guarantee of convergence toward the global optimum. Proving and ensuring this convergence is generally not possible unless the convexity of the cost and constraint functions can be established. In the case of convex functions outer convexification procedures can be added to the linearization in order to ensure that the iterates converge toward the global optimum¹⁴.

4.1 Problem linearization

The nonlinear terms in the above formulation (Equation 8) are the transfer costs $C_{ij}(a_{ij})$, the transfer durations $T_{ij}(a_{ij}, t_i)$ and the variable products $(s_{ij}C_{ij})$, $(s_{ij}T_{ij})$, $(x_k y_k)$.

4.2 Transfer cost and duration linearization

The cost and duration are linearized for each transfer around reference values of the drift orbit semi-major axis a_{ij} and of the transfer starting date t_i . In order to ease the explanation of the linearization process, we denote :

- a_d (resp a) and t_d (resp t) the reference (resp actual) values of the semi major axis and the starting date
- α and τ the differences between the actual and the reference values :
$$\begin{cases} a = a_d + \alpha \\ t = t_d + \tau \end{cases}$$
- $[a_{min}; a_{max}]$ and $[t_{min}; t_{max}]$ the intervals of validity of the linearization. This will be discussed when detailing the iterative process. The corresponding intervals on the variables α and τ are $[\alpha_{min}; \alpha_{max}]$ and $[\tau_{min}; \tau_{max}]$.

The nonlinear functions $C(a)$ and $T(a,t)$ are replaced by linear approximations $C_L(\alpha)$ and $T_L(\alpha, \tau)$ in these intervals.

$$\begin{cases} C_L(\alpha) = C(a_d) + \frac{\partial C}{\partial a} \alpha \\ T_L(\alpha, \tau) = T(a_{min}, t_d) + \frac{\partial T}{\partial a} (\alpha - \alpha_{min}) + \frac{\partial T}{\partial t} \tau \end{cases} \quad (11)$$

The cost C and the duration T partial derivatives are approximated in the linearization intervals by secant formulae :

$$\left\{ \begin{array}{l} \frac{\partial C}{\partial a} \approx \frac{C(a_{\max}) - C(a_{\min})}{a_{\max} - a_{\min}} \\ \frac{\partial T}{\partial a} \approx \frac{T(a_{\max}, t_d) - T(a_{\min}, t_d)}{a_{\max} - a_{\min}} \quad \text{assessed at } t = t_d \\ \frac{\partial T}{\partial t} \approx \frac{T(a_d, t_{\max}) - T(a_d, t_{\min})}{t_{\max} - t_{\min}} \quad \text{assessed at } a = a_d \end{array} \right. \quad (12)$$

This linearization choice for the duration function T (Equation 6) is driven by the function shape (Figure 13). This function is linear with respect to t_d , convex and highly nonlinear with respect to a_d with an asymptote at $a_d = a_2$ if $I_d = I_2$ (or near a_2 if $I_d \approx I_2$). Indeed the duration for the RAAN correction becomes infinite if the precession rates are the same.

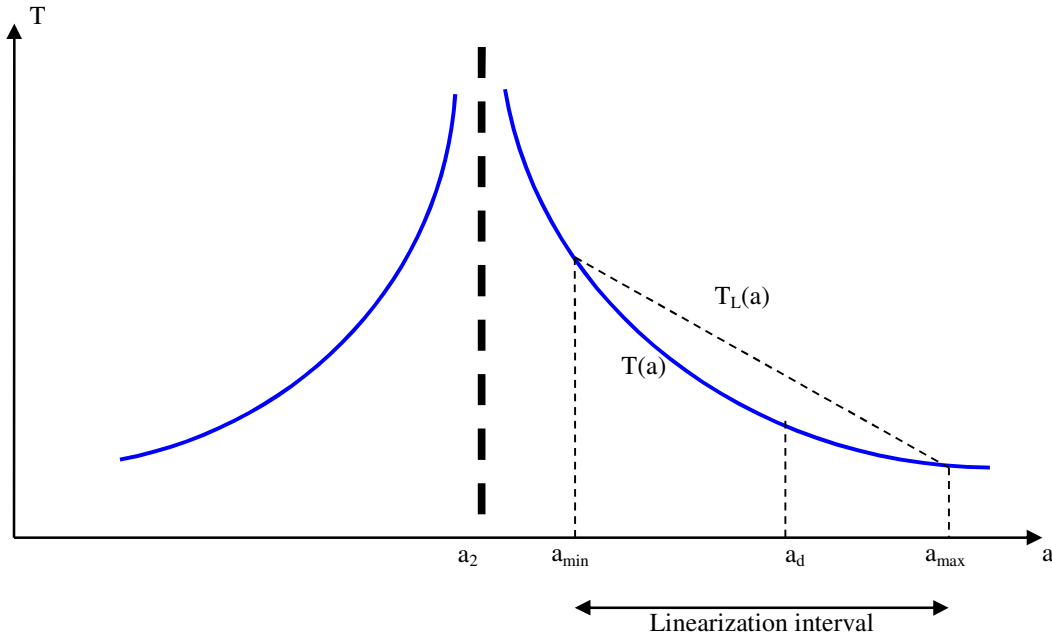


Figure 13 : Linearization of the duration function

The first linearization terms $T(a_{\min}, t_d) + \frac{\partial T}{\partial a}(a - a_{\min})$ yield an upper approximation of the true duration at the reference date t_d , while the last term $\frac{\partial T}{\partial t} \tau$ accounts for the linear dependence on the starting date. The choice of an upper approximation ensures that the linearized approximation overestimates the true duration of each transfer. The linearized solution is therefore admissible but sub-optimal with respect to the global duration constraint. The way to

converge on the optimal solution saturating the duration constraint consists in tightening the intervals $[\alpha_{\min}; \alpha_{\max}]$ and $[\tau_{\min}; \tau_{\max}]$ from one iteration to the other after the linearized solution stabilizes on the optimal path (see §4.6.2).

Alternative linearization approaches

Two alternative linearization approaches with respect to the semi-major axis variable have been envisioned.

The first one consists in considering directly the tangent at the linearization point instead of the secant on the linearization interval :

$$T_L(\alpha, \tau) = T(a_d, t_d) + \frac{\partial T}{\partial a}(a_d, t_d) \cdot \alpha + \frac{\partial T}{\partial t}(a_d, t_d) \cdot \tau$$

This linearized duration function underestimates the true duration (the tangent is below the curve). The mission real duration corresponding to the linearized solutions is therefore systematically higher than 1 year. The intermediate iterations yield never feasible solutions with respect to that constraint until the convergence is achieved. This first approach has proved practically difficult to control within the iteration process.

The second approach aims at meeting more accurately the true duration constraint at every iteration. It consists in approximating the duration function by 2 half segments as pictured on Figure 14 :

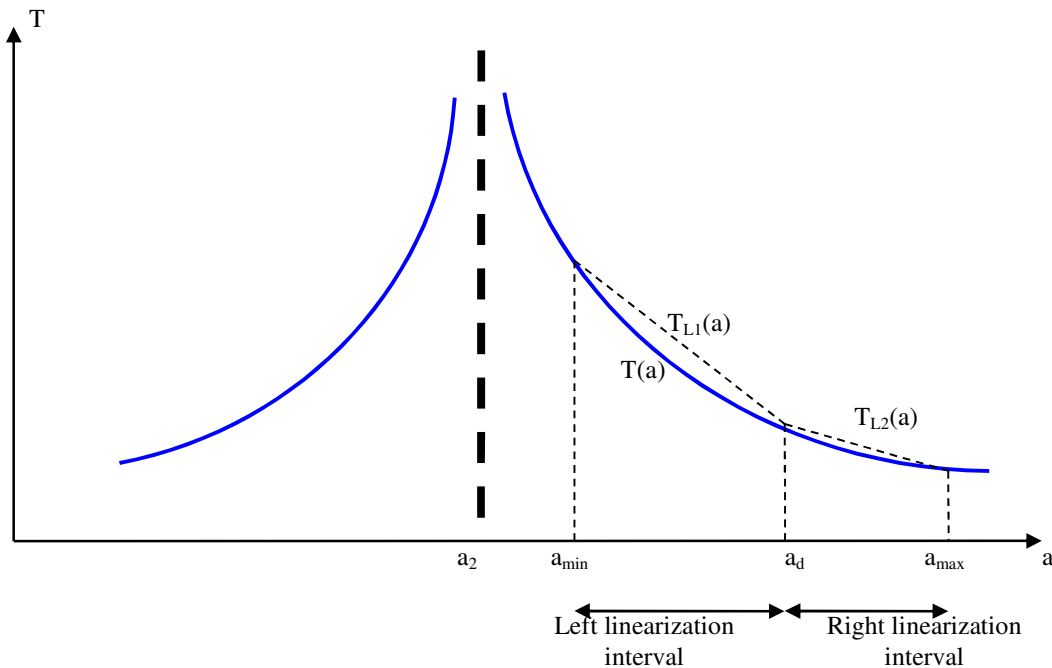


Figure 14 : Duration function approximation with two half segments

This approach makes necessary to take into account the sign of α_{ij} when linearizing the duration of the transfer (i-j) and it leads to an increase of the linearized problem dimension. The time of solving this larger dimension problem can nevertheless be balanced by a reduction of the number of iterations necessary to meet accurately the global duration constraint. The choice must be made case by case, depending on the iteration behavior and on the computation time observed.

4.3 Variable product linearization

A product of a binary variable $u \in \{0;1\}$ with a real variable $v \in [v_{\min}; v_{\max}]$ is linearized by introducing one additional real variable w representing the product $(u.v)$ and four inequality constraints²¹ :

$$\begin{cases} u.v_{\min} \leq w \leq u.v_{\max} \\ v-(1-u).v_{\max} \leq w \leq v-(1-u).v_{\min} \end{cases} \quad (13)$$

The variable w is then used in the problem formulation in order to replace the products $(u.v)$. If the variable v is a binary one the same transformation applies with $v_{\min} = 0$ and $v_{\max} = 1$. This transformation is applied to all the products $p_{ij}=s_{ij}.\alpha_{ij}$, $q_{ij}=s_{ij}.\tau_i$ and $z_k=x_k.y_k$, adding thus $2N^2-N$ real variables and $8N^2-4N$ constraints.

4.4 Linearized problem

The linearized problem formulation is obtained by successively :

- Choosing for each transfer (i-j) reference values of a_{ij} and t_i
- Replacing the variables a_{ij} and t_i by α_{ij} and τ_i
- Replacing the true cost C_{ij} and duration T_{ij} by their linear approximations C_{Lij} and T_{Lij} with their respective linearization coefficients (C_{0ij}, C_{1ij}) and $(T_{0ij}, T_{1ij}, T_{2ij})$:

$$\begin{cases} C_{Lij}(\alpha_{ij}) = C_{0ij} + C_{1ij}\alpha_{ij} \\ T_{Lij}(\alpha, \tau) = T_{0ij} + T_{1ij}\alpha_{ij} + T_{2ij}\tau_i \end{cases} \quad (14)$$

- Replacing the products by the associated variables and constraints (Equation 13). The product linearization variables are denoted :

$$\begin{cases} p_{ij} = s_{ij}\alpha_{ij} \\ q_{ij} = s_{ij}\tau_i \end{cases} \quad (15)$$

The SDC problem formulation after linearization is :

$$\begin{array}{c}
 \text{Min} \\
 s_{ij}, x_k, y_k, z_k, s_k, \alpha_{ij}, \tau_i, p_{ij}, q_{ij}
 \end{array}
 C_L \quad \text{s.t.} \quad \left\{ \begin{array}{l}
 \sum_{i,j} s_{ij} = n-1 \\
 \sum_k z_k = n-2 \\
 x_k = \sum_i s_{ik} \\
 y_k = \sum_j s_{kj} \\
 s_k = x_k + y_k - z_k \\
 \tau_i - \tau_j + T_{\max} \cdot s_{ij} \leq T_{\max} - T_{\text{deorb}} \\
 T_L \leq T_{\max} \\
 + \text{products linearization constraints for } z_k, p_{ij}, q_{ij}
 \end{array} \right. \quad (16)$$

with the linearized cost C_L and duration T_L given by :

$$\left\{ \begin{array}{l}
 C_L = \sum_{i,j} s_{ij} C_{0ij} + \sum_{i,j} p_{ij} C_{1ij} + \sum_k s_k C_k \\
 T_L = \sum_{i,j} s_{ij} T_{0ij} + \sum_{i,j} p_{ij} T_{1ij} + \sum_{i,j} q_{ij} T_{2ij} + \sum_k s_k T_k
 \end{array} \right. \quad (17)$$

The numbers of variables and constraints of the linearized problem are :

- $N^2 + 3N$ binary variables : $N(N-1)$ for the s_{ij} and $4N$ for the x_k, y_k, z_k, s_k
- $3N^2-2N$ real variables : $3N(N-1)$ for the $\alpha_{ij}, p_{ij}, q_{ij}$ and N for the τ_i
- $9N^2 - 2N + 3$ constraints + variable bound constraints.

4.5 Initialization

The initialization goals are to pre-optimize the drift orbit parameters in order to start with an already good solution and also to eliminate unfeasible or too expensive transfers of the possible paths in order to reduce the problem size.

4.5.1 Pre-optimization problem

Denoting the initial, drift and final orbits of a single transfer with the respective subscripts 1, d and 2, the pre-optimization consists in minimizing the transfer cost ΔV_{12} in less than a given duration T_0 :

$$\begin{array}{c}
 \text{Min} \\
 a_d, e_d, I_d, t_1, t_2
 \end{array}
 \Delta V_{12}$$

$$\text{s.t.} \quad T_{12} = t_2 - t_1 = \frac{\Omega_2(t_0) - \Omega_1(t_0) + (\dot{\Omega}_2 - \dot{\Omega}_1) \times (t_1 - t_0)}{\dot{\Omega}_d - \dot{\Omega}_2} \leq T_0 \quad (18)$$

Remark 5

The preliminary assessment of the drift duration (Figure 11) has shown that only circular drift orbits could yield reasonable transfer durations and the drift eccentricity e_d has been removed from the global problem formulation. Nevertheless the pre-optimization problems are solved with a free drift eccentricity variable. The optimum is found to be a null eccentricity as expected, validating the assumption in practical cases.

4.5.2 Duration constraint

The SDC mission is composed of 4 transfers to be completed within one year. The average transfer duration is thus 3 months. Since the linearized approximation T_L overestimates the true duration, the duration constraint T_0 must be set to strictly less than 3 months (e.g. 2 months) at the initialization stage, so that the initial solution remains feasible after linearization. The linearized solution will tend progressively to saturate the global duration constraint throughout the iterations by tightening the linearization intervals.

Remark 6

The duration constraint is generally active. Indeed a short duration implies high precession rates differences and expensive drift orbits. The constraint is inactive only if the debris precession rates are sufficiently different so that the RAAN correction is naturally completed on the initial orbits within the prescribed duration.

4.5.3 Solution diagnosis and linearization

For each transfer, the pre-optimization diagnoses if the transfer is unfeasible in the prescribed duration T_0 (this can happen because of the drift altitude bounds) or more expensive than a prescribed cost threshold ΔV_{\max} (in order to keep only the candidate transfers that are compliant with the envisioned mission cost). These transfers are eliminated by fixing the corresponding selection variable s_{ij} to zero.

For a feasible transfer, the pre-optimization yields values of the semi-major axis a_d , the inclination I_d and the initial date t_1 . These values are taken as starting linearization values, with adapted linearization intervals $[\alpha_{\min}; \alpha_{\max}]$ and $[\tau_{\min}; \tau_{\max}]$ compliant with :

- The operational altitude range : $a_{\min} < a_d < a_{\max}$.
- The mission initial and final dates : $t_0 < t_1 < t_2 < t_0 + T_{\max}$.

In order to have a valid linearized modeling of the transfer duration, the drift altitude interval must also lie entirely on the same side of the targeted debris altitude. Indeed the transfer duration function has an asymptote for $a_d=a_2$ and $I_d=I_2$ (drift orbit identical to target orbit) and the linearization is subject to growing errors when one of the bounds $(a_d+\alpha_{\min})$ or $(a_d+\alpha_{\max})$ approaches the limit $a_d=a_2$ (Figure 13). This holds even if I_d is not strictly equal to I_2 , since all debris and drift orbits are in a narrow inclination range.

For each transfer, the side chosen for a_d (lower or higher than a_2) at the initialization stage is thus definitive for the subsequent iterations, since the linearized solution is bounded on one side of the asymptote. Consequently the selected inclination (either $I_d=I_1$ or $I_d=I_2$) is kept unchanged throughout the iterations.

These initial assumptions about the drift altitudes and inclinations are re-examined at the end of the iterations, in order to check their validity with respect to the optimal path found (see §4.6.3).

4.6 Iteration process

4.6.1 Solution update

We denote with a superscript (k) the drift orbit parameters at the k^{th} iteration. The problem is linearized around the local solution $t_i^{(k)}, a_{ij}^{(k)}, e_{ij}^{(k)}, I_{ij}^{(k)}$ using the difference variables $\tau_i^{(k)}, \alpha_{ij}^{(k)}$.

The eccentricity and inclinations are determined once and for all at the initialization step as explained in §3.8 :

$$\begin{cases} e_{ij}^{(k)} = e_{ij}^{(0)} = 0 & \text{(minimum RAAN correction duration)} \\ I_{ij}^{(k)} = I_{ij}^{(0)} & \text{(minimum cost and minimum RAAN correction duration)} \end{cases}$$

The iterative process is depicted on Figure 15.

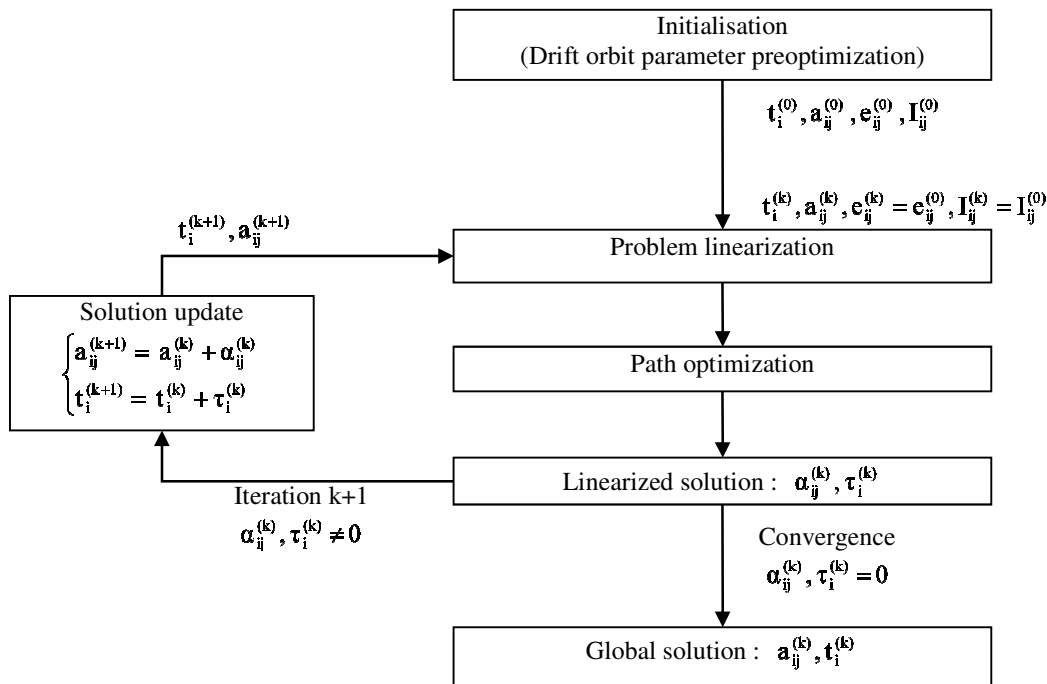


Figure 15 : Iteration process

4.6.2 Convergence

Two successive phases are observed throughout the iterations :

- The first phase starts from the pre-optimized solution with large linearization intervals. During that phase the drift orbit parameters may vary widely at each iteration and the optimal path may change. As long as the path changes, it is better to keep the linearization intervals large, or even not to change the initial ones.
- Once the optimal path stabilizes, the second phase aims at converging accurately on the mission duration constraint. Indeed the linearization leads to an over-constrained solution, since it overestimates the true duration. The convergence is obtained by reducing progressively (for example by halving) the linearization intervals. Although this was not observed on practical cases, the path may again change during that phase : in that case it is better to restore larger linearization intervals until stabilization on a new path.

Remark 7

The set of selected debris issued from the pre-optimized solution is already optimal in most cases, because these debris are the “closest” in terms of RAAN differences. Starting from a less refined initialization requires a higher number of iterations with changes of selected debris, but it does not prevent the process convergence.

Remark 8

Even if the solution stabilization has been observed on practical cases, there is no guarantee that the global optimum has been found. The global optimality of the solution can be ensured in the case of convex cost and constraint functions. For that purpose it would be of interest to study the convexity properties of the transfer cost and transfer duration wrt the drift orbit semi major axis and the transfer starting date. If this convexity could be established in the intervals of variation, the MINLP solution procedure could be extended for example by outer convexification¹⁴ in order to ensure the convergence toward the global optimum.

4.6.3 Solution check

Once the convergence is completed, one must check the solution optimality with respect to the choices made at the initialization stage regarding the side (lower or higher) and the inclination of the transfer orbits. The initial choices were based on a pre-optimization of each transfer with a fixed duration upper bound T_0 . For different values of T_0 , the optimal sense for the RAAN correction may be inverted, and these choices should be changed. If the linearized solution exhibits drift altitudes on their bounds and near the asymptote (Figure 13), it indicates that the iterations should be resumed with updated assumptions. In practical cases, the debris selected on the initial path are close in terms of RAAN values, so that the initial sense of RAAN correction is generally unchanged at the end of the iterations. The preliminary assumptions on the drift altitudes and inclinations are thus optimal at the first attempt.

4.7 Post-optimization

The iterations yield the optimal path (debris selection and order) and the associated cost, considering the specific transfer strategy defined in §3.6 with simplifying assumptions : impulsive manoeuvres, circular drift orbits, pre-optimized drift inclinations, rendezvous at null cost and duration. The post-optimization stage consists in removing these assumptions in order to refine the trajectories with more realistic models, without changing the path. This is a classical optimal control problem without integer variables.

4.8 Algorithms and software

The solution process requires three optimization algorithms respectively for the problem initialization and post-optimization, the linearized problem solution and the Branch and Bound search.

4.8.1 Problem initialization

For every pair of debris, the transfer pre-optimization problem defined by Eq 18 is a small size (3 variables, 1 constraint) nonlinear problem. A generalized reduced gradient method is used. In order to ensure the solution robustness, several optimizations are repeated for the same problem starting from different initialization values. The best result is retained as the initial reference solution for the linearization.

4.8.2 Branch and Bound

For a problem with binary variables, the set of all possible combinations is represented as a binary tree¹¹. A node of the tree is an instance of the optimization problem where part of the binary variables has been fixed either to 0 or to 1. The root node corresponds to the fully relaxed problem, i.e. with the binary variables considered as real ones. Each node is separated into two children by fixing a selected binary variable at 0 or 1. The relaxed instances are of decreasing dimension when going downwards in the tree since more and more binary variables become fixed.

Since the whole tree can potentially be explored, the search strategy must be implemented using as little computer memory as possible²⁵. For that purpose only the active nodes (i.e. the nodes still not fully separated into their two children) are stored in a stack. Each node is dynamically allocated as a derived data type, containing the relaxed problem data and a pointer linking the node either to its parent (depth search) or to the next best one in the stack (breadth search). The linked list of active node starts with only one node (root node), is updated by eliminations and/or separations as the search goes on, until it becomes empty when the tree exploration is completed.

The binary tree is explored downwards from the root node^{8,11}. Separating a node consists in solving the associated relaxed problem, with a part of the binary variables fixed to 0 or 1, while the others are treated as real variables.

After the relaxed problem is solved, the following situations may occur :

- The relaxed problem yields a feasible solution with respect to the integrity constraints. This solution is compared to the best feasible solution already available from the previously separated nodes. If better, this

new best solution is stored replacing the previous one. There can be no better solution in the branches outgoing from that node. The node is pruned and all its outgoing sub-branches are cut off.

- The relaxed problem is unfeasible because the fixed binary variables are not compliant with the constraints, or the relaxed solution is worse than the reference best solution stored. There can be no better solution in the branches outgoing from that node. The node is pruned and all its outgoing sub-branches are cut off.
- The relaxed problem yields a solution that is neither feasible with respect to the integrity constraints nor worse than the reference best solution. The node is then separated into two children. Each child is a copy of the parent node with one selected variable (the separation variable) being fixed respectively to 0 and 1.

Having a good reference solution as soon as possible is desirable in order to speed up the solution by branches cut offs. For that purpose a greedy solution is built at the pre-optimization stage by selecting the best arrangement meeting the mission constraint. This solution is stored as the initial reference solution.

The efficiency of the Branch and Bound method depends on the tree exploration strategy (“Branch”) and the node evaluation function (“Bound”) :

- The exploration strategy specifies the way to choose the next node to separate and the separation variable. The depth search strategy consists in separating always the most downward node with the hope to reach quickly a bottom node of the tree and issue a first reference solution. The breadth search strategy consists in separating the active node with the best evaluation hoping that the best feasible solution lies in a sub-branch of that node (Figure 16). Several possible choices of the separation variable are among others : by initial order (the most simple), or by constraints (the variable appearing in the maximum number of constraints), or by cost penalty (the variable producing the largest variation in the cost function when changed from 0 to 1).
- The separation rules (nodes and variables) must be tried case by case in order to assess their practical efficiency on a given problem. The number of nodes examined before issuing the problem solution may vary by large factors depending on these choices and on the problem features.

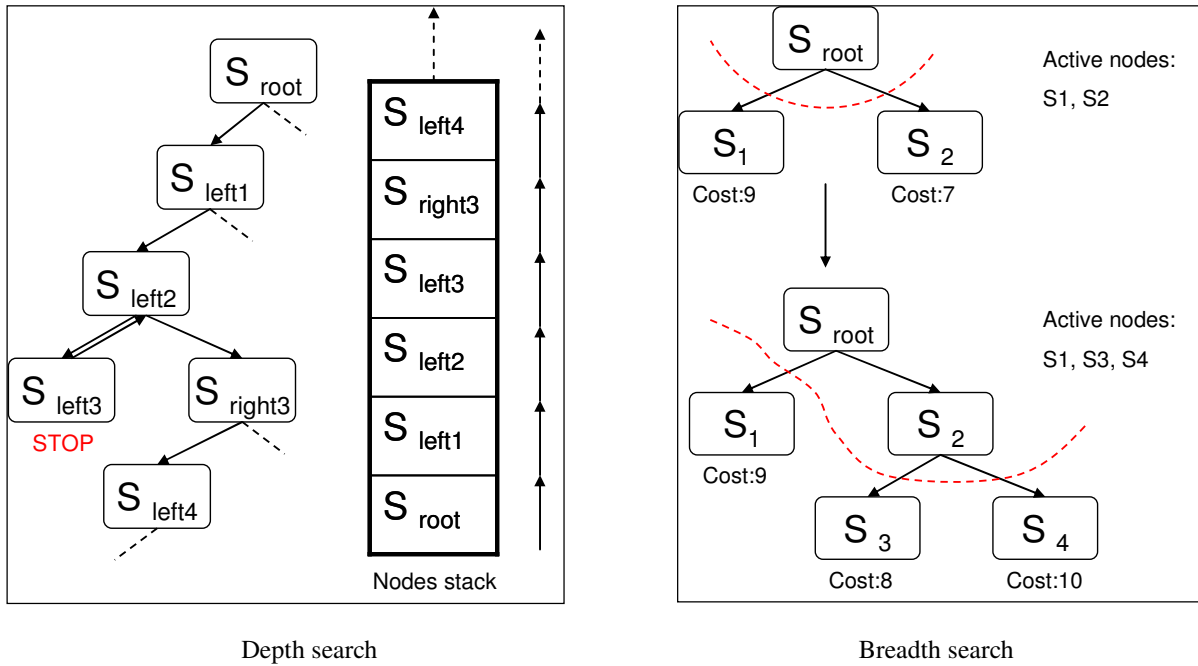


Figure 16 : Depth and Breadth search strategies

The node evaluation (or pricing) aims at assessing the best solution that could be found among all the node children. There is a compromise to make between the time to evaluate the node, the evaluation accuracy and the robustness. The goal is to assess as precisely as possible the best solution contained in the node, without consuming an excessive computation time. Solving the node relaxed problem by linear programming is the most robust approach, and it favours the linearization approach chosen for the SDC problem.

For the SDC mission study, several exploration strategies have been tried in order to select the most efficient one. On the practical run cases the breadth search strategy coupled with the separation on the most constrained variable has given the best performance in terms of number of nodes assessed. The depth and breadth strategies performances are compared for the application case presented in Chapter 5.

4.8.3 Linear programming

Three algorithms are used within the Branch and Bound process to solve the successive relaxed linear problems :

- A primal simplex to solve the root problem at the top of the tree, when starting from scratch.
- A dual simplex to solve the successive nodes. The parent node solution is used to restore a feasible dual basis, taking into account the fixing of the binary variable chosen to separate the node²⁰. This warm-start procedure avoids the risk of failure when searching an initial feasible basis for the current node. It results in robustness and computation time gains in the tree exploration.

- An interior point method as backup algorithm. When the simplex fails (about 1% of the cases), either because of numerical rounding errors or degeneracy, the interior point solver is called to restart the node solution. Similarly to the dual simplex, the parent node solution is retrieved as initialization. The interior point solver is generally more robust, but slower and less accurate than the simplex on medium size problems.

Several reduction techniques²¹ are applied to decrease the size of the linear problem before trying to solve it:

- Some binary variables can be fixed by constraints elimination.
- The dual variables at the parent node solution provide a lower bound on the current node cost. This allows either directly pruning the current node or fixing some binary variables depending on their multipliers.

4.8.4 Software

The generalized reduced gradient software used for the pre- and post-optimization stages has been developed at Astrium Space Transportation years ago and is largely used for mission analysis studies.

The Branch and Bound software with the linear programming solvers have been developed from scratch for the present application, in Fortran 90. Efficient free or commercial softwares like X-Press, OSL or CPLEX^{13,14,26} are available that solve mixed integer linear programming problems. In particular for convex MINLP problems several software packages ensure the convergence toward the global optimum. The drawback is that they generally work as black-boxes, making it difficult to diagnose correctly the problems that occur (erroneous formulation, inefficient exploration strategy, insufficient node evaluation robustness, numerical accuracy issues,...). Also the convexity of the cost and constraint functions as formulated in (10) has not been studied so that there is not guarantee that these softwares would behave correctly.

Mastering the entire source code allows one to first check that the problem is correctly formulated and solved, then to upgrade the solution strategy in order to solve as efficiently as possible the specific application considered. Also the iteration and linearization process has been automated, so that practical cases are solved with very limited manual interventions.

4.9 Assumptions and simplifications recap

The method presented in the previous sections yields a valid solution (from the optimality point of view) under the following assumptions :

- The mission starts from the first debris selected on the optimal path. The cost and duration required to reach that debris are null, assuming that a launcher has performed the required manoeuvres. It is possible to release this assumption, by adding a fictitious starting debris located on the launcher injection orbit.
- The vehicle uses a high thrust engine, so that the manoeuvre durations are negligible with respect to the coast arc durations. An impulsive modeling is then representative and the problem becomes of finite dimension.
- The duration constraint is weak, and a drift strategy allows performing the RAAN corrections at null cost.

- The durations of the debris operations (capture, deorbitation, release) are fixed and identical for all debris.
- The drift orbits are circular, with their inclinations fixed at the pre-optimization phase. The correctness of the inclination choice is checked a posteriori on the solution. Allowing non null eccentricities could improve the global optimum, at the expense of increasing the problem dimension.

The contributions of the mission phases to the global cost and duration are summarized on Table 3.

	Cost	Duration
Transfer manoeuvres	Counted	Neglected
Drift phases	Null	Counted
Debris operations	Counted	Neglected

Table 3 : Cost and duration contributors

4.10 Current status and further work

The solution method has been automated. The initialization, problem linearization and Branch and Bound solution are processed sequentially and iterated until convergence. The user needs only to check the validity of each iteration result before running the next one, and if necessary restrict the linearization intervals. Medium size problems like the one presented in Chapter 5 can be solved in a short time, allowing sensitivity assessments to the mission data.

The solution method described in this document applies only to a high thrust vehicle, when the powered manoeuvres can be modeled as velocity impulses. In this scope the enhancements envisioned are the following :

- Allowing non circular drift orbits may reduce the mission cost. This additional degree of freedom leads to an increase of the problem dimension and it will slow down the Branch and Bound solution. Presently the targeted debris are on LEO and SSO. Before modifying the transfer strategy for such debris, it must be checked on a continuous problem (fixed path) if significant performance gains can be expected from elliptic drift orbits. On the other hand, if a SDC mission is envisioned for debris on elliptic orbits, like launcher upper stages left on Geostationary Transfer Orbits (GTO), this enhancement becomes necessary.
- Considering non impulsive manoeuvres is possible at the initialization stage. The linearized cost and duration function resulting from the transfer pre-optimization will then be more representative of the vehicle thrust capabilities. The same solution method can be retrieved as long as the transfer durations remain small with respect to the drift durations. If not, a new formulation must be devised taking into account the coupling between the transfer and the drift phases since a part of the RAAN correction is achieved during the transfer.
- In the case of a deorbitation manoeuvre performed by the vehicle after the debris capture, the vehicle has to return on a stable orbit before starting the next transfer. This orbit should be optimized within the global mission optimization process.

For a low thrust vehicle, the durations of the transfer and drift phases become of the same order of magnitude. They can no longer be considered as uncoupled, since a part of the RAAN correction is realized during the transfer. Also the rendezvous constraint is no longer negligible in terms of duration and cost. A different transfer strategy and solution method has to be devised for this category of vehicles.

5 Application example

The solution method is illustrated with an application case consisting in selecting $n=5$ debris to deorbit among a list of $N=11$. The deorbitation is assumed to be performed by the debris themselves with a device supplied by the vehicle. The problem initial dimension is 154 binary variables, 341 real variables and 1070 constraints.

5.1 Debris orbits

The debris are on Sun-Synchronous Orbits with regularly spaced inclinations ranging from 98 to 99 deg, and initial RAAN ranging from 160.2 to 235.0 deg at the mission beginning. These orbits are representative of end of life operational satellites dedicated to Earth observation missions. For each orbit the semi-major depends on the inclination (Equation 5 with $\dot{\Omega}=0.986$ deg/day). The debris orbits are given in Table 4.

	Semi-major axis (km)	Eccentricity (-)	Inclination (deg)	Initial RAAN (deg)
Debris 1	7030.5	0.0001	98.0	221.1
Debris 2	7055.3	0.0001	98.1	188.3
Debris 3	7080.0	0.0001	98.2	164.4
Debris 4	7104.4	0.0003	98.3	235.0
Debris 5	7128.5	0.0000	98.4	174.7
Debris 6	7152.5	0.0001	98.5	194.1
Debris 7	7176.3	0.0001	98.6	149.0
Debris 8	7200.0	0.0001	98.7	180.3
Debris 9	7223.2	0.0002	98.8	200.6
Debris 10	7246.4	0.0001	98.9	191.0
Debris 11	7269.3	0.0003	99.0	160.2

Table 4 : List of 11 candidate debris

Remark 9

The solution methodology has been applied in the same way for debris orbits that do not have the sun-synchronism property. The above example is motivated by the fact that most observation satellites are on SSOs, so that a future SDC vehicle should be first designed for such debris.

Remark 10

Taking into account the deorbiting manoeuvre (if the vehicle has to deorbit the debris) would not change the optimal path. Indeed the debris being on very close circular orbits, the deorbitation costs are quasi identical for all of them, and they do not influence the debris choice.

The drift orbit altitudes are bounded in the range 400 km – 1200 km. The duration for the debris operations is assumed to be null : $T_{\text{deorb}}=0$.

5.2 Initialization

Every transfer from any debris to any other is pre-optimized for a fixed duration of 2 months. Once all the optimizations are completed, the unfeasible transfers are discarded from the problem by setting their selection variables s_{ij} to 0. The remaining feasible transfers are linearized around their solution in order to initialize the iteration process for the Branch and Bound. This pre-processing allows a reduction of the problem dimension from 500 to 200 variables and from 1000 to 500 constraints.

At this initialization stage, with fixed transfer durations set to 2 months, the optimal path would be (5→8→2→10→6) , for a total ΔV of 711 m/s and a total duration of 8 months (2 months per transfer). The intermediate orbits on this initial path are presented in Table 5. This 2 months allocation per transfer does not saturate the global duration constraint of one year, and it thus gives some freedom to the subsequent iterations in order to converge toward the optimal solution. Fixing this initial allocation too high (e.g. slightly less than 3 months) with the hope to reduce the number of iterations would in fact freeze the initial path.

It can be observed at this initialization stage that the mission cost is driven by the RAAN differences rather than the inclinations or the altitudes. The best strategy consists in selecting the debris with the minimum RAAN differences in order to have drift orbits as close as possible to the debris orbits. The drift orbits are lower than the debris orbits, with higher precession rates, so that the vehicle catches the debris forwards. The debris are therefore ordered by increasing RAAN values. This can be considered as a “rule of thumb” in order to guess a quite good solution.

Initial path	Semi-major axis (km)	Inclination (deg)	Cost ΔV (m/s)	Duration ΔT (days)
Debris 5	7128.5	98.4	0	0
Drift 5 → 8	7019.6	98.7	173.3	61
Debris 8	7200.0	98.7	0	0
Drift 8 → 2	6947.9	98.7	246.9	61
Debris 2	7055.3	98.1	0	0
Drift 2 → 10	7140.9	98.9	176.5	61
Debris 10	7246.4	98.9	0	0
Drift 10 → 6	7125.5	98.9	114.1	61
Debris 6	7152.5	98.5	0	0
Total			710.8	244

Table 5 : Initial path

5.3 Iterations

The initial linearized problem is solved by a Branch and Bound method starting from the pre-optimized path as feasible initialization. The depth and breadth search strategies are tried at this first iteration in order to determine the most efficient one for the further iterations. The numbers of nodes assessed are respectively 135 for the depth search and 99 for the breadth search. The difference is explained by the fact that an already good reference solution is provided by the initial path. The depth search whose goal is to find a feasible solution as soon as possible, while neglecting node elimination (opposite to the breadth search) is therefore less efficient. The solutions issued from the successive iterations are presented in Table 6.

Iteration	Number of nodes	Path	Cost ΔV (m/s)	Duration ΔT (days)
Initial		5 → 8 → 2 → 10 → 6	710.8	244.0
1	99	5 → 8 → 2 → 10 → 6	652.3	269.3
2	53	5 → 8 → 2 → 10 → 6	594.8	299.1
3	41	5 → 8 → 2 → 10 → 6	540.7	335.7
4	33	5 → 8 → 2 → 6 → 10	508.0	363.3
5	14	5 → 8 → 2 → 6 → 10	502.8	364.1
6	13	5 → 8 → 2 → 6 → 10	500.7	366.0

Table 6 : Iterations

The set of selected debris issued by the pre-optimization is unchanged, but the optimal path changes at the 4th iteration with a permutation of the last two debris, and becomes (5→8→2→6→10). This behavior is observed in most practical cases (cf. §4.6.2 Remark 7). In cases with debris very near to each other, there may be changes in the selected debris set from one iteration to the other, and the optimized set may be different from the initial one. In such cases, the cost improvement due to the debris change is generally very small.

A total ΔV of 501 m/s is required for a total mission duration of 12 months (maximal duration allowed).

The iterations require less and less node evaluations, while the solution approaches the optimum, because the initial reference solution is better and allows more efficient cut offs. The saturation of the duration constraint is expected since taking as long time as possible for the RAAN correction allow the drift orbits to be closer to the debris orbits, and thus minimizes the transfer costs.

5.4 Optimal solution

The optimized solution yields a total ΔV of 501 m/s. The solution that would be obtained at the initialization stage with a 3 months allocation per transfer yields a total ΔV of 517 m/s. This solution corresponds to the rule of thumb : “Order the debris by RAAN initial values and select the closest ones”. Although the optimal path may be different,

this rule yields a good debris selection so that the problem reduces to optimizing the drift durations. A correct order of magnitude for the required ΔV can thus be assessed from a purely continuous NLP problem.

The intermediate orbits on the optimal path are presented in Table 7. It is observed that the optimal debris order is no longer by increasing RAAN values. It proves more favorable to change the sense of the RAAN correction for the last transfer, in order to reduce the inclination change manoeuvres.

If the vehicle has to deorbit the debris, the deorbitation manoeuvre must lower the perigee inside the atmosphere in order to ensure a natural fall out and the vehicle must afterwards restore a stable orbit. The order of magnitude of this manoeuvre is 200 m/s per debris, so that the total mission cost increase amounts to 1000 m/s. This value is conservative since it assumes that the vehicle returns on the initial orbit after the deorbitation manoeuvre is achieved and before starting the next transfer. An improvement of this strategy would consist in returning on an optimized orbit (instead of the initial orbit) before the next transfer. The parameters of this orbit should then be added to the global problem formulation.

Optimal path	Semi-major axis (km)	Inclination (deg)	RAAN change (deg)	Cost ΔV (m/s)	Duration ΔT (days)
Debris 5	7128.5	98.4		0	0
Drift 5 \rightarrow 8	7090.6	98.7	+3.6	107.9	103.0
Debris 8	7200.0	98.7		0	0
Drift 8 \rightarrow 2	7042.6	98.7	+8.0	165.2	100.8
Debris 2	7055.3	98.1		0	0
Drift 2 \rightarrow 6	7028.2	98.5	+5.8	126.4	92.8
Debris 6	7152.5	98.5		0	0
Drift 6 \rightarrow 10	7247.7	98.5	-3.1	101.2	69.4
Debris 10	7246.4	98.9		0	0
Total				500.7	366

Table 7: Optimal path

The vehicle trajectory (transfer and drift orbits), the RAAN evolution (difference between the vehicle RAAN and the debris RAAN), and the mission cost (before and after optimization) are plotted on Figure 17, 18 and 19.

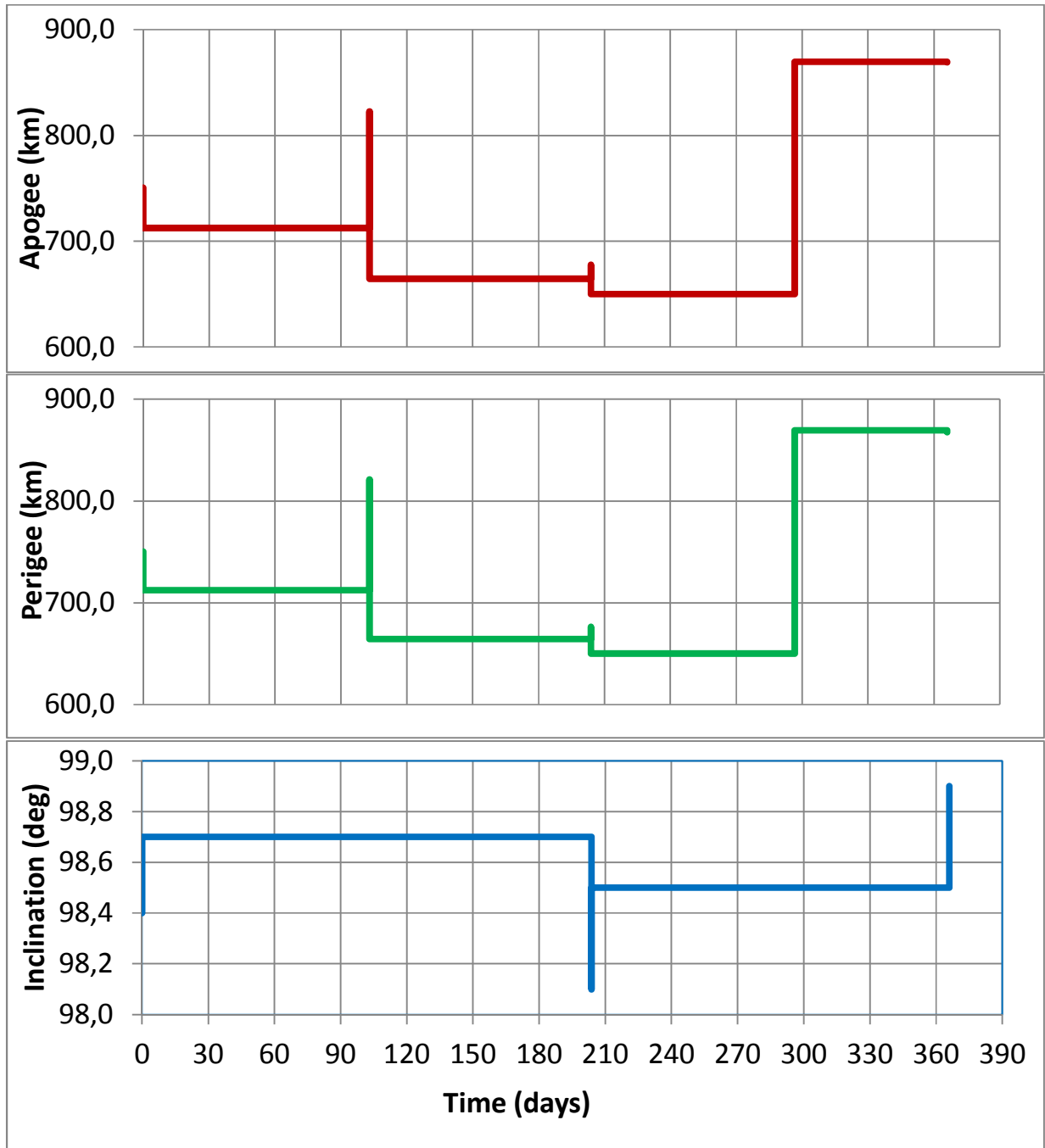


Figure 17 : Optimal path orbits

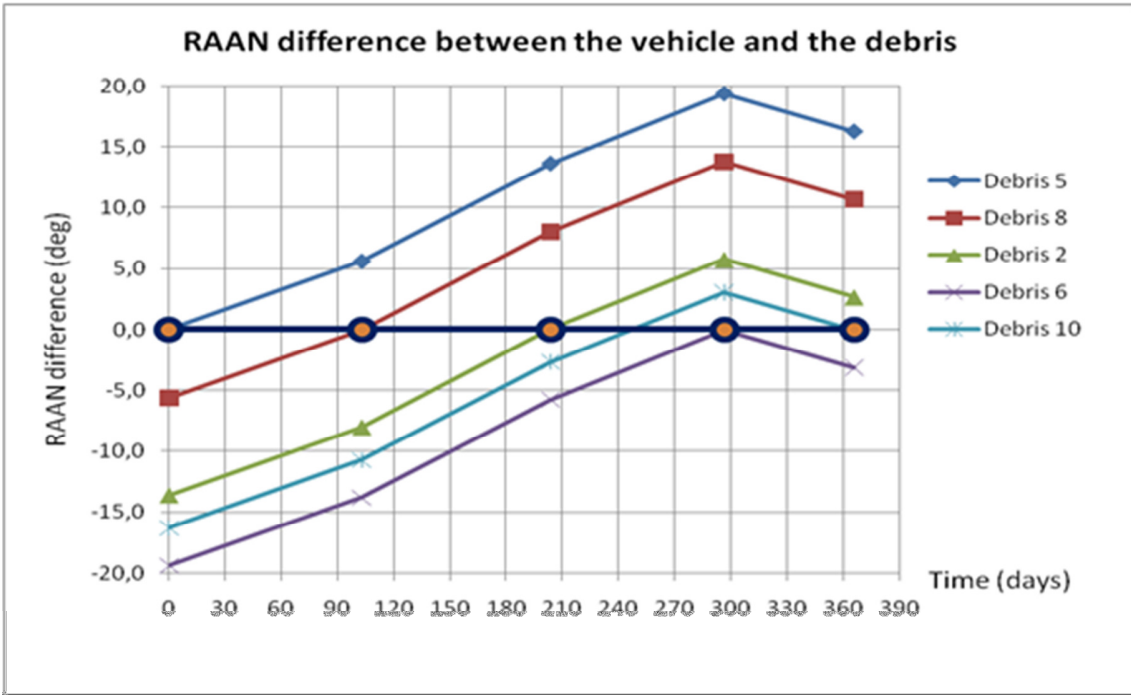


Figure 18 : RAAN correction

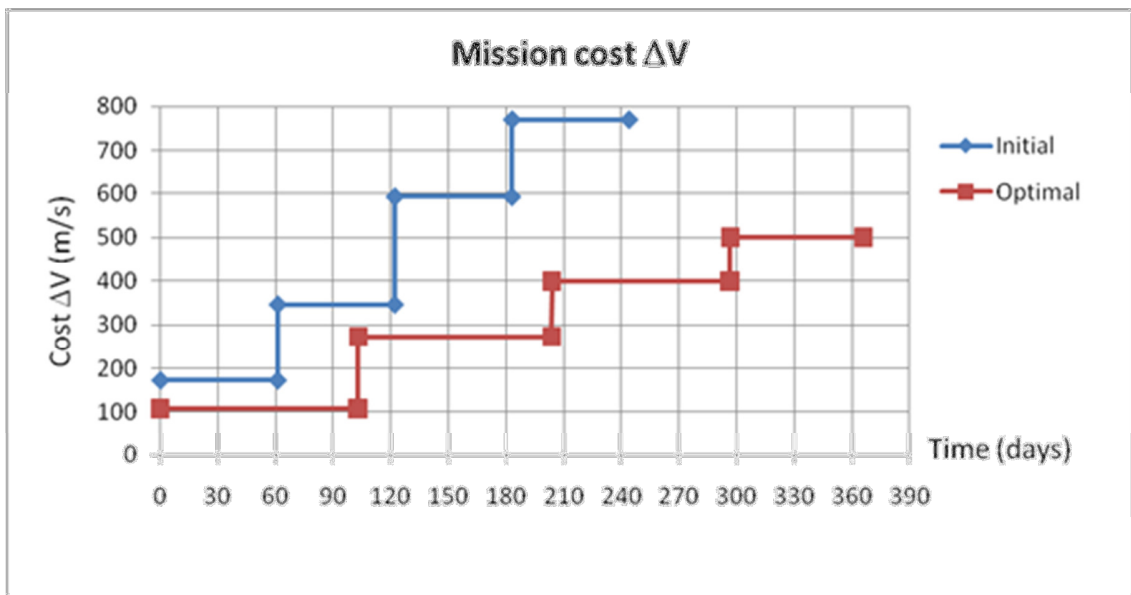


Figure 19 : Mission cost before and after optimization

We observe from Table 7 that roughly 20m/s – 30m/s per degree of RAAN change are required for each transfer. These costs are much lower than those required by a direct change strategy (§3.2) that would amount to 130m/s per

degree of RAAN change (§3.2). On this application example dealing with SSO debris, the overall mission cost is decreased by a factor 5 between the direct strategy and the drift strategy.

5.5 Computation time

All computations have been performed on Sun Solaris OS 5.10 workstation. The initialization stage consists in pre-optimizing the $11 \times 10 = 110$ transfers from any debris to any other. These 110 optimizations are achieved in about 2 minutes. Each iteration to solve the mixed integer linear optimization problem by the Branch and Bound method requires then about 15 minutes. The linearized solution is used to move the reference solution and re-linearize the problem around the new reference. The user's main tasks consist in checking that the new reference solution reduces the cost, and updating the linearization bounds depending on the iteration result (§4.6). In the case of an unsuccessful iteration (i.e. the new reference solution does not reduce the cost), the previous iteration must be restarted with lower bounds until getting an effective improvement. With these computation times, and including the checks of the intermediate linearized solutions, it is possible to solve medium size problems from scratch within typically two days.

6 Conclusion

This document proposes a method to optimize simultaneously the debris selection and the trajectories in view of a Space Debris Collecting mission. A specific transfer strategy with impulsive manoeuvres is defined so that the problem becomes of finite dimension. This MINLP problem is then linearized around an initial reference solution in order to apply a Branch and Bound algorithm. The process is iterated until the solution stabilizes on the optimal path. Although this stabilization has been observed in practice, there is no proof that the solution found correspond systematically to the global optimum. A way to investigate the global optimality would be to study the convexity of the cost and constraint functions. In that case the convergence toward the global optimum could be ensured by adding outer convexification constraints to the successive MILP problems, as is done in existing MINLP software packages.

The method is theoretically applicable whatever the numbers of debris (total number and number to deorbit) and whatever the mission duration. The practical limitation comes from the problem size and the associated computation time that grow exponentially with the number of candidate debris. The initialization procedure includes a filtering of the unfeasible or too expensive solutions, allowing thus a subsequent size reduction and a time-efficient numerical solution. Medium size problems (with typically 10 to 20 debris) can thus be tackled efficiently with this method. For larger size problems, it is necessary to turn to stochastic programming methods in order to get approximate solutions within reasonable computation times.

An application case consisting in selecting 5 SSO debris among a list of 11 candidates is presented. The method proves reliable and an optimal path is issued in a few iterations. The optimal solution is slightly better than the one guessed by applying the rule of thumb : "Order the debris by RAAN initial values and select the closest ones". This intuitive rule reduces the problem to optimizing the drift durations. The order of magnitude of the required ΔV can thus be quickly assessed.

The optimal path can then be used as a basis for a more detailed mission analysis, taking into account the vehicle features (thrust level) and the operational constraints (rendezvous and deorbitation operations). Sensitivities to the mission main constraints (duration, altitude bounds) can also be easily issued in order to support design trade-offs for a future SDC vehicle.

The main enhancements envisioned are to allow non-circular drift orbits and to account for non-impulsive manoeuvres in the cost and duration assessment. The two interests of allowing elliptical drift orbits are : to reduce the SDC mission cost for LEO and SSO debris, and to apply the method to GTO debris which are on highly elliptical orbits. Considering continuous instead of impulsive thrust is more representative of the vehicle capabilities when assessing the mission cost. This is possible without changing the solution method as long as the transfer durations remain small with respect to the drift durations. For very low-thrust engines, the transfer and the drift phases become highly coupled, since a part of the RAAN correction is realized during the transfer. Also the rendezvous constraint is no longer negligible in terms of duration and cost and it must be explicitly taken into account. A specific transfer strategy has to be devised for this category of vehicles.

References

1. K. Klinkrad, (s.d.), *Space debris models and risks analysis*, ISBN: 3-540-25448-X .
2. J. Liou, (2008), *An assessment of the current LEO debris environment and what needs to be done to preserve it for future generations*, International Conference on Orbital Debris Removal, Brazil, Chantilly, Virginia.
3. *Position paper on space debris mitigation*, (2005). International Academy of Astronautics.
4. R. Walker, C.E. Martin, P.H. Stokes, J.E. Wilkinson, (2000), *Studies of space debris mitigation options using the debris environment long term analysis (DELTA) model*, 51st International Astronautical Congress, Brazil, (pp. IAA-00_IAA.6.6.07), Rio de Janeiro.
5. B. Bastida Virgili, H. Krag, (2009), *Strategies for active removal in LEO*, 5th European Conference on Space Debris, Germany, ESOC Darmstadt.
6. G. Leitmann, (1962), *Optimization techniques*, Mathematics in science and engineering, Volume 5.
7. E. Trélat, (2005), *Contrôle optimal - Théorie et applications*, Vuibert.
8. P. Lacomme, C. Prins, M. Sevaux, (2003), *Algorithmes de graphes– 2ème Edition*, Eyrolles.
9. J.J. Schneider, S. Kirkpatrick, (2006), *Stochastic optimization*, Springer.
10. G.L. Nemhauser, A.H.G Rinnooy Kan, M.J. Todd, (1989), *Optimization*, Handbooks in operations research and management science, Volume 1, Elsevier.
11. M. Minoux, (2008). *Programmation mathématique - Théorie et algorithmes – 2ème Edition*, Editions Tec&Doc – Lavoisier.
12. J. Dréo, A. Pérowski, P. Siarry, E. Taillard, (2003), *Métaheuristiques pour l'optimisation difficile*, Eyrolles.
13. <http://www.gamsworld.org/minlp/solvers.htm>
14. M.R. Bussieck, S. Vigerske, (2011) *MINLP software solvers* ,Published on Wiley Online Library, DOI: 10.1002/9780470400531.eorms0527.
15. H.J. Oberle, K. Taubert, (1997), *Existence and multiple solutions of the minimum-fuel orbit transfer problem*, J. Optim Theory Appl. 95 , 243-262.
16. J. Gergaud, T. Haberkorn, (2007), *Orbital transfer : some links between the low-thrust and the impulse cases*, Acta Astronautica, 60, no. 6-9 , 649-657.
17. E. Minieka, (1974), *On computing sets of shortest paths in a graph*, Management science and operations research, Volume 17, Number 6.
18. D. Eppstein, (1997), *Finding the k shortest paths*, 35th IEEE Symp. Foundations of Comp. Sci., Santa Fe, 1994, pp. 154-165. Tech. Rep. 94-26
19. J. Hershberger, M. Maxel, S. Suri, (1997), *Finding the k shortest simple paths : a new algorithm and its implementation*, ACM Transactions on Algorithms (TALG), Volume 3 Issue 4, November 2007
20. A. Sedenko-Noda, C. Gonzalez-Martin, (2010), *On the k shortest path trees problem*, European Journal of Operational Research 202 (2010) 628-635
21. A. Billionnet, (2007), *Optimisation discrète*, Dunod.
22. C. Chao, (2005), *Applied orbit perturbation and maintenance*, The Aerospace Press.
23. V. Chobotov, (2002), *Orbital mechanics - Third Edition*, AIAA Education Series.
24. J. Murakami, S. Hokamoto, (2010), *Approach for optimal multi-rendezvous trajectory design for active debris removal*, 61th International Astronautical Congress, (pp. IAC-10.C1.5.8). Prague, CZ.
25. S. J. Chapman, (2004), *Fortran 90/95 for Scientists and Engineers - Second Edition*, McGraw Hill.
26. C. Guéret, C. Prins, M. Sevaux, (2003), *Programmation linéaire – 2ème Edition*, Eyrolles.