Optimisation and optimal control methods for planet sequence design of low-thrust Interplanetary transfer problems with gravity-assists

Joris Olympio

MINES PARIS
ParisTech

ED n 84 : STIC

THESE

pour obtenir le grade de
Docteur de l'Ecole des Mines de Paris
Spécialité: Informatique temps-réel, robotique et automatique

présentée et soutenue publiquement par
Joris T. OLYMPIO

le 27 Octobre 2008

Optimisation and Optimal Control Methods for Planet Sequence Design of Low-Thrust Interplanetary Transfer Problems with Gravity-Assists

Directeur de thèse : Yves ROUCHALEAU

Jury

M. Pierre BERNHARD	Université de Nice - Sophia Antipolis	(Examinateur)
M. Régis BERTRAND	Centre National d'Etudes Spatiales	(Examinateur)
M. Guido COLASURDO	Politecnico di Torino	(Rapporteur)
M. Thierry DARGENT	Thales Alenia Space	(Examinateur)
M. Yves ROUCHALEAU	Ecole des Mines de Paris	(Examinateur)
M. Johannes SCHOENMAKER	European Space Agency	(Examinateur)
M. Emmanuel TRELAT	Université d'Orléans	(Rapporteur)

Whether you believe you can do a thing or not, you are right.

Henry Ford

Unless you try to do something beyond what you have already mastered, you will never grow

Emerson

When a distinguished but elderly scientist states that something is possible, he is almost certainly right. When he states that something is impossible, he is very probably wrong.

Arthur C. Clarke, first law

To my Parents, my Brothers
my Friends,
and everyone this work might inspire.

Acknowledgements

Ce travail de thèse a été financé par Thalès Alenia Space, et le Centre National d'Etude Spatial (CNES). Je remercie Thierry DARGENT (Thales Alenia Space) de m'avoir permis de terminer mon travail de thèse dans son département, et Régis BERTRAND (CNES) pour ses remarques toujours très pertinentes sur mes travaux, me permettant d'avoir un autre regard sur mes résultats.

Je remercie également Jean-Paul MARMORAT et plus particulièrement Yves ROUCHALEAU pour avoir encadré mon travail de thèse dans de bonnes conditions.

Merci aux membres du jury, rapporteurs et examinateurs. A Guido COLASURDO, professeur à la Politecnico di Torino, et Emmanuel TRELAT, professeur à l'Université d'Orléans, d'avoir accepté la difficile tâche de rapporteur dans mon jury de thèse. Leurs remarques et questions m'ont été particulièrement utiles.

A Pierre BERNHARD, professeur à l'Université de Sophia Antipolis, d'avoir accepté de présider le jury de ma thèse. Johannes SCHOENMAKER, ingénieur à l'Agence Spatiale Européenne (ESA/ESOC), Régis BERTRAND, Thierry DARGENT et Yves ROUCHALEAU pour avoir accepté d'être examinateurs dans le jury, en tant spécialistes de la mécanique spatiale, ou des méthodes d'optimisation.

Il convient de remercier l'Ecole des Mine de Paris, et les membres du Centre de Mathématiques Appliquées. Plus particulièrement, je remercie Gilles et Dominique pour leur gentillesse, leur grande disponibilité et leur sérieux, notamment lors des petits couacs de l'informatique, ou lors de démarches administratives.

Enfin, je n'oublie pas tous les doctorants du laboratoire: Jan, Lionel, ... pour l'ambiance de travail. Également, un grand merci à l'équipe RT/PS de Thales Alenia Space (Cannes) pour m'avoir accueillit dans leur équipe durant ma dernière année. Notamment pour l'ambiance agréable dans le bureau: Guillaume pour avoir toujours l'anecdote marrante et dont les qualités de photographe m'ont bluffé, Fabien pour sa sympathie et les discussions passionnées en F1, Grégory avec qui j'ai pu échanger des idées sur mes travaux, mais aussi Florent, Laurent, Isabelle, Catherine, Jean-Paul, Julien, Xavier, Thomas et Damiana. Je leur souhaite à tous, la plus grande réussite dans leurs projets futurs.

Je terminerais en remerciant mes amis qui m'ont permis, parfois difficilement, de me déconnecter de mon travail de thèse...

Résumé (*in French*)

Cette thèse porte sur la conception de trajectoires interplanétaires, à poussée faible. Les systèmes de propulsion électriques, à poussée faible ou continue, ont permis d'accroitre significativement les possibilités de trajectoires, au détriment de mission plus longues. La poussée faible limite également la manoeuvrabilité du système. Afin de parer à ces inconvénients, on utilise généralement des manoeuvres d'assistances gravitationnelles, pour ainsi réduire la consommation et la durée de transfert de la sonde.

Le rôle de l'analyste mission est donc de déterminer le meilleur scénario (la séquence de planètes à visiter). De nos jours, ce problème est résolu de manière expérimentale et heuristique. Cependant, bien que la trajectoire produite soit optimale à scénario donné, il n'y a aucune garantie que le scénario en lui-même soit optimal. De plus, cette approche est relativement fastidieuse.

Notre objectif a donc été de mettre en place des outils et méthodes permettant de trouver des scénario optimaux pour un objectif fixé.

Durant cette thèse, nous avons suivit 2 approches.

La première approche consiste à considérer le problème comme étant un problème d'optimisation globale, à variables discrètes. Un ensemble de scénario est étudié à priori. Pour simplifier et faciliter la recherche de séquences, on a modélisé le problème de transfert à poussée faible, en utilisant un principe d'inversion dynamique. Ce modèle utilise des arcs balistiques pour minimiser la consommation, et introduire des degrés de liberté supplémentaires pour satisfaire des contraintes terminales. On a mis au point un algorithme de complexité polynomiale pour résoudre le problème. Afin de réduire le coût calculatoire, nous avons mis en place des contraintes de " pruning " permettant de réduire l'espace de recherche.

La deuxième approche consiste à formuler le problème comme un problème de commande optimale, où la dynamique inclut les principaux corps perturbateurs. Le scénario est alors déterminé à postériori. On résoud numériquement le problème au N corps. On montre que les méthodes indirectes (Pontryaguin) et directes (Collocation, Transcription) ne nous permettent pas de résoudre ce problème. On a donc mis au point un solveur de deuxième ordre respectant à la fois les conditions d'optimalité et de précision connues des méthodes indirectes, et des propriétés de robustesse généralement attribué aux méthodes directes.

Mots clés: optimisation, commande optimale, optimisation globale, trajectoire interplanétaire, poussée faible, modélisation de trajectoires, inversion dynamique, assistance gravitationnelle, swing-by, contrainte de pruning, méthodes indirectes, calcul de variation.

Abstract

Electrical propulsion systems allow broader a class of spacecraft trajectories than conventional chemical propulsion. But because low-thrust propulsion systems induce limited controllability for the spacecraft and increase overall trajectory transfer duration, it is generally useful to use gravitational assistances. Gravity assists allow reducing both the consumption and the mission duration.

The optimisation of continuous thrust trajectory remains a terrible task. General methods may be difficult to converge. In addition, the optimisation of the scenario, i.e. seeking the optimal planet sequence for the gravity assists, is never included in the optimisation process. The planet sequence is very likely not optimal. Mission analysts usually consider several different planet sequences. The trade-off between the different solutions permit identifying a promising planet sequence. This approach is however time consuming.

The purpose of this thesis is to provide method to design optimal scenarios.

This thesis proposes methods for the determination of the optimal scenario. Two approaches have been considered.

The first approach considers the problem as an integer programming problem, when many sequences are considered a-priori. A low-thrust trajectory model has been designed, using inversion dynamics approach, to compute efficiently approximate solutions to the low-thrust trajectory transfer problem. This low-thrust model uses coast arcs to minimise the consumption, but also to increase the degree of freedom for satisfying terminal constraints. We set up an algorithm with polynomial complexity to solve the multi-gravity-assist low-thrust problem. The computational cost is limited using pruning constraints to reduce the size of the search boxes.

The second approach formulates the problem as an indirect continuous optimal control problem. The dynamic includes all major gravitational bodies. Swing-bys are not introduced with intermediate constraint, but implicit with the dynamic and the appropriate control. We show that usual direct and indirect methods have difficult convergence for this problem. Using a second order gradient method, we seek the optimal control that transfers the spacecraft to its destination while in a multi-body dynamics. In some case, the optimal control manages to introduce gravity assist on the trajectory. The scenario is then given a posteriori, once an optimal control has been found.

Keywords: optimisation, optimal control, global optimisation, interplanetary trajectory, low thrust, trajectory modelling, dynamic inversion, gravity assist, swing-by, pruning constraint, indirect method, calculus of variations.

Introduction

Motivations

The advance in low-thrust propulsion systems makes low-thrust trajectory quite appealing. Also, the optimisation of low thrust trajectories is a demanding task for the mission analyst. However, although low-thrust propulsion systems are, so far, the most practical and efficient way to travel into space, they present limited manoeuvrability. Their use increases mission duration. At this point, gravitational assistance is of particular importance. The use of planet' gravity to propulse quicker and further the spacecraft reduces the overall fuel consumption and the mission time of flight.

There are plenty of methods to optimise and solve interplanetary space trajectory problems. Bertrand [Ber01] provides an efficient way for solving low-thrust interplanetary trajectory problems. His approach provides the optimal trajectory for a given planet sequence. For a particular mission, one can wonder whether the mission is indeed optimal. So far, the approach considered was to provide a planetary sequence and then optimise the trajectory with interior point constraints. The mission analysts investigate many possible different scenarios (planet sequence) according to his own experience and general knowledge. This approach is however time consuming and does not guarantee the optimality of the mission scenario. The purpose of the present study is thus to provide methods to automatically find the optimal scenario.

During my research, this subject has often raised questions from experts. This is understandable since it can be seen as finding a proof of global optimality of a general non-linear and non-convex function. However, this issue is of strong interest in the space community, as can be shown by recent Global Trajectory Optimisation Competitions, initiated by the ESA/ESTEC Advanced Concept Team.

Such solution methods would reduce mission operation duration, risks and costs. It would also be possible to increase the scientific payload of interplanetary spacecrafts and the scientific returns.

Proposed Approaches and Contributions

Two approaches have been considered:

1. The initial and first idea was to consider this problem as a global optimisation problem. In a first step, the best scenario can be found by iterating on most of the possible planet sequences. Because the general low-thrust problem is already difficult to solve when the planet sequence is known, we introduce a low thrust model which provides an approximation of the trajectory. The trajectory problem can then easily be solved. The initial infinite dimensional optimisation problem can now be turned into a parametric finite dimensional optimisation problem. To reduce complexity, specific constraints and assumptions are made. To reduce computational time, the planet sequence length is limited and only the most probable planets are considered. The most probable planets can be found using energetic approaches.

   Dynamic Programming and Global Optimisation algorithms are used for the search. We also consider the Mixed Integer Programming (MIP) approach to find the optimal sequence of planets.

   We reduce as much as possible the engineering expertise needed to find the optimal scenario. However, it would clearly be a mistake to consider those solution methods as "push button" methods.

2. Another approach which was considered is to actually not focus on global optimisation but rather to local optimisation. Indeed, during the research we clearly highlight the fact that most of the current techniques are unable to solve accurately the general multi-gravity interplanetary transfer problem. In other words, most of the optimisation techniques are unable to solve an interplanetary transfer considering swing-by that would not be forced with intermediate constraints and patched conic approximation.

   We consider a general formulation, where the gravity field of most promising planets is taken into account. No interior point constraints are considered. If a swing-by is to be used, it is only because of the dynamics. A challenging task is then to design a solver that would automatically get the locally optimal swing-bys.

   In a first step, we set-up a benchmark problem which should help design the algorithm. The new problem results in solving interplanetary direct transfer considering the initial planet escape and final planet capture phases. This leads to the design a gradient based algorithm. In a second step we consider the general interplanetary transfer problem, considering intermediate planets.

> Planeto-centric spiralling phases last a long time and require precise integration. The capture-escape problem is thus more difficult than the general interplanetary transfer problem. However, the later needs appropriate initialisation to automatically find locally optimal swing-bys.
>
> To produce as much swing-by trajectory as possible, a gravity continuation extension is investigated. This extension proposes to modify the planets gravitational constant to be placed in particular locally convex sub-spaces.

These two approaches have some strong differences, but are complementary. The first approach can be useful to initialise the second one. The major difference is on the property of the optimum found. The first approach is likely to provide a fair global "optimum", whereas the second one would provide an accurate local optimum.

Among main results, those which are original can be listed as:

1. a pruning procedure is proposed to tackle the high computational complexity of multi-gravity assist trajectories, through parametrised trajectory models[Oly07a].
2. a new low-thrust[Oly07b] model that extends previous work in the field. The low-thrust model is very close to the optimal solution, and allows to rapidly find interesting opportunity windows for low-thrust interplanetary missions. It is also very practical, i.e. fast and easy to implement.
3. development of a gradient method, extended specifically to interplanetary transfer problems in multi-body dynamics[Oly08]. Control constraints are handled through a function transformation approach. Convergence is improved with the use of an augmented Lagrangian formulation. An adaptive control mesh procedure is proposed for the continuous formulation. Focus has also been aimed on understanding advantages and flaws of direct/indirect methods.
4. application of gradient methods to find a locally optimal planet sequence, and swing-by.

Thesis Outline

This thesis is split into three parts.

Chapter 1 describes the most important concepts and technology to understand the subject properly. We introduce high and low-thrust propulsion, with brief historical reviews. We introduce the dynamics and physics of swing-by, major concern of the study.

Chapter 2, Chapter 3 and Chapter 4 are the first parts of the study. They consider the problem as a parameter global optimisation problem. As an introduction to space trajectory problems, Chapter 3 deals with the problem of impulsive or chemical thrust trajectories. We describe common approaches to tackle the problem. We introduce an approach that allows reducing the complexity of the multi-gravity assist trajectory problem, when we can parametrise the control. This is the case in Chapter 2, however, in Chapter 3 we introduce a low-thrust parametrised model.

The model presented in Chapter 3 allows using the exact same methods as those presented for the impulsive trajectory design problem of Chapter 2. A pruning policy is introduced to identify potentially good opportunities for a given mission.

Chapter 4 presents some examples of the approach.

Chapter 5, Chapter 6 and Chapter 7 are the second part of the study. They deal with the optimal control problem. Chapter 5 is a brief review of major optimal control methods. The optimal control problem is formulated. The dynamics include all major body dynamics. As a result, the patched conic approximation is never used. We discuss its use for our problem, and try to highlight the difficulties. This allows setting the requirements for a robust optimiser.

Chapter 6 introduces the robust optimiser, which is capable of handling swing-bys.

Chapter 7 presents examples of the trajectory problem, solved with our optimiser. Examples with changes of dynamics are considered. Examples of automatic swing-by optimisation are also treated.

Contents

List of Figures

List of Tables

Mathematical Notations

Vectors and Matrices are written in **bold** face.

$\mathbb{R}$ is the set of real numbers.
$\mathbb{N}$ is the set of natural numbers.
$\mathcal{M}_{mn}(\mathbb{R})$ is the set of matrices of m rows and n column. And, for $\mathbf{M} \in \mathcal{M}_{mn}(\mathbb{R})$, $M_{ij} \in \mathbb{R}, \quad \forall i \in [\![1, m]\!], \quad \forall j \in [\![1, n]\!]$.
$C^p(E, \mathbb{R})$ is the set of application from E to $\mathbb{R}$. C^∞ is the set of smooth functions.

Vectors are written column-wise:

$$\mathbf{v} = \begin{bmatrix} v_1 & v_2 & \dots & v_n \end{bmatrix}^T$$

The transpose of a vector $\mathbf{x}$ is denoted $\mathbf{x}^T$ and it is row-wise.

The transpose of a $n \times m$ matrix $\mathbf{M}$ is noted $\mathbf{M}^T$, and $\mathbf{M^T_{ij}} = \mathbf{M_{ji}}$ for all $i \in [\![1, n]\!]$ and all $j \in [\![1, m]\!]$.

Unless otherwise stated, $\|.\|$ denotes the Euclidean vector norm (two-norm).

We use δ for small quantities, and Δ for differences.
The notation *df* is used for the Frechet differential of an application f.

The gradient of a scalar function $f(\mathbf{v})$, i.e. the derivative of f w.r.t. the vector $\mathbf{v}$, is denoted:

$$\nabla_{\mathbf{v}} f = \frac{\partial f}{\partial \mathbf{v}} = \begin{bmatrix} f_{v_1} & f_{v_2} & \dots & f_{v_n} \end{bmatrix}^T$$

The Jacobian matrix of an application $\mathbf{f}(\mathbf{v})$, i.e. the derivative of $\mathbf{f}$ w.r.t. the vector $\mathbf{v}$, is the $\mathcal{M}_{mn}(\mathbb{R})$ matrix denoted:

$$\begin{aligned}\nabla_{\mathbf{v}}\mathbf{f} &= \begin{bmatrix}\nabla^T_{\mathbf{v}} f_1 & \nabla^T_{\mathbf{v}} f_2 & \dots & \nabla^T_{\mathbf{v}} f_n\end{bmatrix}^T \\ &= \begin{bmatrix}\frac{\partial f_1}{\partial v_1} & \frac{\partial f_1}{\partial v_2} & \cdots & \frac{\partial f_1}{\partial v_n} \\ \frac{\partial f_2}{\partial v_1} & \frac{\partial f_2}{\partial v_2} & \cdots & \frac{\partial f_2}{\partial v_n} \\ \vdots & \dots & \ddots & \vdots \\ \frac{\partial f_m}{\partial v_1} & \frac{\partial f_m}{\partial v_2} & \cdots & \frac{\partial f_m}{\partial v_n}\end{bmatrix}\end{aligned}$$

We will often use the compact notation: $\mathbf{f_v} = \nabla_{\mathbf{v}}\mathbf{f}$.

The Hessian of a scalar function $f(\mathbf{v})$, i.e. the second derivative matrix of f w.r.t. the vector $\mathbf{v}$, is the $\mathcal{M}_n(\mathbb{R})$ symmetric matrix denoted:

$$\nabla^2{}_{\mathbf{v}} f = \begin{bmatrix}\frac{\partial^2 f}{\partial v_1^2} & \frac{\partial^2 f}{\partial v_1 v_2} & \cdots & \frac{\partial^2 f}{\partial v_1 v_n} \\ \frac{\partial^2 f}{\partial v_2 v_1} & \frac{\partial^2 f}{\partial v_2^2} & \cdots & \frac{\partial^2 f}{\partial v_n v_2} \\ \vdots & \dots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial v_1 v_n} & \frac{\partial^2 f}{\partial v_2 v_n} & \cdots & \frac{\partial^2 f}{\partial v_n^2}\end{bmatrix}$$

We will often use the compact notations:

$$\mathbf{f_v} = \nabla_{\mathbf{v}}\mathbf{f} \tag{1}$$

$$\mathbf{f_{vv}} = \nabla^2{}_{\mathbf{v}}\mathbf{f} \tag{2}$$

Chapter 1

Introduction to Space Trajectory Problems

1.1 Space Propulsion Systems

1.1.1 General considerations

Any body in space is under the influence of a general central force field. Because of the large distance between planets and the Sun, the spacecraft is up to first approximation under the influence of a single massive body. Second order influences include perturbations such as:

- the gravity potential of the planets
- the non homogeneous material of the body (J_2)
- the solar radiation pressure
- the luni-solar potential

A spacecraft in space is thus primarily under the influence of a major body, called *primary*. It is said to be in a *heliocentric region* if the Sun is the primary body, or a *planetocentric* region if a planet is the primary body. These regions are often called gravispheres for planets and can be related to the Sphere of Influence or the region of Hill, depending on the dynamics considered.

Under no thrust conditions, its movement around this major body is (neglecting second order perturbations) a Keplerian arc referred to as a conic (circles, ellipses, hyperbolas and parabolas*. This results from the integration of the fol-

*These are conics of different energy. Closed shapes have a negative energy ($E \leq 0$), while parabolas and hyperbolas have a positive energy ($E > 0$)

lowing equation:

$$\frac{d\mathbf{V}}{dt} = -\mu \frac{\mathbf{R}}{\|\mathbf{R}\|^3} \tag{1.1}$$

where $\mathbf{R}$ and $\mathbf{V}$ are the spacecraft position and velocity wrt the central body, and μ is the central body gravitational parameter.

In this study, we shall only consider the gravity potentials of the planets. In practice, though, the second order perturbations have an influence over long time periods.

No energy is required to move along this arc, however to place the spacecraft at the desired location and at a desired date, we need to create a force that changes the current orbit energy.

In rocket propulsion, acceleration is caused by reaction (second law of Kepler). We create a thrust with mass m and celerity c.

There are two ways of doing this:

- aerobic: the mass comes from the outside (air) and is accelerated. ex: Turbopropulsor, Turboreactor... mainly aircraft engines
- anaerobic: the mass belongs to the system and is expelled. ex: Rocket.

Because of the vacuum, only anaerobic propulsion is used on spacecrafts.

The thrust of a rocket is defined by the resultant of the forces of pressure on the overall system. With c the exhaust velocity, P_s the exit pressure, and P_a the ambient pressure (free stream pressure), applying the Euler theorem to an elementary mass leads to:

$$F = c\dot{m} + A(P_s - P_a) \tag{1.2}$$

In a vacuum, we usually simplify this expression by writing $F = c\dot{m}$, where c includes the exhaust velocity plus the pressure term.

From this equation, there are basically two ways of producing high thrust. We can either have a very high flow rate $\dot{m}$, or have a very important exit velocity c.

We define the specific impulse, I_{sp}, by the ratio:

$$I_{sp} = \frac{c}{g_0} \tag{1.3}$$

The specific impulse is, for historic reasons, defined in seconds [s]. It represents the amount of thrust we can generate with the weight of propellant expelled during 1

s. The specific impulse is often used to compare thrust systems, under the same working conditions.

In mission analysis, and for impulsive manoeuvres, we often work with ΔV which is the quantity that represents the velocity increment needed to perform a manoeuvre. With:

$$F = m\frac{dV}{dt} = c\dot{m}$$

The velocity increment ΔV is given by Tsiolkovsky formula. The ΔV increase due to the thrust, from t_0 to t_f, is:

$$\begin{aligned} \Delta V &= I_{sp} g_0 \ln \frac{m(t_0)}{m(t_f)} \\ &= I_{sp} g_0 \ln(1 + pmf) \end{aligned} \tag{1.4}$$

with the introduction of the propellant mass fraction:

$$pmf = \frac{\Delta m}{m(t_f)} \tag{1.5}$$

ΔV accounts for the change in energy required to perform a manoeuvre, or a change of orbit. It is usual to use the ΔV as an indicator of the performance of the spacecraft [SKL+02].

1.1.2 Chemical Propulsion

Brief History

The very first solid rocket probably dates back to the first centuries AD in China, with a special form of gunpowder. Many experiments have been carried out since then, but only with solid fuel.

A Russian teacher, Tsiolkovsky (1857 - 1935), studied many concepts of rockets. Both Robert Goddard (1882 - 1945) and Tsiolkovsky drew the conclusion that liquid propellants have more power than solid propellants. In 1926, Robert Goddard successfully launched the first rocket using liquid chemical propulsion.

In Germany, in 1934, Werhner von Braun (1912 - 1977) successfully launched an ethanol - liquid oxygen A2 rocket, stabilised with a gyroscope. In 1942, he launched the first successful A4, later known as V-2. This same rocket was later improved and used for the US space program. With the Cold War and rapid rocket development, chemical propulsion and rockets were first used by the Soviet Union, for space applications, in October 1957, with Sputnik 1. Sputnik was launched with a two-stage R7 rocket, using liquid Oxygen and Kerosene.

Walter Hohmann (1880 - 1941) provided the first calculation of interplanetary orbit transfers. Tsiolkovsky demonstrated the necessity of multi-stage rockets for reaching space orbits. It is important to note that the first ideas of launching a satellite into orbit are mainly due to famous scientific authors such as Jules Verne (1828 - 1905), Edward Hale (1822 - 1902), or Arthur C. Clarke (1917 - 2008).

Physics

Chemical propulsion is limited by the chemical energy of the propellant. The chemical energy is equal to the difference in heat of formation between the reactants and the products of the reaction, usually noted ΔH. Indeed, the best bi-propellant combination is LOX/LH2. This combination is theoretically limited to about $\Delta H = 3.7kWh/kg$. This gives a temperature of combustion, from which we deduce the characteristic velocity $c = 2435m/s$. The practical upper Isp for LOX/LH2 is then 450s in the vacuum.

In fact, all the chemical energy ΔH is not practically available due to dissociation products at high temperature. Also, the final Isp depends on the geometrical configuration of the thruster. A slight improvement on the Isp can generally increase the payload mass by few very useful kilogrammes.

Model

With chemical propulsion, with high thrust, the duration of burn arcs is low compared to mission duration. Manoeuvres can be considered as impulsive and isolated (no other impulse in the immediate neighbourhood).

$$\int_t^{t+\delta t} \frac{\mathbf{F}}{m} dt \rightarrow \Delta \mathbf{V}(t) \tag{1.6}$$

Thus, applying a thrust manoeuvre at time t_m results in the following change of velocity:

$$\Delta \mathbf{V}(t_m) = \mathbf{v}(t_m^+) - \mathbf{v}(t_m^-) \tag{1.7}$$

This is the difference between the velocity before t_m^- and after t_m^+, the impulse. A method to evaluate $\Delta \mathbf{V}(t_m)$ is provided in the next section 2.2.2. The amount of fuel required to perform the manoeuvre is calculated using equation 1.4.

Types of Chemical Propulsion

Chemical propulsion encompasses solid and liquid propulsion.

- Solid : With an Isp varying between 250-300s, solid propulsion is essentially used for launcher boosters. They are simple to integrate but due to noticeable dispersion in impulse, they are not use for precise manoeuvre. They also cannot be restarted, limiting their use to the first stage of launchers.

- Liquid:
  - Monopropellant engines are the best known propulsion system, with high reliability and moderate cost. Their Isp ranges from 350s to 450s.
  - Bipropellant engines can be restarted, but are generally more complex and costly than monopropellant engines.

1.1.3 Electrical Propulsion Systems

Brief History

As early as 1906, Robert Goddard (1882 - 1945) was the first to demonstrate the attractiveness of Electrical Propulsion[JC02]. He noted that charged particles could be accelerated to high speed, which is not possible with chemical propulsion due to heating wall issues. He concluded that electrically accelerated particle systems could provide high exhaust velocity propulsion systems. He also envisioned the utility of electrical propulsion for interplanetary transfers!

Later followed the work of Tolstoiski (1911), Oberth (*Possibilities of Space Flight* in 1929, *Man Into Space* in 1957) and Stuhlinger (*Possibilities of Electrical Space Ship Propulsion* at IAC 1955).

The U.S. ambition in space rapidly helped research in Electrical Propulsion thrive. Those systems have existed now for more than 40 years. Electrical Propulsion was first tested in the early 1970s. In the 1990s, the US used electrical propulsion on communication satellites (e.g. the Iridium satellite constellation), but only as a station keeping and attitude control option. They have only been used as a primary propulsion system since the late 1990's, mainly because of a lack of confidence in those systems.

However, those systems are far more efficient than conventional chemical propulsion systems. They allow bigger ΔV, allowing more complex missions.

The first use of such systems, as a primary propulsion system, was on NASA's Deep Space 1 mission in 1998 (figure 1.1). Europe tested its first electrical propulsion system in space with spacecraft Artemis in 2001. But Smart-1, on its way to the Moon, was the first successful European probe to use electrical propulsion in 2003. Later on that year, Japan's Hayabusa spacecraft used electrical propulsion for an asteroid sample return mission.

The BepiColombo mission to planet Mercury (launch scheduled on 2013) will use both chemical and electrical propulsion systems (SEP). Based on their experience with SMART-1, ESA will combine low-thrust propulsion and gravity assist to approach planet Mercury. A special capture strategy using chemical propulsion will be used to place the spacecraft into orbit.

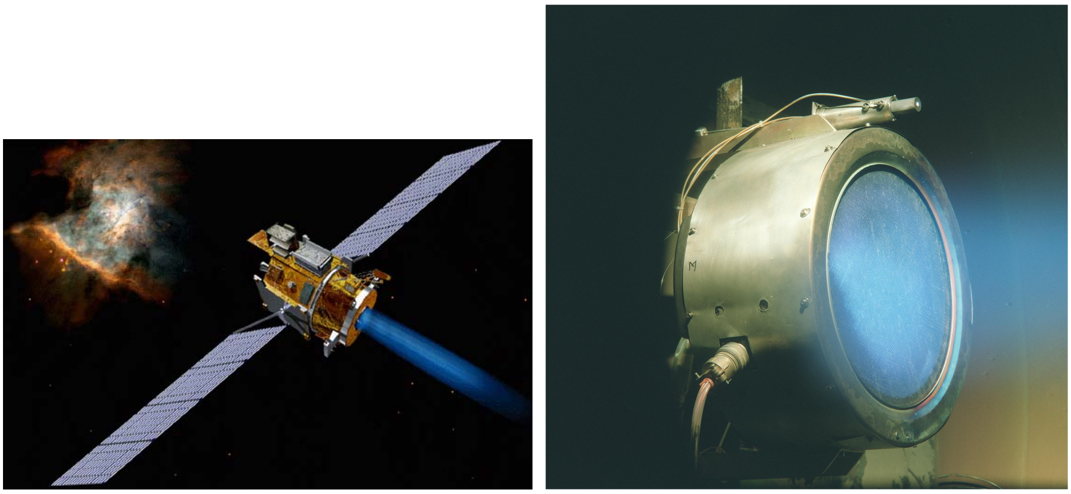

Figure 1.1: Deep Space One (DS1) Ion Propulsion (©NASA/JPL)

The current DAWN mission[RFRR04, RFRR05], launched in 2007, is using electrical propulsion to reach comet Ceres and asteroid Vesta. Its accumulated thrust time is about 6 years, for a total mission time of about 8 years.

Physics

Ion propulsion systems basically accelerate ions. Such a propulsion has no energy limit. Energy can come from a nuclear cell or can be from solar origin. Thus, for solar power units, the limitation comes from the sun illumination.

The propellant velocity is a few orders of magnitude greater than for chemical propulsion. The total impulses are of the same order, although electrical propulsion works over a long duration. This is called *continuous thrust*. When the ratio of thrust over gravity is small, we say we have *low-thrust propulsion*.

Thus, as opposed to chemical propulsion, thrust manoeuvre cannot be approximated by an instantaneous change in velocity.

Electrothermal

Electrothermal propulsion has similarities with chemical liquid propulsion. A propellant gas is warmed, and accelerated by expansion through a divergent nozzle. Nuclear thermal propulsion is limited by maximum wall temperature. In practice, the maximum Isp is around 1000s.

Systems: Arcjets, Resistojets, Microwave plasma.

Electrostatic (ion propulsion)

Charged particles are accelerated by electrostatic forces.

With an Electrostatic force field $\mathbf{E}$:

$$\mathbf{F} = q\mathbf{E} \tag{1.8}$$

The efficiency of ion propulsion comes from the association between very light particles and great electrical charges, to create important particle acceleration.

Systems: Gridded electrostatic ion, Field Emission (FEEP), (Contact Ion).
Propellant for ion propulsion can be: Carbon 60, Cesium, Krypton, or Xenon.

Electromagnetic (plasma)

A stream of conduction material is electromagnetically accelerated.

With the Lorentz force, a charged particle q is accelerated with the electrical force field $\mathbf{E}$ and deviated by the magnetic field $\mathbf{B}$:

$$\mathbf{F} = q(\mathbf{E} + \mathbf{v} \times \mathbf{B}) \tag{1.9}$$

Systems: Pulsed Inductive, Hall effect, (Magneto-plasma-dynamics).
Propellant for ion propulsion can be: Argon, Hydrogen, Nitrogen.

The Variable Specific Impulse Magnetoplasma Rocket (VASIMIR (^{TM}) [EFP04, BBS$^+$06]) bridges the gap between chemical propulsion and low-thrust propulsion systems.

1.1.4 Comparisons

Figure 1.2 and Table 1.1 compare the range of thrust for different space propulsion systems.

Basically, with electrical systems, we have more thrust for the same mass, compared to chemical propulsion. However, with electrical propulsion systems, the engine must operate for a longer time.

For these reasons, Electrical and Chemical Propulsion have their specific, and sometimes common, uses. Ion propulsion cannot be used when great acceleration is required.

Typically, missions requiring high thrust (or chemical propulsion) can be:

- Takeoff and landing
- apogee and perigee correction
- manned missions (so far?)

Missions requiring high Isp (or electrical propulsion systems) can be:

- Deep Space mission

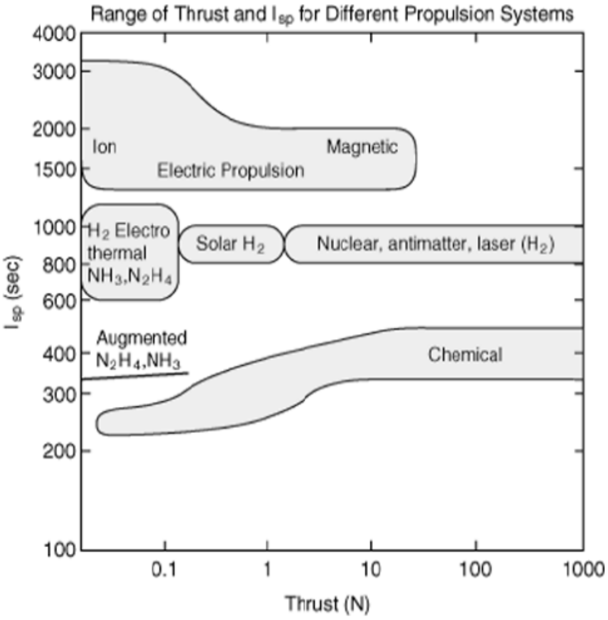

Figure 1.2: Space Propulsion ranges (source:http://www-ssc.igpp.ucla.edu)

System	Isp [s]	Thrust [N]	Application
	Chemical Liquid		
Monopropellant	150 - 235	0.1 to 100	Upper stage
Bipropellant	320 - 460	to 10^7	First stage, S/C
	Chemical Solid		
	260 - 360	10^3 to 10^7	Rocket booster
	Electric		
Electrothermal	500 - 1000	10^{-2} to 10	Low-Thrust S/C
Electromagnetic	1000 - 7000	10^{-3} to 10	Low-Thrust S/C
Electrostatic	2000 - 10000	10^{-6} to 10^{-3}	Low-Thrust S/C
Nuclear	800 - 1100	up to 1.2 10^7	
VASIMIR	1000 - 30000	up to 1200	no current use

Table 1.1: Typical thrust systems

- long term correction

- compensation (e.g. drag effect for ESA mission GOCE, 2008)

1.2 Multi body dynamics

1.2.1 General Dynamical Equations

The spacecraft is supposed to be only under the influence of the Sun and the gravity force fields of the planets. We define the reference frame by the centre of the Solar System, with axes oriented in a fixed direction. The reference frame is thus inertial. Using $\mathbf{u}$ as position vector, the spacecraft dynamical equation is :

$$\frac{d^2\mathbf{u}_{sc}}{dt^2} = -\mu_{SUN}\frac{\mathbf{u}_{sc} - \mathbf{u}_{SUN}}{\|\mathbf{u_{sc}} - \mathbf{u_{SUN}}\|^3} + \sum_{i=1,9} \mu_i \left(\frac{\mathbf{u}_{sc} - \mathbf{u}_i}{\|\mathbf{u_{sc}} - \mathbf{u_i}\|^3}\right) \tag{1.10}$$

Where $\mu_* = Gm_*$, G universal gravitational constant, and m_* is the attractive body mass.

Now consider the Sun dynamics in the same reference frame. The position of the spacecraft with respect to the Sun position is:

$$\mathbf{r}_{sc} = \mathbf{u}_{sc} - \mathbf{u}_{SUN} \tag{1.11}$$

The dynamics of the spacecraft with respect to the Sun, in the reference frame, are then:

$$\frac{d^2\mathbf{r}_{sc}}{dt^2} = -(\mu_{SUN} + \mu_{sc})\frac{\mathbf{r}_{sc}}{\|\mathbf{r_{sc}}\|^3} - \sum_{i=1,9} \mu_i \left(\frac{\mathbf{r}_{sc} - \mathbf{r}_i}{\|\mathbf{r_{sc}} - \mathbf{r_i}\|^3} + \frac{\mathbf{r}_i}{\|\mathbf{r_i}\|^3}\right)$$

As the spacecraft mass m_{sc} is negligible compared to the Sun mass m_{SUN}, we can simplify to:

$$\frac{d^2\mathbf{r}_{sc}}{dt^2} = -\mu_{SUN}\frac{\mathbf{r}}{\|\mathbf{r}\|^3} - \sum_{i=1,9} \mu_i \left(\frac{\mathbf{r} - \mathbf{r}_i}{\|\mathbf{r} - \mathbf{r_i}\|^3} + \frac{\mathbf{r}_i}{\|\mathbf{r_i}\|^3}\right) \tag{1.12}$$

The sum is negligible away from planets, but becomes important when close to a planet. Typically, as we will define it later, in the Sphere of Influence of a planet, the Sun influence can be discarded.

1.2.2 Sphere of Influence

Consider an object SC of mass m under the influence of 2 primaries, S and P of respective mass m_1 and m_2, with $m_2 < m_1$.The action of P acting on SC is:

$$F_{P\to SC} = -G\frac{mm_2}{d^2_{P\to SC}} \tag{1.13}$$

We denote $d_{A\to B}$ the distance between A and B. The constant G is the universal gravitational constant.

Similarly, the action of S on SC is:

$$F_{S\to SC} = -G\frac{mm_1}{d^2_{S\to SC}} \tag{1.14}$$

Comparing $F_{P \to SC}$ and $F_{S \to SC}$ provides a way to establish when one force overcomes the other.

The Sphere of Influence(SOI) defines the spherical limit where the planet gravitational influence overcomes other gravitational forces (disturbing acceleration) and in particular the Sun influence. In other words, according to this approximation, a spacecraft on orbit around a planet is not influenced by the Sun.

This notion is however just a definition and an approximation. It does not underline any physical phenomenon. Indeed, the gravity potentials are continuous functions in space.

The Sphere of Influence radius is thus given by[Bat01]:

$$R_{SOI} = d_{P \to SC} \left(\frac{m_2}{m_1} \right)^{0.4} \tag{1.15}$$

On the sphere of influence frontier, we admit we are at an infinite distance of the planet.

1.2.3 Patched Conic Approximation

The Patched Conic approximation is often used for interplanetary missions [MS65, PK94].

We often consider massless planets [MS63] [Joh69] [Ber01]. The trajectory is split into 2-body dynamical legs. The spacecraft is never under the influence of two bodies or more. The Sphere of Influence is used to place the limit between each leg. The conics are patched together according to position and velocity constraints.

This approximation often provides satisfactory results.

1.3 Gravitational Assist (Gravity Assist, Swing-by)

1.3.1 Description

When a point mass (e.g. a spacecraft) passes close to a planet with hyperbolic speed, it is not captured by the planet's gravity onto a negative energy orbit. But the planet gravity perturbs the mass object trajectory. Because of the conservative law of motion, the total linear momentum is conserved, i.e. the linear momentum before and after the encounter is the same. Thus, there has been an exchange of linear momentum between the spacecraft and the planet. Indeed, the spacecraft has changed speed, as well as the planet, however the relative velocity of the spacecraft, at infinity, with respect to the planet has not changed in magnitude. The manoeuvre has only re-oriented the relative velocity vector. Swing-bys are interesting to modify, sometimes significantly, the trajectory without expending more fuel. Because of their respective mass, the velocity change of the planet is negligible.

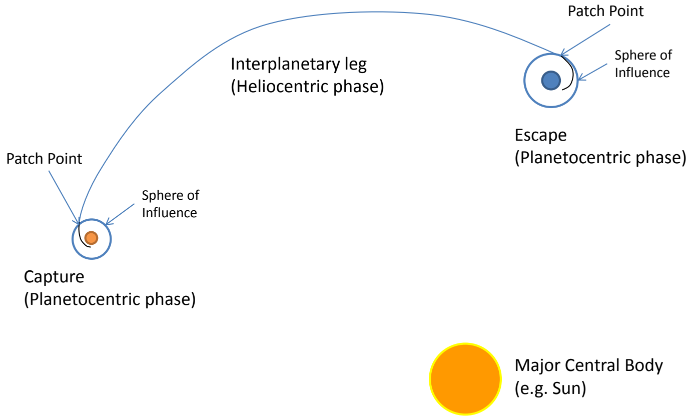

Figure 1.3: Patched Conic scheme

Mathematically[VA03], we can write:

$$m_{pl}\Delta E_{pl} + m_{sc}\Delta E_{sc} = 0 \tag{1.16}$$

The quantity m and E stand respectively for mass and orbit energy, whereas the subscripts sc and pl stand respectively for the spacecraft and the planet variables.

This phenomenon if called *gravitational assist* or *swing-by*. Sometimes it is also referred to as fly-by. A fly-by is usually more related to the close approach to an object rather than the physical phenomenon itself.

Planet Mercury can provide the biggest energy exchange, whereas Jupiter can provide the biggest angular deviation.

The first idea of using planets to gain or lose energy came way back in 1918 to Yuri Kondratyuk (1897 - 1942). The first concepts came a short time later with Lawden and Firshoff in 1954.

The first mission to use a gravity assist was Mariner 10, launched in 1973. The Mariner 10 spacecraft made a swing-by of Venus (Feb. 1974) before multiple swing-byes of Mercury.

Voyager 1 and Voyager 2 , both launched in 1977 (Voyager 2 was launched first) are probably the most impressive missions using gravity assists (see figure 1.3.1). The initial 4 year mission spacecrafts, Voyager 1 and Voyager 2, both used a swing-by of Jupiter to reach Saturn. Voyager 2 continued its route to Uranus, Neptune, using perfectly synchronised swing-byes. It is likely that at this current

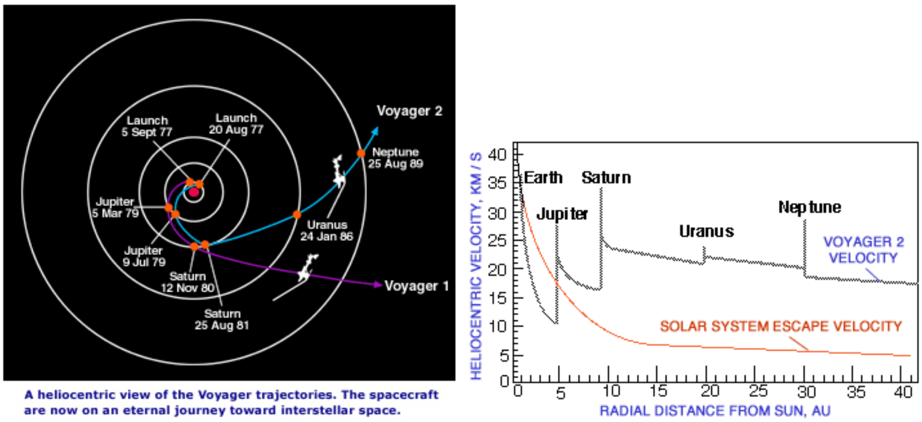

Figure 1.4: [Left]Voyager 1 and Voyager 2 trajectories. [Right] Voyager 2 velocity profile.

date, such a journey would not be possible without thrusting to have the perfect swing-by timing conditions. Voyager 1 continues its route after a Saturn swing-by to reach interstellar space. In September 2008, Voyager 1 should be at about 107.6 AU from the Sun. The initial 720 kg spacecraft has only used gravity assists to reach that point, and no thrust (Voyagers' thrusters have a thrust of less than 1 N!). It should wander into interstellar space for long, although the community is divided on whether it has passed the Termination Shock yet.

Swing-bys modify the heliocentric velocity (or energy) of the spacecraft. The inclination, and all other orbital parameters, are also modified. As an example, the Ulysses mission was able to be positioned over the poles of the Sun with a Jupiter swing-by.

One of the most complex mission to date is Cassini. Launched in 1997, the spacecraft used swing-bys of Venus, Earth and Jupiter to reach Saturn. Once on orbit around Saturn, the spacecraft will performs till 30 swing-byes of the Saturn moon Titan to explore the Saturnian system.

More recently, in the New Horizon program, the mission to Pluto was significantly reduced in time and consumption thanks to a Jupiter swing-by. The use of the swing-by permits to reduce the flight time by reorienting the velocity vector at Jupiter to reach Pluto more rapidly. The spacecraft should reach Pluto at 32 AU from the Earth in "only" 9.5 years.

1.3.2 Simplified Model

Since the spacecraft making the swing-by must not be captured, its movement in the sphere of influence is a hyperbola. The point of closest approach is thus called *periapsis*.

A swing-by can be described by the incoming and outgoing hyperbolic velocity

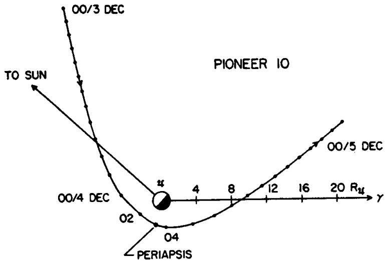

Figure 1.5: Ecliptic projection of the Pioneer 10 swing-by with planet Jupiter, on December 1973. Ticks represent 2 hours time intervals.

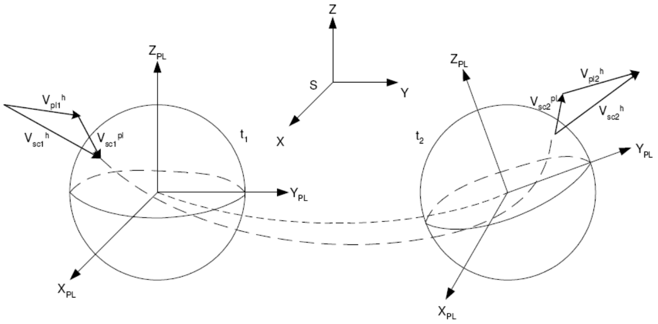

Figure 1.6: Geometrical configuration (source: [Tad04])

vectors, respectively noted V_{sc1}^{pl} and V_{sc2}^{pl}, or also V_{∞}^{-} and V_{∞}^{+} (figure 1.6), and:

$$\mathbf{V}_{sc1}^{pl} = \mathbf{V}_{sc}(t_1) - \mathbf{V}_{pl}(t_1)$$
$$\mathbf{V}_{sc2}^{pl} = \mathbf{V}_{sc}(t_2) - \mathbf{V}_{pl}(t_2)$$

These vectors are evaluated at the boundary of the Sphere of Influence, thus the velocity at an "infinite" distance from the planet. They define the hyperbolic trajectory in this region. Because of angular momentum conservation, these two vectors lie in the same plane, the trajectory plane.

To describe the geometry of the swing-by, we should introduce a reference frame. The geometrical description is given by figure 1.7. The B-plane is perpendicular to the incoming hyperbolic velocity vector $\mathbf{V}_{\infty}^{-}$, and goes through the central body gravity centre. Vector T is the vector leaving the central body, leading to the point of intersection between the vector $\mathbf{V}_{\infty}^{-}$ and the B-plane. We define vector **B** as the vector coming from the central body, along the line of intersection between the B-plane and the trajectory plane (the ecliptic, or any other plane of reference).

When no thrust manoeuvre is performed within the Sphere of Influence (SOI)[Bat01],

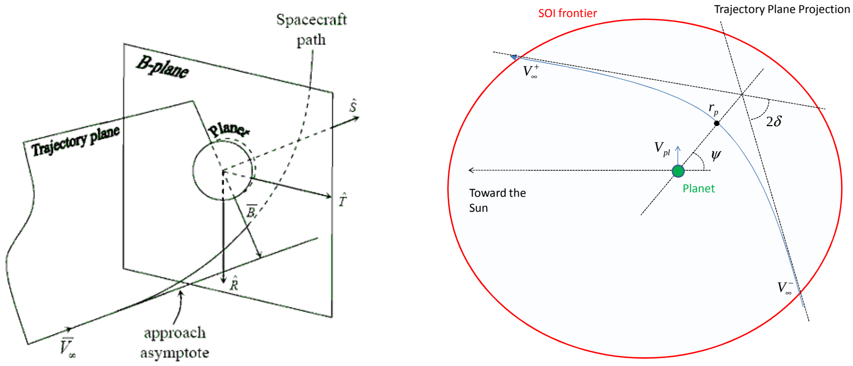

Figure 1.7: Swing-by geometrical description

the spacecraft energy is conserved. Thus, the incoming and outgoing hyperbolic velocities at the Sphere of Influence are the same:

$$\|\mathbf{V}_\infty^+\| = \|\mathbf{V}_\infty^-\| \tag{1.17}$$

The angular deviation due to the swing-by is given by:

$$\sin\delta = \frac{\mu}{\mu + rV_\infty^2} \tag{1.18}$$

end the angular rotation between the vectors $\mathbf{V}_\infty^+$ and $\mathbf{V}_\infty^-$ is:

$$\phi = 2\delta = 2\arctan\frac{\mu_p l}{r_p {V_\infty}^2} \tag{1.19}$$

The constant μ defines the swing-by planet gravity constant.

A swing-by is practically feasible only if the pericenter radius is beyond the planet radius. We thus deduce the maximum deviation:

$$\sin\delta_{max} = \frac{\mu}{\mu + r_{min}V_\infty^2} \tag{1.20}$$

The velocity increase module is given by:

$$\begin{aligned} \Delta V &= \|\mathbf{V_{sc}(t_2)} - \mathbf{V_{sc}(t_1)}\| && (1.21)\\ &= 2V_\infty \sin\delta && (1.22) \end{aligned}$$

In the patched-conic and massless swing-by planet approximation, a swing-by is equivalent to performing an impulsive manoeuvre. Table 1.3.2 shows the maximum ΔV for each planet.

The time spend in the SOI is given by the hyperbolic motion. This time can be computed using Kepler's equation and the SOI hyperbola parameters. We have then[BGRC04]:

Planet	max ΔV	R_{SOI}
Mercury	3.01 km/s	112400 km
Venus	7.33 km/s	616300 km
Earth	7.91 km/s	927000 km ($145R_{Earth}$)
Mars	3.55 km/s	577200 km
Jupiter	42.73 km/s	48.3 10^6 km ($657R_{Jupiter}$)
Saturn	25.62 km/s	54.74 10^6 km
Uranus	15.8 km/s	51.64 10^6 km
Neptune	16.78 km/s	86.59 10^6 km
Pluto	1.10 km/s	31.34 10^6 km

Table 1.2: Swing-By effects from planets. ΔV referred as the maximum hyperbolic velocity increase after a swing-by. R_{SOI} defines the radius of the Sphere of influence.

$$\Delta t = 2\sqrt{\frac{a^3}{\mu_pl}}\left(\csc\frac{\phi}{2}\sinh H - H\right) \tag{1.23}$$

$$\cosh H = \left(1 + \frac{r_{SOI}}{a}\right)\sin\frac{\phi}{2} \tag{1.24}$$

$$\frac{1}{a} = \frac{v_1^2}{\mu_pl} - \frac{2}{r_{SOI}} \tag{1.25}$$

Usually we suppose the time spent in the SOI negligible compared to the mission duration (see for example figure 1.5). In most cases, for interplanetary transfers, the swing-by can be considered as instantaneous.

With the angular momentum vector $\mathbf{H} = [h_x, h_y, h_z]^T$, we define the rotation matrix $\mathbf{R}(\mathbf{H}, \phi)$. The rotation matrix $\mathbf{R}(\mathbf{H}, \phi)$ performs a rotation of angle ϕ around $\mathbf{H}$:

$$\mathbf{R}(\mathbf{H}, \phi) = \begin{pmatrix} h_x^2 + \cos\phi(h_x^2 - 1) & h_xh_y(1-\cos\phi) - h_z\sin\phi & h_zh_x(1-\cos\phi) + h_y\sin\phi \\ h_xh_y(1-\cos\phi) + h_z\sin\phi & h_y^2 + \cos\phi(1 - h_y^2) & h_yh_z(1-\cos\phi) - h_x\sin\phi \\ h_zh_x(1-\cos\phi) - h_z\sin\phi & h_yh_z(1-\cos\phi) + h_x sin\phi & h_z^2 + \cos\phi(1 - h_z^2) \end{pmatrix} \tag{1.26}$$

Thus, using $\mathbf{H}$ as the spacecraft angular momentum during the swing-by, this is the rotation matrix bringing $\mathbf{V}_{sc}(t_1)$ to $\mathbf{V}_{sc}(t_2)$.

1.3.3 Tisserand Criterion

A swing-by is a useful tool to change a spacecraft orbit. However, it is not always possible to place the spacecraft onto the desired orbit. The conservation of kinetic and angular momentum only allows certain configurations. The changes of energy for the spacecraft and/or the planet are bounded.

Tisserand (1845 - 1896) demonstrated that for 2 point masses m (the spacecraft) and M (the planet), $m << M$, orbiting around the Sun, there is a parameter provided by the orbital elements of m and invariant with the dynamics of M. As explained with the swing-by, the point mass m is mostly under the influence of the Sun on an elliptical orbit, but close to M. Hence, before and after the swing-by, the point mass m is on a regular orbit (i.e. elliptical with constant semi-major axis, eccentricity and inclination). In general however the orbital parameters will change after the encounter.

Consider the Restricted Three Body Problem (RTBP)[KLMR01, KMLR02] $\{\text{Sun}, M, m\}$, with (x, y, z) and $(\dot{x}, \dot{y}, \dot{z})$, respectively the position and velocity of m with respect to the central body. The RTBP frame is rotating with respect to a fixed frame, at the constant angular velocity ω. We have the Jacobi equation [BGRC04, Bat01]:

$$C(x, y, z, \dot{x}, \dot{y}, \dot{z}) = 2\Omega(x, y, z) - \left(\dot{x}_r^2 + \dot{y}_r^2 + \dot{z}_r^2\right) \tag{1.27}$$

With:

$$\Omega(x, y, z) = \frac{\omega(x_r^2 + y_r^2)}{2} + \frac{\mu_1}{r_1} + \frac{\mu_2}{r_2} \tag{1.28}$$

Then note with $\mu_2 << \mu_1 = \mu$ and $\rho = r_1$:

$$\begin{aligned} \dot{x}_r^2 + \dot{y}_r^2 + \dot{z}_r^2 &= V^2 && (1.29)\\ &= \mu(-\frac{1}{a} + \frac{2}{\rho}) \\ \omega(x_r^2 + y_r^2) &= h\cos(i) && (1.30)\\ &= \sqrt{a(1-e^2)}\cos(i) \end{aligned}$$

Thus during a swing-by:

$$\frac{C}{\mu} = \frac{1}{a} + 2\sqrt{\frac{a(1-e^2)}{\rho^3}}\cos(i) \tag{1.31}$$

This is known as the *Tisserand Criterion*. It depends on the semi-major axis a, the eccentricity e and the inclination i with respect to the point mass M orbit plane.

Figure 1.8 shows the Tisserand Criterion evolution. During a swing-by, we can "jump" from one curve to another as long as the criterion is kept constant. The limits are given by the planet-to-reach perihelion and aphelion. These ensure that we get the good parameter to reach the desired planet after a swing-by.

In practice, one should conclude that the Tisserand Criterion is the same before and after the encounter. It can be used for swing-by design[MW02]. Using

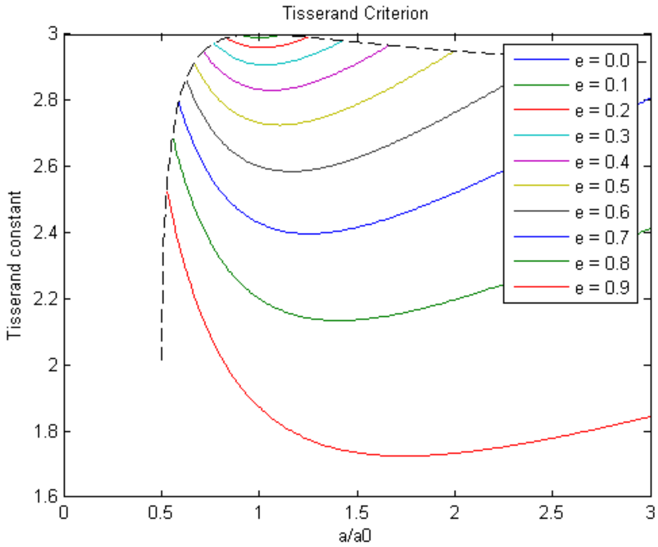

Figure 1.8: Tisserand Parameter evolution

figure 1.8, we can find what are the incoming orbit parameters to obtain the desired outgoing orbit parameters.

Often, astronomers use the Criterion Parameter to determine if the comet they are observing is not a previously discovered one, which has changed its orbital parameters because of a planet interaction.

1.4 Problem Statement

Recent advances in mission analysis and space trajectory optimisation provide efficient numerical techniques for continuous-thrust interplanetary trajectory optimisation with swing-bys.

Such methods are however limited to fixed mission scenario, where the planet sequence is a-priori known. The mission analysts are thus compelled to consider many different planet sequences, and select the one which presents the lowest cost, the best opportunity, and the best robustness. However, there is no evidence that the planet sequence considered is optimal.

The purpose of this thesis is to provide ways to find the "best" planet sequence. We have to solve **simultaneously** and **automatically** the transfer problems and the sequence determination problem.

Part I

Global Optimisation Method and Model

Presentation of Part I

This part deals with the global optimisation of spacecraft interplanetary trajectories.

In Chapter 2, we focus on impulsive trajectories. The impulsive trajectory problem is generally formulated as a parameter optimisation problem. From preceding work, we know that under mild assumptions, it is possible to find a neighbourhood of the global optimum with a tractable computational cost. In this part, we extend and develop past results for impulsive trajectories with Deep Space Manoeuvres.

Then, in Chapter 3, to deal with the low-thrust trajectory problem, we create a model for low-thrust trajectories which simplify the search of solutions. Not only this model allows to transform the infinite dimensional problem (typical in low-thrust propulsion), but it also allows to assess quickly mission opportunities. As the low-thrust model is parameterised, the approach initiated with the impulsive trajectory problem with Deep Space Manoeuvres can still be applied with the low-thrust model problem.

It is a the cost of a fast and effective solving process that we can evaluate different possible planet sequences.

Chapter 2

Automated Approach for Impulsive Interplanetary Trajectories

2.1 Energetics Approaches for promising Planet Sequences

When doing a systematic search of planet sequences, we can use mathematical and physical tools to preclude the calculation of uninteresting possibilities.

2.1.1 GAP Plots

It is possible to quickly assess the attainability of a planet with a given swing-by sequence. Gravity Assist Potential Plots ("GAP plots") introduced by [PLB00] are a simple way to evaluate a planet sequence. This evaluation provides the biggest semi-major axis reachable.

The GAP plots (figure 2.1) display the evolution of the semi-major axis with respect to the departure planet hyperbolic excess velocity V_∞.

To compute the semi-major axis of the final transfer orbit we make the following hypotheses:

- transfers are purely Keplerian
- the launch velocity is co-linear with the departure planet velocity
- phasing and ephemeris are not considered
- all planets lie in the same plane, and their orbits are circular
- for each swing-by the maximum deflection is considered

For the swing-by, the direction of V_∞ is outward if the following planet in the sequence is at a higher radius ($R_{currentplanet} < R_{nextplanet}$). The direction of V_∞ is inward if the following planet is inside.

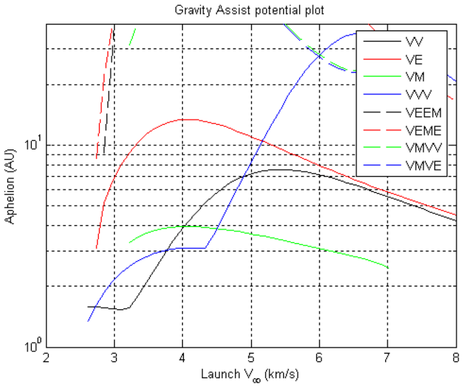

Figure 2.1: GAP plots

2.1.2 V_∞ plot

V_∞ - r_p plot

Strange and Longuski [SL02] introduced an efficient graphical (figure 2.2) method that quickly identifies all possible ballistic gravity-assist sequences. The method is based on the Tisserand criterion (section 1.3.3).

For each V_∞ and for each planet in the sequence, we "shoot" with variable directions $\alpha \in [0, 2\pi]$. For each direction, we evaluate the apoapsis (furthest point) and the periapsis (closest point), as well as the period of the "shooting orbit". We can plot the period against the periapsis. This indicates, for a ballistic transfer, what launch or swing-by energy is required to reach a certain distance from the Sun in space, and what is the average duration of the transfer.

Similarly, Miller and Weeks[MW02] use the Tisserand Criterion to identify the transfer orbits between planets. They consider impulsive transfers, and a given planet sequence. They seek the encounter times with the Tisserand criterion.

These energetic approaches mainly describe ballistic transfer, where the spacecraft does not thrust. We shall therefore study the problem with impulsive manoeuvres. This is a good transition before studying continuous thrust in chapter Chapter 3.

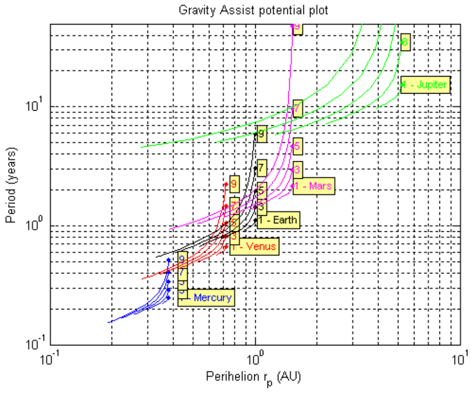

Figure 2.2: V_∞ plots

2.2 Chemical Trajectory Optimisation

2.2.1 Problem statement

With chemical propulsion, manoeuvres can be considered as impulsive and isolated (section 1.1.2). For multi-gravity assist trajectories, we use the patched conic approximation, with massless planets. The objective function, for the minimum mass problem, is to minimise the *characteristic velocity*:

$$J(\mathbf{X}) = \sum_i \|\mathbf{\Delta V_i}\| \tag{2.1}$$

where $\Delta\mathbf{V}_i$ are the impulsive manoeuvres that control the spacecraft trajectory (see figure 2.3), and $\mathbf{X}$ is a decision vector, which will be explained later.

The objective function J can also include the rendezvous manoeuvre cost, depending of the problem. The problem is thus to seek the optimal impulses description and the time of flight for each leg. The planet sequence (mission scenario) is part of the solution we seek. In this part, and as explained in the introduction, the planet sequence is always considered sequentially. Indeed, for each possible planet sequence, a new transfer problem is solved. We should be able to compute quickly and efficiently the solution transfers.

This problem is very non-linear and not strictly convex. It is a parameter optimisation problem (POP). There are many local minima. To find the best a-priori scenario we use global optimisation techniques, but the global optimum can be difficult to find. Usually seeking the global optimum can be reduced to seeking the best local optimum.

2.2.2 Impulsive Trajectories

For interplanetary transfers with swing-bys, the POP needs an efficient Lambert's problem [Bat01] solver.

Gibb's problem [VM97] Suppose given three non zero points, coplanar but not co-linear position vectors, representing the position of the satellite in orbit at three different dates. Gibb's problem is to determine the parameters of the orbit passing through these position vectors.

This orbit is defined by the three quantities:

- kinetic momentum vector
- eccentricity vector, pointed toward the periapsis (perigee, perihelion)
- semi-latus rectum, defined as the distance from a focus to the ellipse measured along a line perpendicular to the major axis.

Clearly, the number of conics linking 2 points is infinite. However, we can demonstrate that this number is finite if the transfer time is given. This is equivalent to adding a third point. We have Lambert's theorem[Bat01]:

Theorem 2.2.1 (Lambert's theorem)**.** *According to Gibb's theorem the time T required for the transfer from A to B depends only on the semi-major axis, the length of the chord between A and B, and the sum of A and B radii measured from the centre of attraction.*

This theorem is useful when seeking a transfer orbit. Lambert's problem is the problem of calculating a conic arc passing through 2 points A and B with a given time length T. If the conic arc is less than a revolution, Lambert's problem has a unique solution.

Solving Lambert's problem reduces to seeking the orbital elements of the desired transfer orbit. Providing the orbital elements permits evaluating the initial and final velocity, respectively at A and B. We thus have a way to calculate the required ΔV.

Solving this problem is not difficult, and it has been widely used in the literature[Pru79, Bat01].

Using the patched conic approximation, and the swing-by simplified model we can compute the manoeuvres to apply along the trajectory. The optimal control problem (OCP) can be formulated considering a parameter optimisation problem (POP). We thus work in a finite dimensional space.

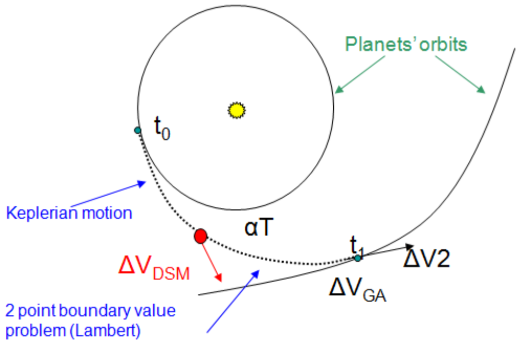

Figure 2.3: Trajectory with one intermediate impulse (1 DSM)

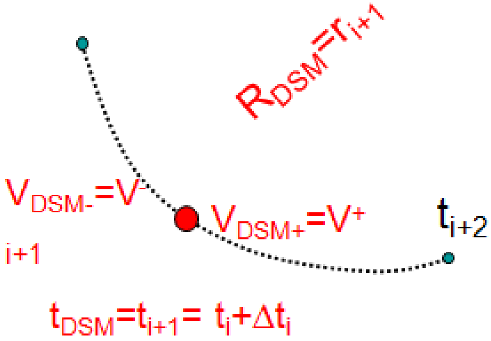

Figure 2.4: Intermediate Impulse Description (DSM)

2.2.3 Impulsive Trajectories with Deep Space Manoeuvres

If we now consider using intermediate manoeuvres along the trajectory (Deep Space Manoeuvres (DSM)), it becomes much more difficult to seek the global optimum. First, transfers from one planet to the next are not uniquely given. Second, the number of variables has widely increased. The complexity is no more polynomial.

As shown on figure 2.4, a deep space manoeuvre (DSM) is defined by at least 4 variables. We use here a time-position model, and the DSM is described by:

$$\mathbf{X}_{DSM} = [t_{DSM}, \mathbf{R}_{DSM}] \tag{2.2}$$

The date of the manoeuvre is given by t_{DSM}. The position of the manoeuvre in the position space is given by $\mathbf{R}_{DSM}$.

2.2.4 Multi-Gravity Assist Trajectory problems

Using pruning techniques and appropriate meshing of the solution space, we can easily compute the set of extrema. The Gravity Assist Space Pruning algorithm GASP [Mya03] evaluates quite efficiently the solution space to produce restricted search boxes in the solution space. These search boxes include interesting local optima. With an appropriate meshing the global optimum belongs to the set of local optima. Heuristic solvers such as Particle Swarm Optimisers (PSO) or Differential Evolution solvers (DE) can thus work in smaller sub-spaces and provide the global optimum with good certainty. This approach has a polynomial complexity in the number of phases.

2.2.5 Multi-Gravity Assist Trajectory problems with Deep Space Manoeuvres

The multi-gravity assist problem, with deep space manoeuvres (MGA DSM problem), is more complex than the MGA problem.

With the objective function:

$$J(\mathbf{X}) = \Delta V_0(\mathbf{X}) + \Delta V_f(\mathbf{X}) + \sum_{i=1}^{n} \Delta V_{DSM(n)}(\mathbf{X}) \tag{2.3}$$

we can describe the MGADSM problem with the decision vector:

$$\mathbf{X} = [t_0, V_0, \alpha_0, \beta_0, t_{DSM}^1, ..., t_i, \phi_i, r_p^i, t_{DSM}^i, ..., t_{n-1}, \phi_n, r_p^n, t_{DSM}^n, t_f]^T \tag{2.4}$$

The launch hyperbolic velocity $\mathbf{V}_0$ is given by $[V_0, \alpha_0, \beta_0]$. Variables ϕ_i and r_p^i define the swing-by conditions, and also the hyperbolic excess velocity vector $\mathbf{V}_{\infty,i}$ outgoing from the swing-by (equation 1.19).

The trajectory is integrated with the outgoing conditions $\mathbf{V}_{\infty,i}$ of the swing-byes, or the initial velocity $\mathbf{V}_0$, from t_i to $t_{DSM,i}$. This results in the position and velocity before the DSM is applied. The resulting DSM position permits to formulate and solve a Lambert's problem. By subtracting the change of velocity, between the resulting velocity and the Lambert's problem results (see figure 2.3), we can evaluate the manoeuvre velocity change $\mathbf{\Delta V}_{DSM,i}$.

We can the evaluate the objective function J (equation 2.3).

This formulation leads to a medium/large scale problem. It results in a difficult resolution. This problem has already been tackled in the literature, but with restrictions. Ceolin [Ceo98] uses the Melder-Mead simplex to optimise a DSM-trajectory. The number of impulses is fixed and rarely exceeds 1 DSM per leg.

The Primer Vector Theory [Law92] has been extended to multi-gravity assist trajectory[Oly07a]. The number of DSM per leg is thus free. This allows finding the best general impulsive trajectory without restrictions.

The complexity of this problem is important. This problem can also solved with global optimisation algorithms such as Branch and Bound, or heuristic algorithms like PSO, when the number of DSM is known.

2.3 Global Optimisation

2.3.1 Branch and Bound

Branch and Bound[Neu04, TZ88] is an optimisation algorithm particularly suited for solving discrete and combinatorial problems. The basic principle consists in evaluating an objective function $J(\mathbf{X})$ on different subset $S \ni \mathbf{X}$. The lower and upper bounds of the objective function J on each S are used to *prune* safely the solution space. The procedure starts again by splitting the best subset into multiple subsets, until the candidate subset reduces to a single element.

This algorithm is particularly useful when we have an a-priori understanding of the problem structure. Branch and Bound algorithms provide suitable methods for solving many global optimisation problems.

For impulsive space trajectory problems, with gravity assists, the space can be readily split according to the phases of the problem. As the cost is additive along the phases, we can easily prune out the solution space.

For example, suppose an Earth-Venus-Mercury (EVY) transfer. Suppose, that we have one complete solution. We are thus able to give an upper bound C_M on the minimal cost c^* for this problem, over the search sub-space $S = S_{EV} \cup S_{EY}$. Now, for each Earth-Venus leg in $s \subset S_{EV}$, having a cost $c_{EV} > c_M$, we can safely admit that each complete solution, starting with this sequence and for the problem definition in $s \subset S_{EV}$, has a final cost $c_{EVY} = c_{EV} + c_{VY} > c_M$. Thus, we avoid the computation of all subsequent legs following the sub-sequence EV.

2.3.2 Particle Swarm Optimisation

Particle Swarm Optimisation [KE95, PV95] (PSO) algorithms are influenced by natural animal behavior (bird flock and fish school). It permits to search for an optimum of a problem in a hyperspace.

Consider a particle i described by its position $\mathbf{x}_i$ and velocity $\mathbf{v}_i$, at time k, with the solution vector $\mathbf{p}_i$:

$$\mathbf{p}_i^k = \begin{bmatrix} \mathbf{x}_i^k \\ \mathbf{v}_i^k \end{bmatrix}$$

The particle position represents the decision vector. Each particle i moves then in the solution space, with velocity $\mathbf{v}_i^k$. As the solution space directions do not represent the same physical quantities, the decision vector is usually scaled. This then improves the exploration of the space by the particles.

The particle position and velocity are updated at each time k, with the updating dynamical equations:

$$\mathbf{v}_i^{k+1} = \mathbf{v}_i^k + \gamma_i^1(\mathbf{p}_{Li} - \mathbf{x}_i^k) + \gamma_i^2(\mathbf{p}_G - \mathbf{x}_i^k) \tag{2.5}$$

$$\mathbf{x}_i^{k+1} = \mathbf{x}_i^k + \mathbf{v}_i^{k+1} \tag{2.6}$$

where i is the particle index, k is the position in time of the particle, $\mathbf{x}$ and $\mathbf{v}$ are respectively the position and the velocity of the particle, $\mathbf{p}_{Li}$ is the best solution found by the particle, $\mathbf{p}_G$ is the best solution found so far by the whole population (or swarm), and γ_i^1 and γ_i^2 are random numbers of $\Omega(0,1)$. Random initial positions and velocities permit initialising the algorithm.

Most PSO algorithms use a slight modification, including an inertia variable I:

$$\mathbf{v}_i^{k+1} = I^k\mathbf{v}_i^k + \gamma_i^1(\mathbf{p}_i - \mathbf{x}_i^k) + \gamma_i^2(\mathbf{p}_G - \mathbf{x}_i^k) \tag{2.7}$$

The inertia I permits to take or not advantage of the preceding velocity. For example, a decreasing inertia function permits minimising the influence of the past velocity.

The acceleration constant γ^1 has a local influence on the best local optimum: it is the independent behavior. The acceleration constant γ^2 has an influence on the global optimum: it is the social behavior.

This a-priori random behavior allows the algorithm to not get trapped into a local minimum, so that a global optimum may be found. This algorithm is quite close to evolutionary algorithms because of the random part of the algorithm. At the same time, the "crossover" operation between local and swarm optimum has similarities with genetic algorithms.

The parameters that need to be defined are then:

- the number of particles or population size.
- acceleration constants γ^1 and γ^2.
- initial and final weight inertia.
- the initial velocity of the particles.

No general rules exist for choosing them.

2.3.3 Brief synthesis

The advantages of PSO algorithms (or others heuristic algorithms such as DE) are that they are easy to code, have a small computational cost. But the main advantage, like any other stochastic/evolutionary algorithms, is that they do not ask for the value function to be convex or linear and we may lack information on the derivatives. These algorithm can find the best optimum even though the value function is not well behaved.

Their major drawback is that they are slow to converge when close to the optimal point, as opposed to gradient methods (super linear).

DE is known to be one of the most efficient heuristic optimisation algorithm, with fewer function evaluations than PSO for example.

Other global optimisation methods include Interval Arithmetic[Kea97], Homotopy [AG90, HMG04].

Name	Methods	Comment
Deterministic	Branch and Bound, DIRECT, MCS	Deterministic, very expensive
Heuristic	PSO, DE, ...	Evolutionary and Memetics
Stochastic	SA, MONTE-CARLO, ...	random
Interval Arithmetic		exact, global optimum
Local	Continuation, Homotopy	Problem specific

Table 2.1: Global Optimisation methods

Heuristic solvers are widely used because they provide fast results. It is then possible to quickly compute "good" trajectories, and focus on sequence determination. They should however be restricted to preliminary studies, when an a-priori good knowledge of the problem is not required.

On the other hand, deterministic methods ensure we find the global optimum of the trajectory transfer problems, but require more computational time due to the high complexity of the problems. The sequence determination is thus longer.

A deterministic algorithm and systematic study of the sequences should always be preferred to heuristics, even when computational resources are limited. Heuristics should be preferred to understand the a-priori structure of the solution.

In the last section, we provide a new approach that allows to rapidly prune parts of the search space. Combining this approach with deterministic or heuristic methods provides a good algorithm. Note that the quality of the sequence determination is highly dependent on the resolution of the intermediate transfer problems.

2.4 Simplifying the Search Space of Parameterised Trajectories for Global Optimisation

2.4.1 General problem and objectives

Consider the general minimisation problem:

$$\min_{\mathbf{X} \in D_x} J(\mathbf{X}) \tag{2.8}$$

without constraints, where $J : D_x \to \mathbb{R}$ is assumed to be C^2 continuous, and D_x is a bounded set of $\mathbb{R}^M$, $M \in \mathbb{N}$. We suppose J additive.

We want to reduce the complexity of a grid search algorithm (brute search). With gravity assist, the complexity of such algorithm is exponential with the number of phases. We present an approach to have a polynomial complexity.

The following approach has been presented in [Oly07a], where we considered a MGADSM problem. This approach reduces the complexity of the MGADSM problem. The cost of calling a solver increases polynomially with the number of phases considered.

2.4.2 General Approach

Creating independent sub-problems

The exponential complexity is mainly the result of phases that depend of each other because of the swing-by parameters. It is however possible to turn the single initial problem into many smaller sub-problems.

The procedure can be summarised in 2 steps:

1. Following the phase number or a natural decomposition, duplicate each boundary variable and assign a copy to every process that needs the original. The original decision vector $\mathbf{X}$ turns into $\tilde{\mathbf{X}} \in \mathbb{R}^K$, with $K > M$.

2. Partition $\tilde{\mathbf{X}}$ into n variables $\mathbf{x}_i \in \mathbb{R}^N$, and $nN = K = M + D$ where D is the number of duplicated boundary variables. There are as many duplications as junctions.

Each sub-problems are independent, and describe different swing-by possibilities.

Because we now consider the sub-problems independently, we need to set up intermediate constraints to construct complete solutions.

Setting linking conditions

Linking conditions to compute complete solutions are only the result of the duplicated variables. The duplicated variables act as coordination variables between the sub-problems.

Let's find $C \in \mathcal{M}(\mathbb{R})_{D,K}$. K is related to the dimension of the intermediate constraints. For intermediate rendezvous we simply have $K = 0$.

We have an equality condition between the duplicated variables and their respective originals:

$$C \begin{bmatrix} \mathbf{x}_0 \\ ... \\ \mathbf{x}_n \end{bmatrix} = 0 \tag{2.9}$$

The constant matrix C is needed to account for the matching conditions on the duplicated variables, referred to a $\tilde{}$. C is a sparse matrix with one -1 and one 1 element on each line.

It can be decomposed into blocks $C_i \in \mathcal{M}(\mathbb{R})_{D,2N}$ following the problem partitioning:

$$C_i \begin{bmatrix} \mathbf{x}_i \\ \mathbf{x}_{i+1} \end{bmatrix} = 0 \tag{2.10}$$

The constraints ensure we can construct a complete trajectory.

Constructing complete solutions

To solve the original problem, we simply need to solve the sub-problems:

$$\min_{\mathbf{x}_i \in \mathbb{R}^N} J_i(\mathbf{x}_i) \tag{2.11}$$

with the constraints given in equation 2.10.

However, when solving the sub-problem i with decision vector $\mathbf{x}_i$, the decision vector $\mathbf{x}_{i+1}$ (defined in equation 2.10) is unknown. The sub-problem must then be solved for all the different boundary conditions defined by $\mathbf{x}_{i+1}$ in order to be able to construct a complete solution. This can indeed be done with the coordination variables already included in $\mathbf{x}_i$.

Proposition 2.4.1. *Note X_i^F the feasible region for variable $\mathbf{x}_i$, and X^F the feasible region for the original problem. We have:*

$$X^F \subset \bigcup_i X_i^F \tag{2.12}$$

No information has been lost when duplicating the variables and partitioning the problem.

2.4.3 Application: MGADSM Space trajectory

We want to solve a MGADSM problem (see equation 2.3), with $n+1$ phases (n swing-bys). The planet sequence is given by vector $\mathbf{p}$. The objective function is to minimise the characteristic velocity. We consider the following limitations:

- one unique DSM per leg.
- non powered swing-by.

Setting the partitioned problems

With equation 2.3 and the decision vector 2.4, we apply the following partitioning:

$$J_0(\mathbf{x}_0) = \Delta V_1(t_0, V_\infty^1, \phi_1, r_p^1, t_{DSM,1}) + \Delta V_0(t_0, t_{DSM,1}) \tag{2.13}$$
$$J_i(\mathbf{x}_i) = \Delta V_i(t_i, V_\infty^i, \phi_i, r_p^i, t_{DSM,i}) \tag{2.14}$$
$$J_f(\mathbf{x}_f) = \Delta V_n(t_n, V_\infty^n, \phi_n, r_p^n, t_{DSM,n}) + \Delta V_f(t_f, V_\infty^n, \phi_n, r_p^n, t_{DSM,n}) \tag{2.15}$$

where:

- t_0: initial date
- t_i: intermediate date, planet encounter
- t_f: final date
- $t_{DSM,i}$: swing-by date for the phase i
- $[r_p, \phi]$ or $[r_p, V_\infty]$: swing-bys description

The decision vectors $\mathbf{x}_0$, $\mathbf{x}_i$ and $\mathbf{x}_f$ of each sub-problem must describe its entire state space, regardless of the other sub-problems. We use the same description for each leg, as we suppose the same dimension for each sub-problems. The partitioning and the set of variables are resumed on figure 2.5.

And:

$$\mathbf{X}_i = [t_i, V_\infty^i, \phi^i, r_p^i, t_{DSM}^i, \tilde{V}_\infty^i, \tilde{\phi}^i, \tilde{r}_p^i, t_{i+1}] \tag{2.16}$$

Indeed, since the trajectory is forward propagated, each leg needs information from the preceding leg, and must provide information to the following leg. Junction variables r_p and ϕ (i.e. defining the swing-by) are thus duplicated and assigned to their respective sub-problems. Because there is no degree of freedom on the value of t_i, these variables are not duplicated.

Note that $\tilde{\mathbf{V}}_\infty^i$ is indeed the (incoming) hyperbolic excess velocity at the next swing-by.

Proposition 2.4.2. *To simplify the search process, we removed the constraint on the angular deviation of the swing-by. We suppose, a-priori, all the swing-bys are feasible. This point shall be checked later on in the process, once we have complete solutions.*

This allows replacing $\mathbf{V}_{\infty f}$ with $V_{\infty f}$ in the decision vector 2.16. The variables $\tilde{\phi}^i$ and $\tilde{r}_p^i$ can be dropped out.

The decision vector for each sub-problem then reduces to:

$$\mathbf{X}_i = [t_0^i, V_{\infty 0}^i, \alpha_0^i, \beta_0^i, t_{DSM}^i, V_{\infty f}^i, t_f^i]^T \tag{2.17}$$

Where:

- t_0^i is the departure date for the single leg.
- t_{DSM}^i is the date of the Deep Space Manoeuvre (DSM).
- t_f^i is the arrival date for the single leg.
- $V_{\infty 0}^i$ is the hyperbolic excess velocity at departure of the leg.
- α_0^i and β_0^i gives the $\mathbf{V}_{\infty 0}^i$ vector direction.
- $V_{\infty f}^i$ is the hyperbolic excess velocity upon arrival.

Because of this simplification, we cannot check if the swing-byes are feasible or not. We may compute swing-by infeasible trajectories. But this is easily checked when patching the legs together when constructing complete transfers.

Patching Constraints

The tilde ˜variables are a copy of the preceding non tilde variables.

The redundant variables must coincide, so we have of course:

$$\tilde{V}_{\infty}^{i} = V_{\infty}^{i+1} \tag{2.18}$$

This expression allows constructing complete trajectories.

Because of proposition 2.4.2 we do not need to use:

$$\tilde{\phi}^i = \phi_{i+1} \tag{2.19}$$

$$\tilde{r_p^i} = r_p^{i+1} \tag{2.20}$$

Solving the sub-problems

To simplify the resolution of the sub-problems, let's consider discrete and continuous variables of the sub-problem decision vector. The discrete variables are:

$$\mathbf{X}_{Di} = [t_0^i, V_{\infty 0}^i, t_{DSM}^i, V_{\infty f}^i, t_f^i] \tag{2.21}$$

The continuous variables are:

$$\mathbf{X}_{Ci} = [\alpha_0^i, \beta_0^i] \tag{2.22}$$

The sub-problems to solve once $\mathbf{X}_{Di}$ is given, rely only on seeking $\mathbf{X}_{Ci}$, which minimises the functions J_0, J_i and J_f (equations 2.13, 2.14 and 2.15), with the constraints:

$$\Psi_i^1(\mathbf{X}_{Di}) = \left\| \mathbf{V_{p(i)}}(\mathbf{t_0^i}) - \mathbf{V}(\mathbf{t_0^i}) \right\| - V_{\infty 0}^i \tag{2.23}$$

$$\Psi_i^2(\mathbf{X}_{Di}) = \left\| \mathbf{V_{p(i+1)}}(\mathbf{t_f^i}) - \mathbf{V}(\mathbf{t_f^i}) \right\| - V_{\infty f}^i \tag{2.24}$$

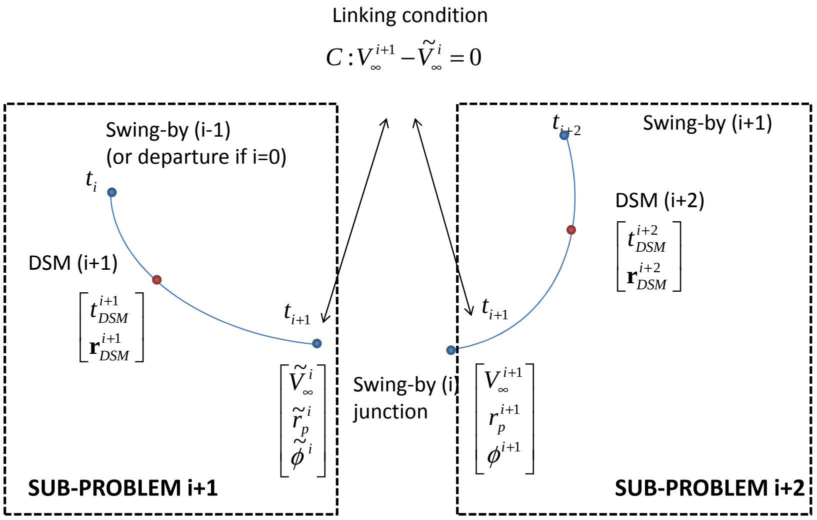
Linking condition
$C : V_\infty^{i+1} - \tilde{V}_\infty^{i} = 0$
Swing-by (i-1)
(or departure if i=0)
t_i
DSM (i+1)
$\begin{bmatrix} t_{DSM}^{i+1} \\ \mathbf{r}_{DSM}^{i+1} \end{bmatrix}$
t_{i+1}
$\begin{bmatrix} \tilde{V}_\infty^{i} \\ \tilde{r}_p^{i} \\ \tilde{\phi}^{i} \end{bmatrix}$
SUB-PROBLEM i+1
Swing-by (i)
junction
t_{i+2}
Swing-by (i+1)
DSM (i+2)
$\begin{bmatrix} t_{DSM}^{i+2} \\ \mathbf{r}_{DSM}^{i+2} \end{bmatrix}$
t_{i+1}
$\begin{bmatrix} V_\infty^{i+1} \\ r_p^{i+1} \\ \phi^{i+1} \end{bmatrix}$
SUB-PROBLEM i+2

Figure 2.5: Decomposition scheme

where $\mathbf{V}_{p(i)}$ and $\mathbf{V}_{p(i+1)}$ are respectively the departure and arrival planets velocity.

This constraint permits to construct solutions with the desired hyperbolic excess velocity upon departure and arrival, describing different swing-by possibilities.

Then, the sub-problems can be formulated as:

$$\forall \mathbf{X}_{Di} \quad \min_{\mathbf{X}_{Ci}} J_i(\mathbf{X}_{Di}, \mathbf{X}_{Ci}) \tag{2.25}$$

$$s.t. \quad \Psi_i^1(\mathbf{X}_{Di}) = 0 \tag{2.26}$$

$$s.t. \quad \Psi_i^2(\mathbf{X}_{Di}) = 0 \tag{2.27}$$

Figure 2.5 shows the sub-problem definition.

Sub-problems must be solved for all points of the map defining $\mathbf{X}_{Di}$ (see algorithm 1).

Remark We now have to solve M problems in a search space of dimension N, whereas in the initial approach we solved 1 problem in a search space of dimension $L > N$, where M is the number of phases, and N is the size of the decision vector for each sub-problem.

The construction of the overall problem solutions, with equation 2.18 and the date of encounter with the planet, takes place once we have computed all the extremals for the $\mathbf{X}_{Di}$ hypercubes. This constructs complete trajectories by patching legs of each of the sub-problem solution spaces.

This approach allows to apply pruning methods on the sub problems, before constructing the solution. In addition the complexity is reduced compared to the initial problem.

Pruning

To limit the search space, we place pruning constraints. Pruning constraints are constraints on the state that avoid seeking in uninteresting sub-spaces without jeopardising any potentially good solutions.

We use the following pruning strategies:

1. constraints on the initial departure manoeuvre $V_{\infty 0} = \Delta V_0$
2. constraints on the final breaking manoeuvre $V_{\infty n} = \Delta V_f$
3. limitation on the swing-bys hyperbolic excess velocity V_∞ and $\tilde{V}_\infty$.
4. constraint on $\Delta V_{DSM(i)}$ amplitude.
5. forward pruning [BMN+05, IBM+06].

When all sub-problems have been solve, each strategy permits to remove branch of the solution tree. Once the sub-solutions have been appropriately sort out, we can construct complete solution, using the constraints describe above. To construct solution, we should construct the solution tree, and with a dynamic programming algorithm we can get the best solution. However, from a mission analyst point of view, we can also construct all possible solution, and then sort them.

Complexity

It is necessary to compute the complexity of the algorithm. In particular, the number of calls to the solver of the sub-problems (partitioned problems defined by equations 2.13, 2.14, 2.15). Pose:

- N_P the number of phases or sub-problems.
- N_{T0} the number of bins of the initial launch date mesh M_{T0}.
- N_t the number of bins in the time of flight mesh M_{T0}.
- N_V the number of bins in the $V\infty$ mesh M_{T0}.

```
Input: Sequence of planets $\{B\}_{i=1..N}$, time grid $[t_0 : dt : t_f]$, time of flight
       grid $[tof_{min} : dt : tof_{max}]$, $[V_{\infty min} : \Delta V_\infty : V_{\infty max}]$ grid
Output: solutions on the grids
List of arrival dates of rank 0: $T_f^0 \leftarrow [t_0 : dt : t_f]$;
foreach i (rank) from 1 to $N-1$ do
    Departure Body = $B_i$;
    Arrival Body = $B_{i+1}$;
    List of departure dates of rank i: $T_0^i \leftarrow T_f^{i-1}$;
    foreach Departure date $t0 \in T_0^i$ do
        foreach Time of flight $tof_i$ in the time grid $[tof_{min} : dt : tof_{max}]$ do
            foreach $V_{\infty i}$ and $V_{\infty j}$ in the $[V_{\infty min} : \Delta V_\infty : V_{\infty max}]$ grid do
                $P = [t0, B_i, B_{i+1}, tof_i, V_{\infty i}]$;
                Minimise $J(P, X_{DSM})$ s.t. $\Psi(V_{\infty j}, B_{i+1}) = 0$;
                Store the solution and associated cost in the table
                $S_{t0,V_{\infty i},tof_i,B_i,B_{i+1}}$.;
            end
        end
    end
    Prune solutions.;
    List of arrival date of rank i: $T_f^i \leftarrow$ List of arrival date of for the
    existing solution of the rank i;
end
Apply dynamic programming to table $S$ and construct complete
trajectories.;
```

Algorithm 1: GASP-DSM algorithm

We can thus construct a mesh for the arrival dates on each phase i: $M^i_{Tf} = M^i_{T0} + M_t$.

For phase i, the number of calls to the solver is:

$$C_i = iN_{T0}(N_tN_V^2) \tag{2.28}$$

From one phase to the next, we have $M^{i+1}_{T0} = M^i_{Tf}$. By a simple recurrence, assuming the mesh step size and the number of mesh points are the same for $T0$ and t, the number of launch date mesh points increases for each phase, from N_{T0} to N_PN_{T0}.

Thus, the number of calls to the solver is given by:

$$C = N_{T0}(N_tN_V^2)N_P\frac{N_P+1}{2} \tag{2.29}$$

The complexity of calling the solver is polynomial. This should be compared to the number of calls that would be required if each possible decision vector of the grids is considered separately. In this case the complexity would be defined by the number of points in the hypercube grid:

$$C_{MAX} = N_{T0}N_t^{N_P}N_V^{N_P-1} \tag{2.30}$$

This approach starts to be beneficial when $M > 1$, but the benefits become great for a high number of legs. More importantly, with this approach we did not make any assumption that would prevent us from finding the global optimum, except that we constrained the body to body legs to have exactly one DSM.

2.5 Conclusions

The problem of global optimisation is a difficult one. In this section, we noted that solving the MGADSM problem was already a difficult task. The question of finding the optimal scenario for this class of trajectory problem is a bit challenging. The difficulty does not rely on the mathematical problem formulation, but more on the computational complexity.

We provided an approach allowing the use of a pruning strategy, and reducing the computational complexity. But even with this approach, we hardly made the problem tractable. A fair amount of time is needed to cope with the computational load of global optimisation problems. The approach described here can be applied to any problems regardless of separability issues.

To cope with the global problem optimisation, we can often use heuristics[VSD05]. Our approach, in comparison, is conservative with respect to the global optimum, and ensures we can find it with reasonable precision. The approach allows parallelling the computations, thus reducing the computational time and giving a rapid solution to the mission analyst.

Chapter 3

Automated Approach for Low-Thrust Interplanetary Trajectories

3.1 Introduction

3.1.1 Low-Thrust Trajectory vs Impulsive Trajectory Sequence Design

As opposed to impulsive trajectories, continuous thrust trajectories have significant burning time. It is not possible to parametrise the control exactly. We need to find a control function $u(t)$ in a Hilbert Space, that minimises a given objective function. For the sequence design, it is important to have a reliable approach.

Indirect methods usually lack of robustness. The basin of attraction is small, and it is difficult to implement an automated program[Ber01]. Considering many possible sequences with an indirect approach can lead to wrong conclusions. If the algorithm fails to converge, we cannot conclude that there is no solution. Direct methods are fast and provide good solutions. Usually, direct methods should be favoured when considering many sequences.

Another approach, which can be even faster than direct methods, is the use of models. It is possible to build an approximation to the optimal control and continuous thrust dynamics with models. Modeling the control consists in choosing a family of functions that best fits the expected optimal control. This approach is useful to initiate exact algorithms. In addition, the algorithm presented in 2.4.2 can be used to efficiently found solutions to transfer problems.

Alemany and Braun[AB07] present the global optimisation problem for low-thrust asteroid selection missions. They briefly survey local optimisation methods for low-thrust trajectory optimisation and global optimisation. They mention the prohibitive amount of calculation needed to tackle this global optimisation problem. They define the global optimiser feature we should look for to solve this problem.

3.1.2 Examples of Low-Thrust Global Optimisation Problems

GTOC2 (2006)

GTOC2 was the second issue of the Global Trajectory Optimisation Competition, initiated by the Advanced Concept Team of ESA/ESTEC. It was organised in 2006, by the JPL, the winning team of the 2005 edition. We recall the problem.

> **Problem description** The problem is a multiple asteroid rendezvous. A low-thrust trajectory must be designed including rendezvous manoeuvre with one asteroid from each of four defined groups of asteroids.

> **Mission and Engineering Constraints** The spacecraft is to launch from the Earth, with a hyperbolic excess velocity of up to 3.5 km/s and of unconstrained direction. The year of launch must lie in the range 2015 to 2035, inclusive. After launch, the spacecraft must rendezvous with one asteroid from each group. (...) A stay time of at least 90 days is required at each of the first three asteroids. The flight time from launch to the final rendezvous with the fourth asteroid, must not exceed 20 years. Gravity assists are not permitted. Objective of the optimisation is to maximise the quantity:
>
> $$J = m_f/t_f$$
>
> The spacecraft has a fixed initial mass of 1500 kg (...). The propulsion is by means of a thruster (...) has a constant specific impulse of 4000s (...) maximum thrust level of 0.1N. (...).

Figure 3.1 shows the best reported solution. This problem is very difficult due to the combinatorial complexity. The group order is of importance and allows to reduce the complexity. Most teams found the same group order. Seeking the asteroid sequence is also difficult. Most methods rely on heuristics and experience.

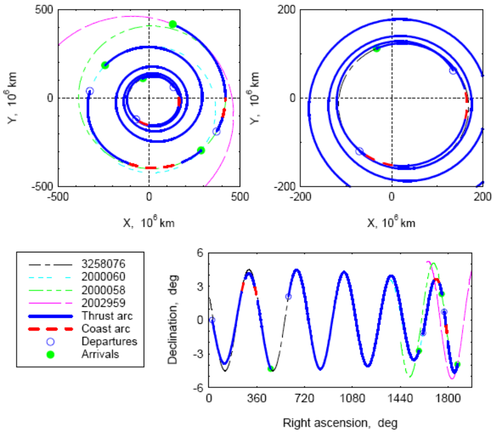

Figure 3.1: GTOC2 best reported solution (Polytecnico di Torino, Italy, 2006)

GTOC3 (2007)

GTOC3 was organised by Politecnico di Torino, Italy. The problem deals with a multiple near-Earth asteroid (NEA) rendezvous mission. The spacecraft is launched from Earth, must rendezvous with three asteroids from a specified group of NEAs, and return to Earth. The maximum mission duration is limited to 10 years.

Spacecraft and Trajectory Constraints The spacecraft is to launch from the Earth, with hyperbolic excess velocity $v1$ of up to 0.5 km/s and of unconstrained direction. The year of launch must lie in the range 2016 to 2025, inclusive. After launch, the spacecraft must first rendezvous with three different asteroids (...), and then rendezvous with the Earth. The choice of the asteroids is part of the optimisation process. The stay times at each of the three asteroids (τ_1, τ_2, τ_3), must be longer than 60 days. The flight time, τ, measured from launch up to the point of rendezvous with the Earth, must not exceed 10 years. Only gravity assists from the Earth are permitted. The spacecraft has a fixed initial mass mi of 2000 kg (it does not change with launch $v1$). The propulsion is by means of a thruster, which can be turned on or off at will, has a constant specific impulse Isp of 3000 s, and has a maximum thrust level T of 0.15 N. There is no constraint on the thrust direction. The spacecraft mass only varies because of the propellant consumption during thrusting and is otherwise constant (no mass dumping or collecting is allowed). (...)

The problem is quite similar to GTOC2. The performance index to maximise is given below.

Performance index Objective of the optimisation is to maximise the non dimensional quantity

$$J = \frac{mf}{mi} + K\frac{\min_{j=1,3} \tau_j}{\tau_{max}}$$

where m_i and m_f are the spacecraft initial and final mass, respectively; τ_j, with $j = 1, 3$, represents the stay-time at the j-th asteroid in the rendezvous sequence and:

$$\min_{j=1,3} \tau_j$$

is the shortest asteroid stay-time; $\tau_m ax = 10$ years is the available trip time, and $K = 0.2$. The performance index is chosen in order to favour low propellant consumption (i.e., large payload) and long stay-times on the asteroids, thus increasing mission scientific return.(...)

As for GTOC2, the problem was to find the best Asteroid Sequence among a restricted set of asteroids, with low thrust propulsion, although, contrary to GTOC2, the restricted set was of lesser size and did not pose any combinatorial issue. The great difficulty of the problem was actually the possibility to use Earth swing-bys.

The best submitted solutions, all considered 3 to 4 swing-by of Earth. The worst solutions did not consider any. From this point, it was important to question their utility, and how to place them. The problem was not combinatorial but relied hugely on the quality of the local optimisation, and the optimisation of the swing-bys. The more swing-byes the solver was able to handle, the more chance we might have to find the best solution.

Figure 3.2 shows the best solution found for this problem. It was identified by the CNES. They use an indirect method approach for the transfer problem. They were thus able to find a very accurate optimal control. However, the sequence of asteroids was found with heuristics. The use of swing-byes mainly followed experience and common sense. In other cases, a systematic approach should be considered.

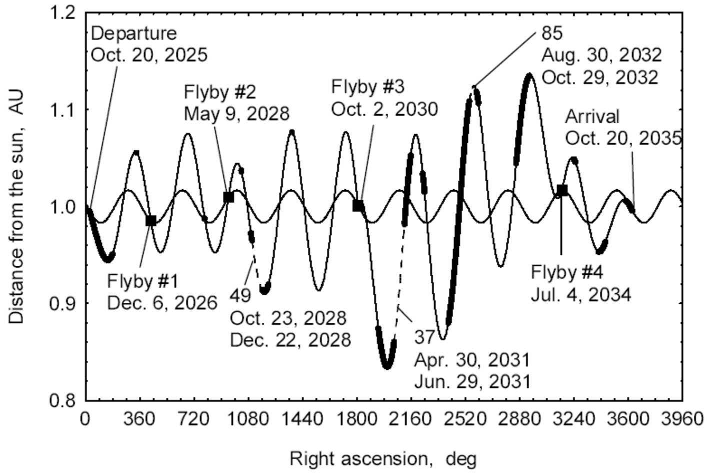

Figure 3.2: GTOC3 best reported solution (CNES, France, 2007)

3.2 Current Models and Limitations

So far, exponential sinusoids [Pet02, PL01, Izz06] are used because they proved their efficiency and their ease of use. However one can emphasise that the initial guess proposed by exponential sinusoid methods has sometimes a cost very far from that of the optimised trajectory. The differences essentially come from a control that is continuously thrusting, with no bang-bang sequences. And the model cannot correctly describe the rendezvous phase. The latter is of great importance and a major issue in most low thrust models. Only shaping methods through parametrised pseudo equinoctial elements [VD06] manage to solve the problem. But they present a violation of the dynamics and errors when propagated. Markopoulos [MC95] introduced a thrusting program, which includes a throttle parameter. In addition, his model allows multiple switching; however he didn't treat the case with the zero throttling parameter which would be needed for coast segments.

In [VSJ05], the authors survey the just mentioned low thrust models and propose a program for the global optimisation of multi-gravity assist low thrust trajectories. They also study the optimality of the exponential sinusoid. Their conclusion is that this model is far from satisfying the necessary condition of optimality, unlike the pseudo-equinoctial elements model. Curiously though, the exponential sinusoid model provides a cost closer to the optimum than the pseudo equinoctial model, whereas the latter provides more flexibility.

One of the most interesting models is the one of Pinkham, who models the thrust instead of the shape of the trajectory to get his low thrust model. Consequently, it is possible to construct a multi-level trajectory, however no such results have been found in the literature.

Sims et al[SF99] use a different approach from the shape based ones. They discretise the trajectory into segments, with an impulse at the middle, and matching conditions at junctions. Then, they formulate a nonlinear programming problem. According to the results published, this method is robust and agrees with more precise indirect method tools.

We introduce a model implementing coast arcs [Oly07b], which improves our low thrust trajectory modeling with few parameters. The model provides the thrusters capability, and informs about the trajectory feasibility. Some conditions on the choice of the model parameters will be given. We shall introduce an optimisation problem that gets the optimal value for the parameters under specific constraints. It is expected, that through this optimiser we can get a good initial guess that can be close to the optimal low thrust trajectory. We show that we manage to get a better cost than any other shape based low-thrust models. We also show that under some conditions, we can reduce the rendezvous manoeuvre cost.

3.3 A Continuous Thrust Model

3.3.1 Dynamics

Our goal is not to introduce another function in the field of shape based methods, but rather to have a formulation as close as possible to optimal low thrust arc. We shall then further introduce switching points between coast (no thrust) - thrust sequences in the trajectory. Bang-bang control is indeed the optimal control for a low-thrust, maximum mass, with final constraints, space trajectory transfer problem.

To simplify our program and model, we need to find a model that is valid for both Keplerian and continuous thrust transfers.

We use the following model equation, in its domain of definition:

$$r(\theta, k) = \frac{p}{1 + e\cos(k\theta - \phi)} \tag{3.1}$$

Where θ is the polar angle, k is what we will later on call the thrust parameter, ϕ is a phasing parameter, p and e are analogous to respectively the Keplerian parameter and eccentricity. This model has been derived from the classical formula for conics, and it is referred as dilated conic. It has been enumerated among others in the endless list of potential shape-based trajectories in the conclusions of Petropoulos' thesis[PLB00, Pet02, PL01]. It is also very similar to Pinkham's spiral, which includes additional exponential terms that allow reaching hyperbolic speeds after an initial spiralling movement. This last feature is not of interest in our case as our goal is to introduce a preliminary design tool for interplanetary transfer, with final rendezvous conditions. Other models are possible. The approach followed here simply extend the model to general multi-thrust level segments.

The advantage of model 3.1 comes from the very few parameters (p, e, k and ϕ) and the possibility to model coast legs. This results in a simplified program.

This will permit to introduce a low thrust gravity assist program that can reliably enumerate preliminary trajectory design options.

3.3.2 Geometrical properties

Using equation 3.1, for all points $(R_1, \theta_1) \in \mathbb{R} \times [0, 2\pi]$ and $(R_2, \theta_2) \in \mathbb{R} \times [0, 2\pi]$, for a given transfer angle $\bar{\theta} = \theta_2 - \theta_1$, there exist $\phi \in \mathbb{R}$ and $k \in \mathbb{R}$ such that:

$$p = R_1 R_2 \frac{\cos\phi - \cos(k\bar{\theta} - \phi)}{R_1 \cos\phi - R_2 \cos(k\bar{\theta} - \phi)} \tag{3.2}$$

$$e = \frac{1}{\cos\phi}\left(\frac{p}{R_1} - 1\right) \tag{3.3}$$

Define a continuous thrust trajectory from (R_1, θ_1) to (R_2, θ_2). The trajectory lies in the plane defined by the normal $\mathbf{R}_1 \times \mathbf{R}_2$.

Proposition 3.3.1. *If $\{p, e, k, \phi\}$ is a transfer from $\mathbf{R}_1(R_1, \theta_1)$ to $\mathbf{R}_2(R_2, \theta_2)$, then $\{p, e, -k, -\phi\}$ represent the exact same transfer.*

We define the *flight path angle* as the angle between the velocity vector and the unit orthoradial vector $\mathbf{u}_\theta$ (see figure 3.3). Thus, the expression for the flight path angle is:

$$tan\gamma = \frac{ek\sin(k\theta - \phi)}{1 + e\cos(k\theta - \phi)} \tag{3.4}$$

We define the *flight path direction* as the direction following the velocity vector.

The ratio ϕ/k represents the angular distance between $\theta = 0$ and the most distant, respectively closest, point of the inward, respectively outward, trajectory from the central body.

Figure 3.4 represents part of the set of trajectories that geometrically match the initial and final position, for different values of the parameter k, and $\phi = 0$.

3.3.3 Physical properties

We are interested in defining properties such as the control, time of flight, consumption, and velocity along the trajectory. In this section, we study the single

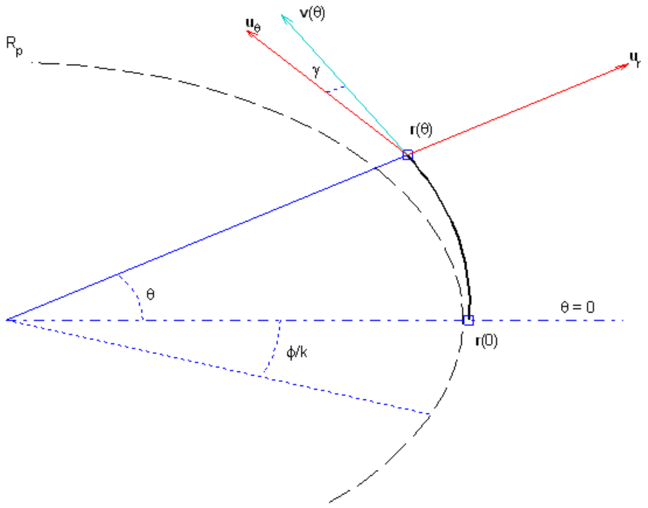

Figure 3.3: Geometrical configuration

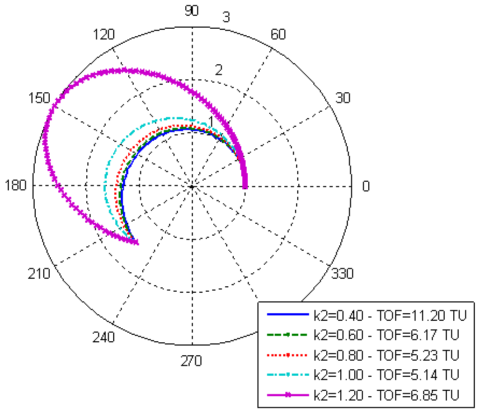

Figure 3.4: Set of transfer legs. $R_1 = 1$, $R_2 = 1.5$, $\bar{\theta} = 5\pi/4$

segment transfer from $\mathbf{R}_1(R_1, \theta_1)$ to $\mathbf{R}_2(R_2, \theta_2)$, $[R_1, R_2] \in \mathbb{R}^2$ and $[\theta_1, \theta_2] \in [0, 2\pi]^2$.

Control Steering

With the control:

$$\mathbf{u} = \cos\alpha \mathbf{u}_\theta + \sin\alpha \mathbf{u_r} \tag{3.5}$$

The equations of motion give the angular momentum which leads to:

$$\frac{dr}{d\theta}{V_\theta}^2 + rV_\theta\frac{dV_\theta}{d\theta} = F_{th}r^2\cos\alpha \tag{3.6}$$

Where F_{th} is the thrust acceleration, α is the thrust direction with respect to the unit orthoradial vector $\mathbf{u}_\theta$.

Proposition 3.3.2. *The trajectory shape given by equation 3.1 can be followed by many different control laws, according to the thrust direction and amplitude.*

Radial and orthoradial thrusts have already been proposed in the literature [Bat01, Bol91, Bol92] for other models. The control law we seek is the one which would apriori reduce consumption while following the trajectory shape. The best approach would be to find the control that minimises consumption while respecting the model geometry. We have to select F_{Th} and/or $\mathbf{u}$ that follow the model.

With the polar dynamical equations for the state variables $\mathbf{x} = [r, \theta, V_r, V_\theta]$, we construct the Hamiltonian H introducing the costate variables $\Lambda = [\lambda_r, \lambda_\theta, \lambda_{V_r}, \lambda_{V_\theta}]$:

$$H(\mathbf{x}, \Lambda, \alpha; t) = \lambda_r V_r + \lambda_\theta\frac{V_\theta}{r} + \lambda_{V_r}\left(-\frac{\mu}{r^2} + \frac{V_\theta^2}{r} + F_{th}\sin\alpha\right) + \lambda_{V_\theta}(-\frac{V_rV_\theta}{r} + F_{th}\cos\alpha) \tag{3.7}$$

A necessary condition of optimality gives:

$$\tan\alpha^* = \frac{\lambda_{V_r}}{\lambda_{V_\theta}} \tag{3.8}$$

Thus, we assume the optimum thrust is in the flight-path direction ($\alpha = \gamma$). We do not apriori satisfy the necessary condition of optimality for the thrust amplitude. This is because we want to seek F_{Th} that follows the geometrical shape defined by equation 3.1. This problem is less than obvious. In the next section 3.3.4, we will show that it is not always possible to find a good F_{th}.

With a derivation wrt θ, and after simplifications, we get the thrust force:

$$F_{th}\left(\frac{r^2}{\mu}\right)\cos\gamma = a \tag{3.9}$$

With:

$$a = \frac{ek\sin(k\theta - \phi)}{2} \frac{(1 + e\cos(k\theta - \phi))(1 - k^2)}{(1 + e\cos(k\theta - \phi)(1 - k^2))^2} \tag{3.10}$$

Or in a compact form:

$$Fth = \frac{tan\gamma}{2\cos\gamma}(1 - k^2)\frac{r^4}{\mu}\left(\frac{d\theta}{dt}\right)^4$$

The parameter k plays the role of a throttling parameter*.

Remark The case $k = 0$ will now be omitted as it does not represent any transfer ($R_1 = R_2$ and $\tan\gamma = 0$).

Clearly, a condition to have a low thrust trajectory is to have $a < 1$ (equation 3.10). That simply means that the thrust acceleration is small compared to the local gravitational acceleration.

The geometry of such a transfer can be seen on figure 3.3.

Time of flight

Because of the assumption on the thrust vector being aligned with velocity, expressions are simplified and we can express the angular velocity with:

$$\frac{d\theta}{dt}^2 = \frac{\mu}{r(k,\phi,\theta)^4} \frac{p(k,\phi)}{1 + e(k,\phi)(1 - k^2)\cos(k\theta - \phi)} \tag{3.11}$$

where μ is the attractive body gravitational constant.

Again, for the particular values $k \in \{-1, 1\}$ and $\phi = 0$ we find the classical equation from Kepler's second law of planetary motion [Bat01].

Then, when $\theta :\longrightarrow \theta(t)$ defines a diffeomorphism, the time of flight is simply expressed as an analytical integral:

$$T = \sqrt{\frac{1}{p(k,\phi)\mu}} \int_{\theta_0}^{\theta_f} r(k,\phi,\theta)^2 \sqrt{1 + e(k,\phi)(1 - k^2)\cos(k\theta - \phi)} d\theta \tag{3.12}$$

A complicated integration shows that this integral can be expressed with Elliptical integrals[Bat01].

Equation 3.12 permits to get the lower limit in the time of flight:

$$T_0 = \sqrt{\frac{p(k = 0,\phi)^3}{\mu}}(\theta_f - \theta_0) \tag{3.13}$$

Proposition 3.3.3. *There is a minimum time that ensures the existence of a solution for the model.*

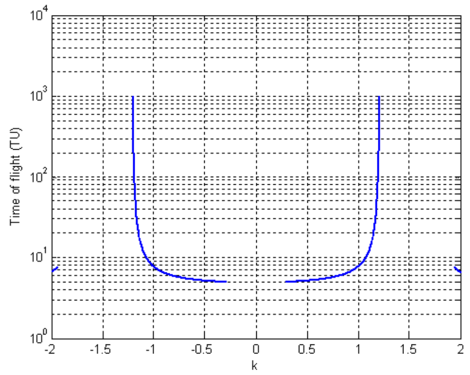

Figure 3.5: Time of flight, $\theta_2 - \theta_1 = 4\pi/3$, $R_1 = 1$, $R_2 = 1.5$, $\mu = 1$, $\phi = 0$

Solutions to the transfer problem with a single segment exist only for a particular interval of time of flight. The lower bound of the interval corresponds to a circular transfer for which $R_1 = R_2$.

From figure 3.5 we note that for a single transfer segment, the solution is not unique. This implies that we can have different thrust level solutions for the same transfer problem.

Consumption

Assuming equation 3.9 is well defined and bounded, the expression of the consumption, for a constant specific impulse, is simplified to:

$$\Delta V_t = \frac{1}{\mu}|1-k^2| \int_{\theta_0}^{\theta_f} |\frac{tan\gamma}{2\cos\gamma}|r^4|\frac{d\theta}{dt}|^3 d\theta \tag{3.14}$$

where μ is the attractive body gravitational constant.

As can be seen on figure 3.6, the values of k around 1 from below, are those which permit to get a low consumption. The two vertical asymptotes at $k = -1$ and $k = 1$, on the semi-logarithmic plot, indicate the $0 - \Delta V$ consumption points.

Because of the integration of an absolute quantity for the consumption function (equation 3.14), the consumption is not differentiable around $k = 1$ and $k = -1$.

Using proposition 3.3.1, we can limit the range of k to positive values.

Velocity

For the rendezvous transfer, it can be of interest to compute the velocity vector on the trajectory. In the transfer plane, writing the radial and orthoradial velocity

[*] For $k \in \{-1, 1, 0\}$, we get a ballistic transfer: $Fth = 0$.

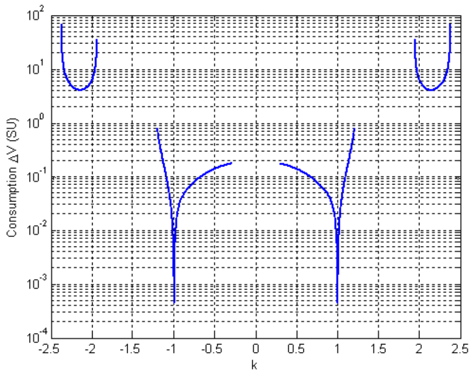

Figure 3.6: Consumption, $\theta_2 - \theta_1 = 4\pi/3$, $R_1 = 1$, $R_2 = 1.5$, $\mu = 1$, $\phi = 0$

and using the definition of the flight path angle, gives:

$$\mathbf{V} = R\frac{d\theta}{dt}\left(\mathbf{u}_\theta + tan\gamma\mathbf{u_r}\right) \tag{3.15}$$

$\mathbf{u}_r$ and $\mathbf{u}_\theta = \mathbf{u}_h \times \mathbf{u}_r$ are respectively the radial and orthoradial vectors. $\mathbf{u}_h$ is the normal to the trajectory plane, co-linear with the momentum vector. The trajectory plane is defined by the initial and final positions, and the position of the sun or the main attractive body.

3.3.4 Existence of solutions

The existence of solutions to the model is obvious by considering the duality of this model with Lambert's problem ($k = 1$). Any ballistic solution of Lambert's problem, is available through this model. But this ballistic solution does not necessary comply with the low thrust condition ($|a| > 0$), as we do not thrust at all!

The transfer from $\mathbf{R}_1(R_1, \theta_1)$ to $\mathbf{R}_2(R_2, \theta_2)$ is coplanar. Considering a single transfer segment (only a thrust arc), the physical transfer is determined by the variables R_1, R_2 and $\theta_2 - \theta_1$. And we seek the model variables k and ϕ. Then for a single segment transfer the thrust segment is completely described with k and ϕ.

We define the physically feasible region D_f with:

$$D_f = \left\{ (k,\phi) \in \mathbb{R}^{+2} s.t. \left\{ \begin{array}{r} r(k,\phi,\theta) > 0 \\ \frac{p}{1+e(1-k^2)\cos(k\theta-\phi)} > 0 \end{array} \right\} \forall \theta \in [0, \bar{\theta}] \right\} \tag{3.16}$$

This ensures the radius to be positive, and the angular velocity to be correctly defined.

We have the following proposition.

Proposition 3.3.4. *Consider the points* $\mathbf{R}_1(R_1, \theta_1)$ *and* $\mathbf{R}_2(R_2, \theta_2)$ *(*$R_1 \neq R_2$*), and a transfer angle* $\bar{\theta} = \theta_2 - \theta_1 + 2n_{rev}\pi > 0$*,* $n_{rev} \in \mathbb{N}$*. The physically feasible region* D_f *is given by:*

$$D_f = \left\{ (k, \phi) \in \mathbb{R}^{+2} s.t. \left\{ \begin{array}{r} p(k,\phi) > 0 \\ |e(k,\phi)| < 1 \\ |1 - k^2| \quad \leq 1 \end{array} \right\} \right\} \tag{3.17}$$

Proof. The positivity of p is required by equation 3.12.

We look for the domain of definition for e. Pose: $c_m = \min_{\theta \in [0,\bar{\theta}]} \cos(k\theta - \phi)$ and $c_M = \max_{\theta \in [0,\bar{\theta}]} \cos(k\theta - \phi)$.

Then:

$$0 < 1 + ec_m < 1 + e\cos(k\theta - \phi) < 1 + ec_M$$

There are 3 cases.

- if $c_m < c_M < 0$ then $0 < e < \frac{-1}{c_m}$.
- if $c_m < 0 < c_M$ then $\frac{-1}{c_M} < e < \frac{-1}{c_m}$.
- if $0 < c_m < c_M$ then $\frac{-1}{c_M} < e < 0$.

It remains to check the domain of definition of k. We must have: $e(1 - k^2)\cos(k\theta - \phi) > -1$. This leads to the condition: $-\frac{c_M}{c_m} \leq 1 - k^2 \leq 1$.

Accepting any possible transfers, and in particular those for which $\bar{\theta} > 2\pi$, we have $c_m = -1$ and $c_M = 1$. The conditions of feasibility turn to be: $|e| < 1$ and $|1 - k^2| \leq 1$.

□

The variables p, e and $\bar{\theta}$ give the range of values for the parameter ϕ for an inbound ($R_2 < R_1$) or outbound ($R_1 < R_2$), and a prograde ($\dot{\theta} > 0$) or retrograde ($\dot{\theta} < 0$) orbit.

Because of the assumption of positiveness of p, ϕ must be in a particular region D_ϕ. We note:

$$D_\phi(k, \bar{\theta}, R_1, R_2) = \left\{ \phi \in \mathbb{R} - \mathbb{Z}\pi/2 s.t. \left\{ \begin{array}{rl} \cos\phi & \geq \cos(\phi - k\bar{\theta}) \\ \frac{R_1}{R_2}\cos\phi & > \cos(\phi - k\bar{\theta}) \end{array} \right\} \right\} \tag{3.18}$$

Indeed, ϕ is explicitly related to the initial γ_0 (see equation 3.4). Choosing ϕ selects the initial V_∞ slope.

3.4 Multi-thrust Segment Transfer

3.4.1 Construction of Coast - Thrust Control

Because of the possibility to model low thrust arcs as well as ballistic arcs, it is appealing to build mixed transfers with thrusting and coasting phase. This leads to a simplified model. We do not need to introduce other models for the ballistic transfer.

At a coast - thrust switching point (the case thrust - coast being similar) we have:

$$k^- = 1 \tag{3.19}$$

$$r^- = r^+ \tag{3.20}$$

$$\gamma^- = \gamma^+ \tag{3.21}$$

$$\frac{d\theta^-}{dt} = \frac{d\theta^+}{dt} \tag{3.22}$$

The superscripts − and + stand respectively for the state before (coast) and after (thrust) the switching point.

Using equation 3.1, we have to solve the following system:

$$\frac{p^-}{1 + e^- \cos(\theta^{\bar{-}} - \phi^-)} = \frac{p^+}{1 + e^+ \cos(\phi^-)} \tag{3.23}$$

$$\frac{p^-}{1 + e^- \cos(\theta^{\bar{-}} - \phi^-)} = \frac{p^+}{1 + e^+ \cos(\phi^-)} \tag{3.24}$$

$$p^- = \frac{p^+}{1 + e^+ \cos(\phi^-)(1 - {k^+}^2)} \tag{3.25}$$

Where $\bar{\theta}$ is the angular position of the switching point. The boundary conditions on the radiuses (r_0, r_f) complete the problem.

3.4.2 Formulation of the multi-switch transfer

The general transfer problem from $\mathbf{r}_0$ to $\mathbf{r}_N$, during time duration $\bar{T}$, comprises N legs, with $N - 1$ switching points r_i, $i \in I = [\![1, N]\!]$, between the coast phases and the thrusting phases.

Legs description is given by the variables : $P_i = \{r_i, r_{i+1}, e_i, p_i, k_i, \phi_i, \bar{\theta}_i\}$. As we define the trajectory as thrust - coast sequences, we define T the set of odd indices in I and C the even indices in I. Thus, P_i for $i \in T$ defines the thrust segments parameters, while P_i for $i \in C$ defines the coast segments parameters.

As opposed to the single thrust segment transfer, we need all the variables to account for the junction conditions 3.19. We can however reduce this search space by considering the model dynamics.

To solve the transfer problem we should seek P_i for all $i \in I$ with $3N - 1$ equations.

For all coast arcs $c \in C$, we have $k_c = 1$ and $\bar{\theta}_c$ is a free variable. Variables e_c, ϕ_c are given by the preceding thrust leg parameters P_t, $t \in T$. Indeed, coast arcs c are only described by their sweep angle θ_c because the coast arcs result from the Keplerian propagation of the final state of the preceding thrust arc.

Equation 3.23 gives explicitly the variable e^+, such that the variables e_i for all $i \in \{2, ..., N\}$ are completely determined, once e_1 is given.

In addition, the choice of the parameters P_i for all $i \in I$ completely describes the sequence of intermediate points $r_{i \in 1,...,N-1}$. The decision vector is then described with a very few parameters:

$$Z = [e_1, \{k_i, \phi_i, \bar{\theta}_i, \bar{\theta}_j\}_{i \in T, j \in C}] \tag{3.26}$$

The multi thrust - coast sequence trajectory is then mainly described by the arc sweep angles $\bar{\theta}_i$, the throttling parameter k_i of each thrust arc, and the phasing parameters ϕ_i for all arcs.

Then assuming the switching points are given, coast arcs need 1 single variable and thrust arcs need 2 variables.

We can then also eliminate N equations, as all the positions are determined. From 3.23, there is now only one equation to use (second or third). We then have only $(N_{Thrust} - 1) + 3$ equations to solve. The defect conditions to satisfy are expressed in equation 3.23 and figure 3.7. The initial condition is on the departure radius and angular position.

$$\xi = \begin{bmatrix} \xi_i \\ \xi_r = r_n - r_f \\ \xi_\theta = \bar{\theta} - \sum_{j=1,N} \bar{\theta}_j \\ \xi_T = \bar{T} - \sum_{j=1,N} t_j \end{bmatrix} = 0 \tag{3.27}$$

With:

$$\xi_i = \frac{p_{2i}}{p_{2i+1}} \frac{e_{2i+1} k_{2i+1} \sin \phi_{2i+1}}{e_{2i} k_{2i} \sin \phi_{2i\bar{\theta}_{2i} - \phi_{2i}}} \tag{3.28}$$

for all $i \in T$.

Variables e_i are given by equation 3.23 for the thrust leg and equation 3.4 for the coast legs. Variables p_i are then given by equations 3.2 and 3.3.

3.4.3 Number of switching points

There is a gain in controllability that goes with the increasing number of switching and the time duration of the arcs. Including coast arc limits the error in control compared to the optimal one, as the trajectory is less prone to the model assumptions.

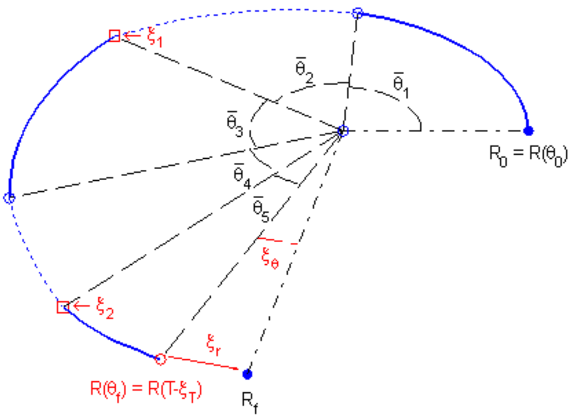

Figure 3.7: Defect conditions overview

As the program proposed by Markopoulos[MC95] suggests, a one segment solution for a particular transfer problem (time of flight and dates given) may not in general always exist. Thus he seeks the transfer in the space of feasible trajectories. Our problem, because of its close resemblance to Lambert's problem, always allows for a solution, which might however not be a continuous thrust solution.

The choice of the switching number can be done by increasing it till the cost benefit becomes small enough to be considered insignificant. The constraint feasibility is also an argument for increasing the number of switchings. Indeed, for a given number of revolutions, increasing the number of switching points permits to get degrees of freedom such that we can reduce boundary constraints violations. For rendezvous transfer problem, this approach asymptotically approaches the optimal low-thrust problem.

Figure 3.9 shows that when increasing the number of switchings the final rendezvous ΔV can be decreased. In addition, it also emphasises that it is better to place a final thrust arc to reduce the rendezvous penalty.

On figures 3.8 and 3.9, the objective function to minimise was, for the purpose of the illustration, the sum of all consumption and delta-V (the initial and the final impulses). The launch date was kept fixed.

Remark We can reduce the final hyperbolic excess velocity either by increasing the number of revolutions of the trajectory, or by improving controllability by adding thrust - coast sequences.

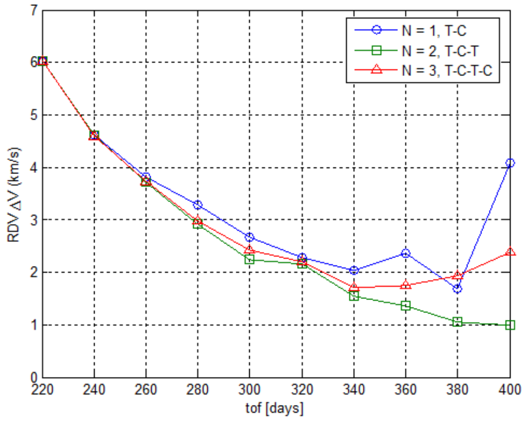

Figure 3.8: Influence of the number of switchings over the consumption for a departing leg. Earth-Mars transfer, launch in 2004, time of flight ranging from 200 to 400 days.

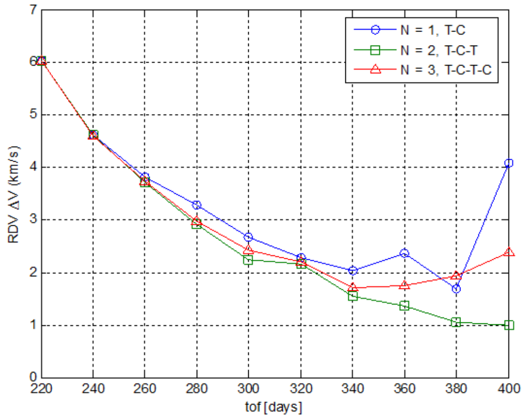

Figure 3.9: Influence of the number of switchings over the rendezvous manoeuvre. Earth-Mars transfer, launch in 2004, time of flight ranging from 200 to 400 days.

3.5 Optimisation problem

3.5.1 Formulation for the parameterised trajectory problem

Direct transfer problem

The biggest challenge in the resolution is to find the switching points that satisfy the constraints qualification, and a cost close to the infimum.

We then propose to formulate an optimisation problem:

$$\min_{\mathbf{z}\in D_f} J(\mathbf{z}) = \int_{t_0}^{t_f} F_{th}(\mathbf{z})dt = \Delta V_t(\mathbf{z}) \tag{3.29}$$

under the constraints:

$$\Delta V_{0min} \leq \Delta V_0 \leq \Delta V_{0max} \tag{3.30}$$

$$\Delta V_{fmin} \leq \Delta V_f \leq \Delta V_{fmax} \tag{3.31}$$

$$\xi(\mathbf{z}) = 0 \tag{3.32}$$

and the initial conditions:

$$r(\theta_0) = r_0 \tag{3.33}$$

$$\theta(t_0) = \theta_0 \tag{3.34}$$

with $\mathbf{z}$ being described by 3.26.

It has been observed that usually this problem leads to a Lambert's problem solution. The sweep angles of the thrust arcs either reduce to zero or the throttling parameter tends toward 1. To avoid this undesirable, but interesting case as it motivates an initialisation algorithm, we added the constraint on the launch velocity ΔV_0 and final rendezvous manoeuvre ΔV_f.

RendezVous problem

The formulation is the same as in section 3.5.1, except that the cost includes the final braking manoeuvre.

$$\min_{\mathbf{z}\in D_f} J(\mathbf{z}) = \Delta V_f(\mathbf{z}) + \Delta V_t(\mathbf{z}) \tag{3.35}$$

The rendezvous problem is described by constraints on position and velocity. In the continuous thrust case, the final braking manoeuvre usually reduces to zero. The best solution is selected according to the lowest rendezvous manoeuvre ΔV_f, and the lowest consumption.

MGA trajectory problem

We can solve the general MGA problem using the approach described in section 2.4.2. The gravity assist is not powered and modelled in the patched conic approximation. It is then important to study all possible values of V_∞ at interior points to span the search space.

We then consider the minimisation sub-problems:

$$\min_{\mathbf{z}_0 \in D_f} J_0(\mathbf{z}_0) = \Delta V_0(\mathbf{z}_0) + \Delta V_t(\mathbf{z}_0) \tag{3.36}$$

$$\min_{\mathbf{z}_i \in D_f} J_i(\mathbf{z}_i) = \Delta V_t(\mathbf{z}_i) \tag{3.37}$$

$$\min_{\mathbf{z}_f \in D_f} J_f(\mathbf{z}_f) = \Delta V_t(\mathbf{z}_i) + \Delta V_f(\mathbf{z}_f) \tag{3.38}$$

under the constraints:

$$\Delta V_{0min} \leq \Delta V_0 \leq \Delta V_{0max} \tag{3.39}$$

$$\Delta V_{fmin} \leq \Delta V_f \leq \Delta V_{fmax} \tag{3.40}$$

$$\xi(\mathbf{z}_i) = 0 \tag{3.41}$$

and the initial conditions:

$$r(\theta_0) = r_0 \tag{3.42}$$

$$\theta(t_0) = \theta_0 \tag{3.43}$$

Each sub-problem can have a different number of thrust segments. They are independent from each other following section 2.4.2.

Lastly, among the set of solutions, we keep those, which are feasible and have a reasonable swing-by altitude. The altitude is calculated using equations 1.19 and 3.15.

The best solution is selected according to the rendezvous manoeuvre ΔV_f, the launch velocity ΔV_0, or the consumption.

3.5.2 Algorithm

To solve the multi-thrust-segment transfer, with or without gravity-assist, we split the decision vector is local and global variables. The local variables k_i, ϕ_i and θ_i are sought with a local SQP solver. These variables describe the thrust segments. The remaining variables, or global variables, θ_{2i} describe the sweep angles of each coast segment. They are sought with a grid search approach.

3.5.3 Local Solver for the Thrust Segments

We introduce the Lagrangian variable vector ν associated with the constraints ξ. With the necessary conditions of optimality, which express the stationarity of the

Lagrangian, we get the following system to solve:

$$F(S) = \begin{bmatrix} \xi(e_1, \bar{\theta}_i, k_i, \phi_i) \\ \frac{\partial J}{\partial e_1} + \nu^T \frac{\partial \xi}{\partial e_1} \\ \frac{\partial J}{\partial k_i} + \nu^T \frac{\partial \xi}{\partial k_i} \\ \frac{\partial J}{\partial \phi_i} + \nu^T \frac{\partial \xi}{\partial \phi_i} \\ \frac{\partial J}{\partial \theta_i} + \nu^T \frac{\partial \xi}{\partial \theta_i} \end{bmatrix} = 0 \tag{3.44}$$

for all $i \in T$

The derivatives of the constraints can be analytically calculated. A symbolic computation tool, or an automatic differentiation code such as ADiMAT[BLV03], permits to overcome a great amount of calculation.

Matrix of equation 3.45 shows the parts of the Jacobian matrix which can be analytically calculated, and those which pose more problems or are the result of transcendental equations and require a numerical computation (noted nc). Theoretically all the elements of the Jacobian matrix (constraints) can be calculated, only the constraint on the time of flight and gradient of the performance index can pose problem, as we have to derivate an integral.

$$\nabla_{\mathbf{x}} \xi = \begin{bmatrix} \xi_{i=1..n} \\ \xi_r \\ \xi_\theta \\ \xi_T \end{bmatrix} = \begin{bmatrix} nc & \frac{\partial \xi_1}{\partial \mathbf{x}_1} & \frac{\partial \xi_1}{\partial \mathbf{x}_2} & nc & ... & nc & nc \\ nc & nc & \frac{\partial \xi_2}{\partial \mathbf{x}_2} & \frac{\xi_2}{\partial \mathbf{x}_3} & ... & nc & nc \\ nc & & & & ... & nc & nc \\ nc & nc & nc & nc & ... & \frac{\partial \xi_{n-1}}{\partial \mathbf{x}_{n-1}} & \frac{\partial \xi_{n-1}}{\partial \mathbf{x}_n} \\ nc & nc & nc & nc & ... & nc & \frac{\partial \xi_r}{\partial \mathbf{x}_n} \\ nc & nc & nc & nc & ... & nc & nc \\ 0 & \frac{\partial \xi_\theta}{\partial \mathbf{x}_1} & \frac{\partial \xi_\theta}{\partial \mathbf{x}_2} & \frac{\partial \xi_\theta}{\partial \mathbf{x}_3} & ... & \frac{\partial \xi_\theta}{\partial \mathbf{x}_{n-1}} & \frac{\partial \xi_\theta}{\partial \mathbf{x}_n} \end{bmatrix} \tag{3.45}$$

With:

$$\mathbf{x}_i = [k_i, \phi_i, \theta_i] \quad \text{for all} \quad i \in T$$

The SQP solver SNOPT [GMS08] has been used to solve the NLP minimisation problem. SNOPT, through the constraint infeasibility procedure, allows to safely evaluate the functions provided linear constraints are given. In addition, it converges quite efficiently because of an augmented Lagrangian approach.

3.5.4 Global Search for the Coast Segments

Completely solving problem 3.29 for $\mathbf{z}$ usually results in poor solutions. We were never sure a better solution might exist for the case considered.

To get the a-priori best trajectory, the global search space consists in the set of coast arcs sweep angle variables $\bar{\theta}_j, j \in C$. Considering m coast arcs, and k

regularly spaced point in $[0, \bar{\theta}]$, we construct a $m \times k$-grid for $\bar{\theta}_j$. For each point of the grid, the local solver seeks the optimal parameters of the thrusting legs. One can decide to stop the algorithm with the first solution, or decide to look down the $m \times k$-grid to find the best optimum. When constructing the grid, it is important to consider the case $\bar{\theta}_j = 0$ as it indicates whether we introduced too many switching points. The upper bound of the search grid can be reduced to $[0, 2\pi]$ for multiple revolution transfers.

Another option investigated is to use promising heuristic algorithm such as the Particle Swarm Optimisation algorithm (see section 2.3.2).

3.6 Conclusions

A continuous thrust model has been presented and studied. Coasting arcs have been inserted in the trajectory to produce Thrust - Coast sequence trajectories. Switching conditions have then been derived. We demonstrate that we need a very small number of parameters to describe the multi thrust - coast sequence trajectory. An optimisation problem for the rendezvous case has been introduced. An optimisation process has been designed to account for multi-gravity assist trajectories where interplanetary legs are not necessarily uniquely fixed by the dates of planet encounters. The model parameters allow different thrust profile regardless of the time of flight or the planet position, as would be the case with Lambert's problem arc or other continuous thrust models.

The optimisation algorithm used a SQP solver, and proved to be very robust and fast. The sequences can be easily evaluated.

The control provided by the model respects the dynamics. It can be used as an initial guess for a low-thrust trajectory problem considering fewer restrictions (direct method solver with low-thrust dynamics, or see chapter 5).

Among the perspectives are the choice of the number of switching points according to an optimality criterion or an implicit switching function. To model 3-D trajectories, inclination corrections and out of plane motion can be performed at each coast - thrust junction with a small impulsive manoeuvre.

Algorithm presented in 2.4 can be used to find LT-MGA trajectories. This is explained in the next examples sections.

Chapter 4

Applications

4.1 Earth - Mars rendezvous transfer

As a test we look for the best direct transfer from Earth to Mars, with rendezvous terminal conditions. We restrict the transfer to less than a revolution, with only 2 switching points or 2 thrust arcs.

Variable	Lower	Upper	Step
Departure (MM/YYYY)	01/2001	01/2010	20
T EM (days)	150	400	20
ΔV_f (km/s)	0	2.5	-
Switch (-)	2	2	-

Table 4.1: Search space characteristics

As impulsive manoeuvre trajectories are more delicate to reproduce with low-thrust propulsion, we favour low cost impulsive manoeuvres. Thus, the solution has been selected according to the lowest rendezvous manoeuvre, and then the lowest impulse ΔV_{imp} [*]. Various other criteria can be used such as the lowest consumption, time of flight, or thrust level.

It as been observed that a harsh constraint on ΔV_0 favours a full thrusting solution.

As expected, the solutions found differ according to the constraint on ΔV_f. It is not sure our optimisation strategy is the best when using the solution as initialisation for an indirect method program. We should keep track of all the solutions found.

[*] $\Delta V_{imp} = \Delta V_0 + \Delta V_f$

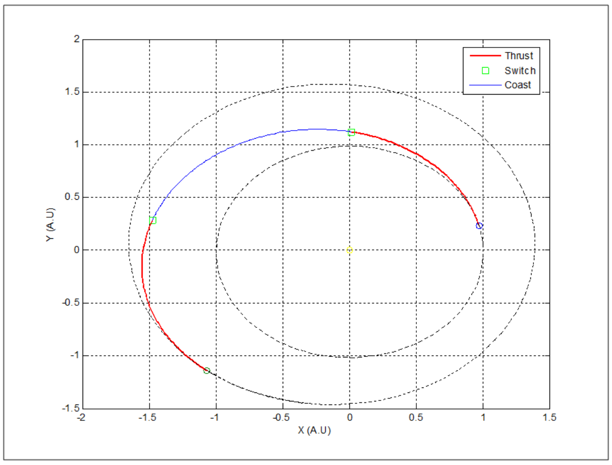

Figure 4.1: Earth - Mars transfer trajectory

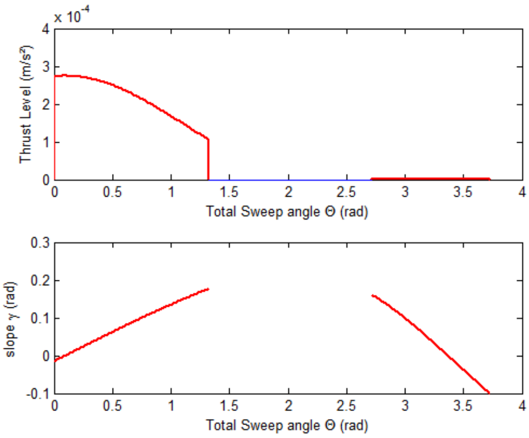

Figure 4.2: Earth - Mars transfer control

Table 4.2 displays three different solutions. Solution 2 represents the trajectory with the lowest rendezvous ΔV, while the trajectory with the lowest overall ΔV is given by solution 3.

Other solutions might exist for a different number of switchings. Figures 4.3 and 4.2 display solution 1 trajectory and control.

The maximum amplitude of the control gives information for the thruster design. According to figure 4.2, thrusters of 0.3N of thrust and an Isp of 2500s for an initial mass of 1000kg permit to follow this trajectory.

However, it is not clear how an indirect solver would deliver the trajectory. We must favour solutions with a low-thrust level variation during burning; otherwise we should expect a different number of burn arcs, or shorter burning time for the

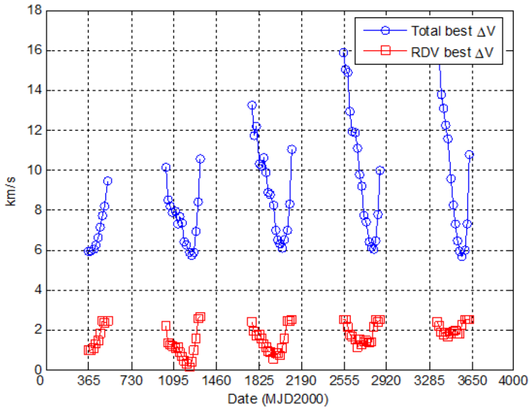

Figure 4.3: Earth - Mars transfer cost map

	Sol. 1	Sol. 2	Sol.3
T0 (dd/mm/20yy)	09/06/05	11/05/03	06/10/09
T EM (days)	390	370	330
ΔV_0 (km/s)	2.93	2.63	2.64
$\Delta V_{on-going}$ (km/s)	2.74	3.18	2.38
ΔV_F (km/s)	0.82	0.141	0.649
ΔV_{imp} (km/s)	3.75	2.77	3.29
$\Sigma \Delta V$ (km/s)	6.49	5.95	5.67

Table 4.2: Solutions

thrusters.

4.2 Earth - Venus - Mercury transfer

The following example is a low-thrust multi-gravity assist problem. This example has been selected for its difficulty toward the rendezvous manoeuvre. Because of the high speed of planet Mercury, the final rendezvous manoeuvre is usually important. A method to reduce it, is to allow multiple gravity-assists as in the MESSENGER mission, or BepiColombo with an EVVYY planet sequence (Earth - 2 Venus - 2 mercurY). However, such a multiple gravity assist scenario will not be considered here. Rather we seek the EVY trajectory (Earth Venus mercurY) allowing the lowest rendezvous manoeuvre. As a preliminary design tool we are mostly interested in a launch window, mission duration and overall consumption.

The algorithm seeks the minimum overall V trajectory. The solution obtained can then be optimised with a tool using an indirect formulation.

To reduce the breaking manoeuvre cost and the maximum thrust acceleration,

	Upper	Step	
Departure (Year)	2002	2003	10 days
T EV (days)	150	250	10
T VY (days)	400	700	10
(km/s)	0	5	-
(km/s)	0	5	-
V V (km/s)	5	10	0.5

Table 4.3: Search space characteristics

it is important in low-thrust propulsion to allow multiple switching times and multiple revolution transfer. The "good" number of revolutions depends on the time of flight and the mean orbital period between the planets. Along with this strategy, 2 switching points are used per revolution or 1 coast period per revolution. Indeed, a long time of flight for a short transfer distance tends to increase consumption. But long transfer distances might tend to increase coast phases. In our case, considering an average thrust profile, this can be seen from equations 3.9, 3.14 and the averaged term:

$$(\frac{d\theta}{dt})^{-1} \approx \frac{T}{\theta}$$

Thus, for the VY phase, the estimated number of full revolutions varies from 2 to 5.

Let's formulate the problem for an use with the algorithm of section 2.4. For the rendezvous problem we have the 2 minimisation sub-problems:

$$J_{EV} = \Delta V_0 + \Delta V_{lowthrust}^{EV} \tag{4.1}$$

$$\psi_{EV}(t_i) = V_\infty(t_i) - V_\infty \tag{4.2}$$

and:

$$J_{VY} = \Delta V_{lowthrust}^{VY} + \Delta V_f \tag{4.3}$$

$$\psi_{VY}(t_i) = V_\infty(t_i) - V_\infty \tag{4.4}$$

with ψ_{EV} and ψ_{VY} defining respectively the terminal and initial conditions, while V_∞ is a linking variables. ΔV_0 and ΔV_f define respectively the launch manoeuvre and the braking manoeuvre. $\Delta V_{lowthrust}$ defines the low-thrust consumption on the segment.

Because the minimal solution for the global problem is not generally the solution minimizing the 2 sub-problems separately, variable V_∞ evolves in a range on values. We have then many legs reaching Venus for different dates t_i and different hyperbolic excess velocities V_∞. These converging solutions generate a new sub space of initial date for the second phase VY.

	Model	Sauer[Sau97]
T_0 (dd/mm/20yy)	31/07/02	27/08/02
T_F (dd/mm/20yy)	11/04/04	22/12/04
T EV (days)	200	185
T VY (days)	400	663
VY revolution	2.9	5.7
Launch V_∞ (km/s)	1.98	2.31
Final V_∞ (km/s)	4.9	0.00
pmf	0.34	0.275
Swing-by V_∞ (km/s)	6.50	5.58
Swing-by rad. (km)	6764	-

Table 4.4: modelled and optimal solutions

The algorithm found 952 converging solutions for the EV phase. As some solutions for the EV phase have the same date of encounter with Venus and the same final V_∞, the points to start from in the VY phase are reduce. This efficiently reduces computational time. Thus, we have 185 initial points $\{t_i, V_\infty\}$ for the phase VY. We then find 360 converging solutions for the VY phase. With all possible combinations, these lead to more than 1400 complete EVY solutions, feasible or infeasible.

Among the possible solutions, we remove those having maximum thrust acceleration above 1 mm/s^2. Because this is a wide margin constraint, which only removes unreasonable trajectories, we have a constraint on the average acceleration. This constraint is given by the thrusters' capabilities. The thrusters considered have a 3kW power unit and provide a maximum acceleration of 0.3 mm/s^2. The VY leg is the most critical for this constraint. Most of the low number of revolution legs have been pruned out, beside the low braking manoeuvre cost ΔV_f , as low as 0.5km/s.

The complete solution has then been selected according to the lowest characteristic velocity (sum of ΔV_0 and ΔV_f) and the lowest consumption. It is very close to the one published by Sauer [Sau97] for the same thrusters performance. The solution of [Sau97] presents 5 coast periods (10 switching times) on the VY phase. However, our case needs a higher launch energy, and in addition the out of plane correction is only done during the swing by.

The solution of Sauer [Sau97] presents 6 coast periods (12 switching times) on the VY phase, but the thrusters used provide an acceleration of 0.3 mm/s^2.

4.3 Earth - Mars - Vesta - Ceres

The Earth to Vesta and Ceres mission (DAWN mission [RFRR04]) is the most recent low thrust mission. It is also the first mission to visit both an asteroid and

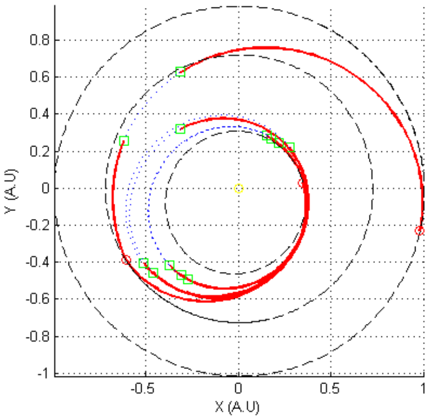

Figure 4.4: Earth - Venus - Mercury transfer trajectory

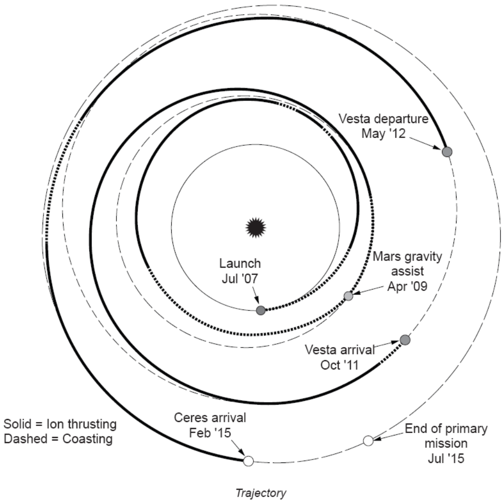

Figure 4.5: DAWN original trajectory

a dwarf planet.

The spacecraft is first targeting asteroid Vesta, after a Mars gravity assist. After nearly 7 months in orbit around Vesta, for science experiments, the spacecraft will target the dwarf planet Ceres.

We allow a maximum of 1 complete revolution per phase, as in the original mission. As before we allow 2 switching times per revolution. To simplify our approach we consider the orbiting time around Vesta variable. That way we can independently optimise the transfer from Earth to Vesta and from Vesta to Ceres. We only limit the orbiting time to be between 3 and 10 months.

In the original mission, time margins have been added during the transfers as

	Upper	Step	
Departure (Year)	2007	2008	10 days
T EM (days)	400	700	10
T M.Vesta (days)	800	1200	25
T Vesta.Ceres (days)	800	1200	25
V_0 (km/s)	0	5	-
V_f (km/s)	0	5	-
Mars V_∞ (km/s)	3	8	0.5

Table 4.5: Search space characteristics

T_{EARTH} (dd/mm/20yy)	25/06/07
T_{VESTA} (dd/mm/20yy)	28/08/11
T EM (days)	700
T M.Vesta (days)	825
C3 (km^2/s^2)	1.65
Vesta V_∞ (km/s)	4.14
Swing-by V_∞ (km/s)	6.50
pmf	0.38
Max Acc (mm/s^2)	0.17

Table 4.6: Earth - Mars - Vesta trajectory

safety measures and to prevent any thrust issues. This has not been taken into account here.

These results are in line with the original mission scenario. However the final V_∞ at Vesta is high, and more switching times should be added on the Mars to Vesta leg to reduce it.

The automated approach permits to identify 2 opportunity windows for the Vesta to Ceres transfer: Nov 2011 and Aug - Nov 2012. Here, the first window has been selected because of the good opportunities for the Earth to Vesta transfer. Thus, the orbiting time found around Vesta is about 4 months.

4.4 GTOC3 problem

For the third edition of GTOC (section 3.1.2) we used the low-thrust model described above, and the pruning strategy defined in chapter 2.4. In a systematic approach, we thus computed:

- E - * : all the legs coming from Earth to each of the asteroids on the list, with:
    - launch date ranging from 2016 to 2020

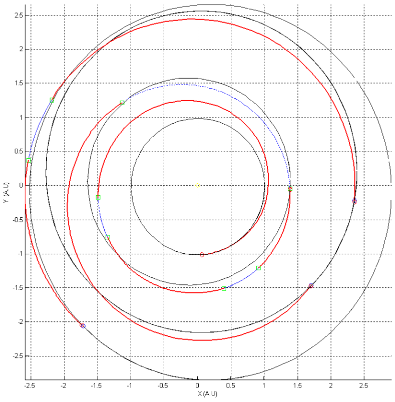

Figure 4.6: Earth - Mars - Vesta - Ceres transfer trajectory

  - initial hyperbolic velocity not exceeding 0.5km/s
  - time of flight comprised between 0.5 to 2 years
  - objective function defined as a combination of the consumption and the initial launch velocity.

- * - * : all asteroid-to-asteroid legs, with:
  - starting date ranging from 2016 to 2034
  - initial and final hyperbolic velocity minimised to zero
  - time of flight ranging from 0.5 to 2 years
  - objective function defined as the consumption

- * - E : all the leg coming from an asteroid to reach planet Earth, with:
  - starting date ranging from 2016 to 2034
  - initial hyperbolic velocity minimised to zero
  - final hyperbolic velocity minimised to zero
  - time of flight ranging from 0.5 to 2 years
  - objective function defined as a combination of the consumption and the breaking manoeuvre ΔV

Besides these search conditions, the maximum acceleration was added to the model.

This approach might seem quite expensive with probably non useful calculations. However, once we computed and recorded all the body to body transfers,

Data base elements	
Total number of legs found AST-AST	3380
Total number of legs found AST-EARTH	3447
Total number of legs found EARTH-AST	1457
Building constraints	
Maximum Mean acceleration	0.000075 m/s^2
Maximum Abs. acceleration	0.001000 m/s^2
Thruster Power:	2206.35 W
Maximum dV1	3000.0 m/s
Maximum dV2	3000.0 m/s
Number of points removed	5024 / 10274
Phase: Construction of the tree	
Phase 2	2079 nodes
Phase 3	14892 nodes
Phase 4	25046 nodes
Number of solutions found	25046

Table 4.7: Statistics of the algorithm

the construction of a list of solutions requires only post processing of the data recorded with a dynamic programming algorithm. We can easily list the possible solutions for different constraints or conditions without much calculation.

Tables 4.7 and 4.4 list the first best solutions. Some sequences appear many time as they account for different date or time of flight. During the competition, we discarded a-priori non interesting asteroids to reduce the data and computation time.

In addition, we can also impose a swing-by of earth for the i^{th} item of the sequence without much calculation. Also, we obviously note that different constraints produce different scenarii.

The only difficulty with the approach is the somewhat cumbersome data to handle. This is however easily handled on today's computer environments.

Inserting an Earth swing-by to the third solution provides the 5th GTOC3 solution. It is important to list and test as many sequences as possible and not only test the apriori best solution delivered by the model.

Figure 4.4 shows a trajectory example. No swing-bys have been considered on this example.

The sequence we found was: E-96-E-88-49-E

As the results of the competition show, to get the maximum value of the objective function, we should have had more Earth swing-bys.

seq. nb	sequence
1	3 86 98 59 3
2	3 86 98 59 3
3	3 98 106 59 3
4	3 86 98 59 3
5	3 86 98 59 3
6	3 86 98 59 3
7	3 98 106 59 3
8	3 86 98 59 3
9	3 86 98 59 3
10	3 98 106 59 3
11	3 86 98 59 3
12	3 86 98 59 3
13	3 86 98 59 3
14	3 86 98 59 3
15	3 98 106 59 3
...	

Table 4.8: List of Different Scenarii

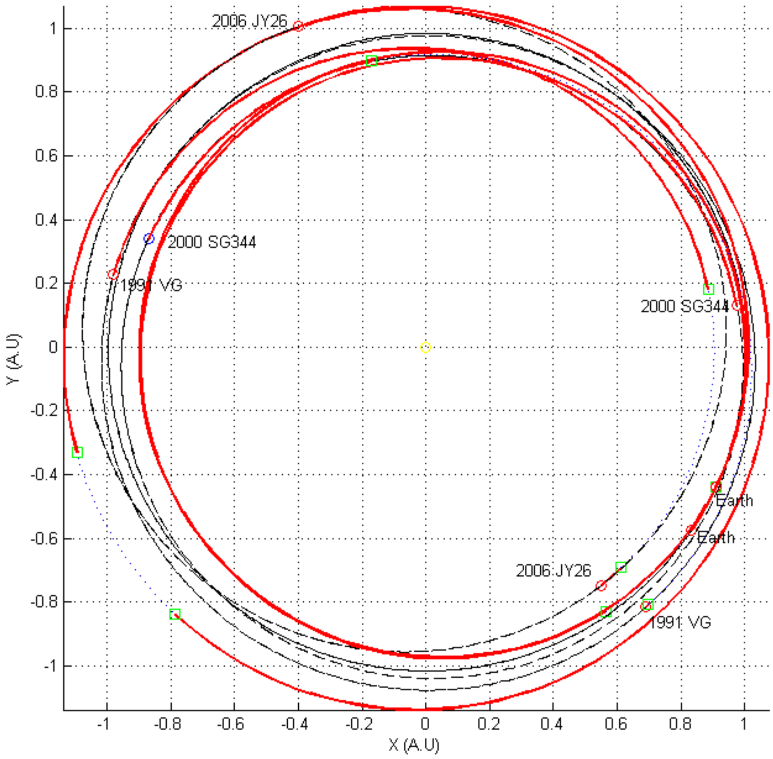

Figure 4.7: Trajectory for GTOC3 without Earth swing-bys

Part II

Optimal Control Methods

Presentation of Part II

This chapter, and the following, are the second part of the research. They present a completely different approach from that followed in previous chapters. From now on, we focus on local optimisation procedures to find locally optimal swing-by scenarii. This is different from the first part, as here we determine the scenario only a-posteriori, as the result of a post-optimisation analysis.

The solution method presented in chapters 2, 3 is great in the general case when we are looking for a sequence of particular bodies such as comets and asteroids which do not necessarily present gravitational properties and we are not looking for gravity assists. When selected bodies have a strong and measurable gravitational field such as planets, other optimisation methods taking advantage of these dynamics can be used.

In this chapter 5 we review and examine local optimisation methods and present their limitations for the problem considered. Next chapter 6 will present another local optimisation method which could be the basis of new solution methods for the problem.

Chapter 5

Review of Optimal Control Methods applied to Low-Thrust Interplanetary Trajectories

5.1 General Problem formulation

5.1.1 Problem Description

Optimal Control Problem Consider the problem of transferring a spacecraft from an initial state (X_0, t_0) to a final state (X_f, t_f) in a given time of flight $T = t_f - t_0 > 0$. The state of the spacecraft is composed of its position $\mathbf{R}(t) \in \mathbb{R}^3$, its velocity $\mathbf{V}(t) \in \mathbb{R}^3$ and its mass $m(t) \in \mathbb{R}$. The control $\mathbf{u}(t) \in \mathbb{R}^3$ provides the thrust and steering of the spacecraft. We are concerned with the minimisation of the quantity of propellant required (thus maximising the final mass).

The propulsion of the spacecraft is done through an electric propulsion system, for which the Specific Impulse is noted I_{Sp} and the Thrust Force is noted F_{Th}. Then, the spacecraft is under the influence of the Sun gravitational force field and the gravity accelerations of the planets. The dynamics are formulated with the

Cartesian model:

$$\begin{aligned}\mathbf{f}(\mathbf{x},\mathbf{u};t) &= \frac{d}{dt}\begin{bmatrix}\mathbf{R}\\ \mathbf{V}\\ m\end{bmatrix}\\ &= \begin{bmatrix}\mathbf{V}\\ -\mu_{SUN}\frac{\mathbf{R}}{\|\mathbf{R}\|^3} - \sum_{i=1}^{9}\mu_i\left(\frac{\mathbf{R}-\mathbf{R}_i}{\|\mathbf{R}-\mathbf{R_i}\|^3}+\frac{\mathbf{R}_i}{\|\mathbf{R_i}\|^3}\right)+F_{Th}\mathbf{u}\\ -\frac{F_{Th}}{g_0 I_{Sp}}\|\mathbf{u}\|\end{bmatrix}\end{aligned} \tag{5.1}$$

with the initial conditions, defined at initial time t_0:

$$X(t_0) = X_0 \tag{5.2}$$

This model has been chosen because of its simplicity for handling gravitational perturbations.

We note $n_x = 7$ the dimension of the state vector.

The final time t_f equality constraints $\psi : \mathbb{R}^{n_x} \to \mathbb{R}^{n_k}$, $n_k \in \mathbb{N}$, are:

$$\psi(X(t_f); t_f) = 0 \tag{5.3}$$

, We admit that the Jacobian $\psi_x(x)$ has full rank.

We have an additional constraint for the limited thrust profile:

$$\|\mathbf{u(t)}\| \leq 1 \tag{5.4}$$

We note $n_u = 3$ the dimension of the control vector. We define U as the admissible set for the control $\mathbf{u}$.

The cost functional $J : \mathbb{R}^{n_x} \times U \to \mathbb{R}$ to minimise, under the Bolza form[Bol33], is:

$$\begin{aligned}J(\mathbf{x},\mathbf{u}) &= \phi(x(t_f)) + \int_{t_0}^{t_f} L(\mathbf{x},\mathbf{u};t)dt\\ &= -m(t_f)\end{aligned} \tag{5.5}$$

This represents the maximum final mass problem. The cost can be indifferently formulated either as an integral cost ($\phi = 0$), or a terminal cost ($L = 0$). We here choose the terminal cost formulation ($L = 0$).

We suppose f is smooth on $\mathbb{R}^{n_x *} \times U$. We suppose functions ϕ and L are continuously differentiable with respect to their arguments.

For 2-body and 3-body problems, the solution can be found analytically in some cases, and are essentially well known. Of course, for an N-body interplanetary transfer ($N > 3$) problem, we are hardly simultaneously under the influence of more than 3 bodies. However, in comparison with the 3-body problem (and its approximations RTBP, CRTBP), we cannot work in a restricted frame, and want we be as general as possible.

5.1.2 State of the art

Some attempts to solve the general multi-body dynamical transfer problem have already been made.

An interesting problem is the problem of escape and capture. Transferring a spacecraft from one planet to another, considering their respective gravity field is challenging. The spiralling movement around the planets, followed by an interplanetary phase, makes optimisation difficult. Most attempts [RO08, NVB01, VN00] consider a patched conic approximation. The trajectory is divided into segments. Other approximations consider curve fits[Klu02], averaging techniques[Gao07], or low-thrust models for the spiralling capture phase. This escape and capture problem will be considered in this study as our benchmark problem.

In [JCGK01] was performed a gravitational capture at Mercury. Using the Sun gravity, weak ballistic capture conditions at Mercury were derived.

A special case of trajectory optimisation in multi-body dynamics concerns the case of R3BP. The R3BP provides the means to study the capture phenomenon[KP95, Gay88]. In [Rus07], the primer vector theory is used, within a global search procedure, for an Europa DRO transfer and an Earth - Moon transfer.

In [Whi01, WS02] the author optimises the trajectory in the multi-body dynamics. According to these articles, the algorithm is capable of automatically finding gravity assists. However, finding an effective swing-by is not a simple task as an appropriate initial guess is needed.

There are basically two formulations that can be used to solve the optimal control problem (OCP) 5.1, 5.4, 5.5 [Bet98]: direct methods and indirect methods.

5.2 Direct Problem Formulation

We said we have a direct method when the change in the cost function is directly related to the change in the control or the decision vector.

5.2.1 Formulation

Suppose we want to minimise the objective function given by eq. 5.5, under the dynamical constraints given by eq. 5.1. Consider also the dynamical constraint on $\mathbf{u}$, and the terminal state constraints $\psi(\mathbf{x}(t_f), \mathbf{x}_f)$ given by eq. 5.3, with initial condition $\mathbf{x}(t_0) = \mathbf{x}_0$.

The time is split into N sub-intervals. The instants t_i are defined at each mesh point: $t_0 < t_1 < ... < t_N$.

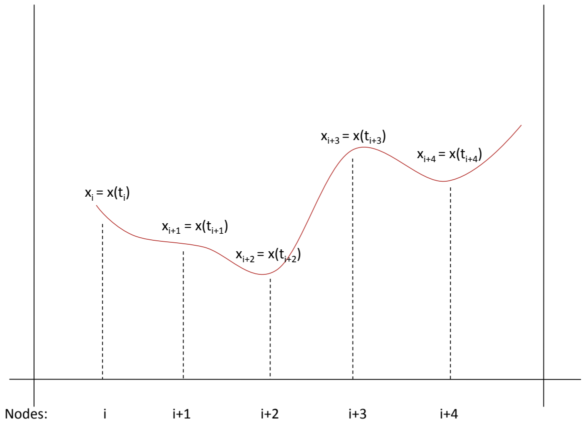

Figure 5.1: Mesh description.

We note:

$$\mathbf{x}_i = \mathbf{x}(t_i)$$
$$\mathbf{u}_i = \mathbf{u}(t_i)$$

The decision vector is thus composed of state variables $\mathbf{x}_i$ and control variables $\mathbf{u}_i$ at different dates $t_i \in [t_0, t_f]$.

The optimal control problem (OCP) is turned into a parameter optimisation problem (POP) or NLP problem (transcription). The values of the state and the control at a mesh point are the NLP variables.

We have then to solve defect equations[TC95] at each mesh point $i = 1, ..., N$. And:

$$\xi_i = \mathbf{x}_{i+1} - \mathbf{x}_i - \int_{t_i}^{t_{i+1}} \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i; t)dt \tag{5.6}$$

Because the state is part of the decision vector, the integral must be approximated to account for the decision vector (figure 5.2). The basis for transcription methods is to replace the ODEs with NLP variables and defect conditions[Bet98].

Clearly, one could have suggested to use only the control $\mathbf{u}_i$ as NLP variables[Hul97]. However this a-priori interesting approach conceals the problem of computing derivatives. In the most general case, these derivatives can only be computed by finite differences, thus requiring high precision integration of the dynamics, and problems of convergence.

The integral is thus approximated by a quadrature, such as:

- Euler, Trapeze collocation [BC95]

- Runge-Kutta, Simpson transcription [EC92, SC94]

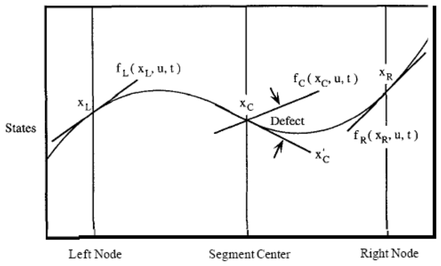

Figure 5.2: Transcription. The points x_L, x_C and x_R must be adjusted to nullify the defect condition on the slope at the center.[TC95]

- Gauss-Lobatto (Pseudo Spectral methods) [FR02, Hun07]

The higher the order of the approximation, the better the accuracy[HC96, FR02]. Usually we assign the term collocation to the Simpson rule, and direct transcription to any implicit integration rule[Hul03].

The general formulation is:

$$\xi_i = \mathbf{x}_{i+1} - \mathbf{x}_i - \sum_{j=1}^{n} \omega_j \mathbf{f}_j \tag{5.7}$$

with:

$$\mathbf{f}_j = \mathbf{f}(\mathbf{x}, \mathbf{u}; t_j) \quad t_j \in [t_i, t_{i+1}]$$

where the constants $\omega_i \in \mathbb{R}$ are given by the integration rule scheme.

For example in the trapeze transcription:

$$\xi_i = \mathbf{x}_{i+1} - \mathbf{x}_i - \frac{h_i}{2}(\mathbf{f}(\mathbf{x}_{i+1}, \mathbf{u}_{i+1}; t_{i+1}) - \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i; t_i)) \tag{5.8}$$

The decision vector can also include midpoint values for the state $\mathbf{x}_m$ and the control $\mathbf{u}_m$. Typically pseudo spectral methods use the root of a particular polynomial to define the mid-points over each sub-interval $[t_i, t_{i+1}]$. One tremendous advantage of pseudo-spectral methods is the possibility to link the NLP multipliers to the OCP costates[Hun07, BHTR06, FR01].

To reduce the parameter space dimension, we can remove the control description from the decision vector. The differential inclusion method consists in eliminating the control from the dynamics. We are replacing the usual ODEs seeking for a control, by ODEs providing the attainable set for the control[Sey94]. This method has been applied successfully for space trajectory problems [CCW94].

Recent tools taking advantage of the sparse structure of the Jacobian can solve this problem quite efficiently (SNOPT, IPOPT [WB06]).

5.2.2 Limitations

The main advantages of direct methods, for the problem considered, are that:

- they do not need an explicit formulation of the necessary conditions of optimality.

- they are not very sensitive, because the control history is guessed. We do not have to use the Euler-Lagrange equations.

This approach has however 2 major flaws: the satisfaction of conditions of optimality, and the accuracy of the solution.

Inadapted Meshing

Direct transcription methods are initially seen as approximation methods. The meshing should be tailored to follow the dynamics as closely as possible. In this case, we can either have a very high order of interpolation, or an adaptive mesh.

Depending on the transcription, some algorithms[GFR08] adapt the mesh during the iterations. Laurent-Varin[LVBBT07] uses the Runge-Kutta scheme, and adapts the mesh formulating an optimal refinement problem. Indeed, he adds mesh points, and tries to minimise the number of subdivision on each interval to respect a maximal local error threshold. Other algorithms[BH98], and codes (SOCS [Bet], RIOTS95[SPC97], DIDO[Ros03]), update the mesh only after a solution has been found, to start the solver again. Often the reference for mesh update is the knowledge of the geometrical structure of the control.

Mostly, these approaches follow only the order of smoothness of the solution. Although these approaches reduce the error in some metric from the continuous solution, the meshing is not determined by an optimal policy.

Lack of Optimality

In addition, the optimality equations are never integrated, and the accuracy of the dual problem might never be met. We may be concerned by the optimality of the control, and expect only a sub-optimal control[Hul03]. The control is often only first order accurate. We can expect a second order $O(h^2)$ approximation of the optimal continuous solution under conditions of coercivity and smoothness of the solution, depending on the quadrature rule[Eng06, DHV01]. However, due to the bang-bang structure of the optimal control, these conditions do not hold. Many published results present a control amplitude which is hardly bang-bang.

It is then not sure whether direct transcription is reliable enough for handling swing-byes, and taking advantage of the swing-by dynamics (equation 5.1). This point should however be explored more closely.

5.3 Indirect Problem Formulation

For indirect methods, the control is implicitly given by variational equations. It is the solution of an ODE (Euler-Lagrange equation). The only unknowns of the problems are some initial and final conditions for the ODE.

5.3.1 Examples of Low-Thrust Trajectory optimisation

In [Rus07], the author performs a global trade-off study. He does not immediately consider the optimal control structure, but supposes the control is bang-bang on the Pareto front. Thus, he does not need to solve the Two Point Boundary Value Problem (TPBVP) for the costate. Then a random initialisation of the initial costate variables permits to find solutions that satisfy the transversality conditions.

In [Tad04], a solution for the VSI (Variable Specific Impulse) problem is used. This solution admits an unbounded control. It is used to estimate initial costate variables for the original CSI problem (Constant Specific Impulse). Thus, an iterative process (using a Powell method) permits to find the optimal control. The switching function is used, with a forward integration of the state equations to satisfy the terminal state constraints.

In [BE02], the author uses continuation and smoothing methods to find the optimal bang-bang structure of the control. In [Ber01], the author uses a decomposition technique to tackle the low-thrust multi-gravity-assist trajectory problem. He does not integrate the dynamical equations during the swing-by. His approach is probably state-of-the-art for indirect algorithms considering intermediate swing-bys (GTOC 3). However, beside the requirement to know the planet sequence apriori, the trajectory is not reliable close to planets, because of the patched conic approximation.

Providing a planet sequence only gives an optimal solution for the problem with these specific swing-by constraints. A better solution might exist, which considers different planets. This is the focus of next chapter 6.

5.3.2 The Maximum Principle

Minimisation problem

Consider the objective function given by equation 5.5, and the problem description of section 5.1. The Lagrangian (or augmented cost functional) $\mathcal{L} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \times U \rightarrow \mathbb{R}$ is the cost functional augmented with the state dynamical constraint and the terminal constraints assigned with Lagrange multipliers $\nu \in \mathbb{R}^{nc}$:

$$\mathcal{L} = \phi(\mathbf{x}(t_f)) + \int_{t_0}^{t_f} \lambda^T(\mathbf{f}(\mathbf{x}, \mathbf{u}; t) - \frac{d\mathbf{x}}{dt})dt + \nu^T \psi(\mathbf{x}(t_f); t_f) \tag{5.9}$$

The costate vector $\lambda(t) \in \mathbf{R}^{n_x}$ play the role of dynamical Lagrange multipliers assigned to the dynamics.

We use the Pontryaguin Maximum Principle to find an optimal control. Pontryaguin Maximum Principle [Pon53] refers to optimisation around a specific trajectory. Note that we use the terminal cost formulation ($L = 0$), while the original formulation of the Maximum Principle uses an integral cost. Had we wanted to use the integral formulation, we would have added an additional state equation and an additional costate variable λ_0.

Theorem 5.3.1 (Pontryaguin Principle)**.** *[Pon53] Let $\mathbf{u}(t)$, $t_0 \leq t \leq t_f$ be an admissible control such that the corresponding trajectory $\mathbf{x}(t)$ which begins at the point $\mathbf{x}_0$ at the time t_0 passes, at some time t_f through a point of line Π. In order that $\mathbf{u}(t)$ and $\mathbf{x}(t)$ be optimal it is necessary that there exist a nonzero continuous function $\lambda(t)$ corresponding to $\mathbf{u}(t)$ and $\mathbf{x}(t)$, such that:*

1. *$\forall t \in [t_0, t_f]$, the function $H(\mathbf{x}(t), \lambda(t), \mathbf{u}) = \lambda^T \mathbf{f}(\mathbf{x}, \mathbf{u})$ of the variable $\mathbf{u} \in U$ attains its maximum at the point $\mathbf{u} = \mathbf{u}(t)$*

$$H(\mathbf{x}(t), \lambda(t), \mathbf{u}(t)) = \sup_{\mathbf{u} \in U} H(\mathbf{x}(t), \lambda(t), \mathbf{u})$$

2. *at the terminal time t_f the relations*

$$\lambda(t_f) \leq 0$$
$$\sup_{\mathbf{u} \in U} H(\mathbf{x}(t_f), \lambda(t_f), \mathbf{u}) = 0$$

are satisfied.

Furthermore, it turns out that if $\lambda(t)$, $\mathbf{x}(t)$ and $\mathbf{u}(t)$ satisfy the Euler-Lagrange equations and the first condition, the second condition can be verified at any time $t \in [t_0, t_f]$.

The function $H(\mathbf{x}(t), \lambda(t), \mathbf{u})$ is usually called the Hamiltonian.

We call *extremal* a solution $(\mathbf{x}, \lambda, \mathbf{u})$ of the Euler-Lagrange equations.

Differentiating the Lagrangian $\mathcal{L}$ provides the necessary conditions of optimality, such as the Euler-Lagrange equations and the transversality conditions. A second order check is often necessary to confirm that the solution is a minimum.

As $L = 0$ (defined in equation 5.5), we do not use λ_0, and the Hamiltonian is simply:

$$H(\mathbf{x}, \lambda, \mathbf{u}; t) = \lambda^T \mathbf{f}(\mathbf{x}, \mathbf{u}; t) \tag{5.10}$$

The Euler-Lagrange equations give states and costates dynamical ODEs:

$$\frac{d\mathbf{x}^T}{dt} = \frac{\partial H(\mathbf{x}, \lambda, \mathbf{u}; t)}{\partial \lambda} \tag{5.11}$$

$$\frac{d\lambda^T}{dt} = -\frac{\partial H(\mathbf{x}, \lambda, \mathbf{u}; t)}{\partial \mathbf{x}} \tag{5.12}$$

We shall note $\mathbf{F} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \times U \to \mathbb{R}^{2n_x}$ the problem dynamics:

$$\mathbf{F}(\mathbf{x}, \lambda, \mathbf{u}; t) = \frac{d}{dt}\begin{bmatrix} \mathbf{x} \\ \lambda \end{bmatrix} \tag{5.13}$$

The optimal control $\mathbf{u}^*$ is given by the minimisation problem:

$$\mathbf{u}^* = \arg \min_{\|\mathbf{u}\| \leq \mathbf{1}} H(\mathbf{x}, \lambda, \mathbf{u}; t) \tag{5.14}$$

The control $\mathbf{u}^*$ is bang-bang[Ber01].

Transversality conditions

The terminal state constraints are transposed in the dual space to completely set the problem. As dates t_0 and t_f are fixed, we have no conditions on the Hamiltonian H.

Transversality conditions are conditions on the initial and final costate vector values. Noting T_xM_0 and T_xM_f the tangent hyperplane [Pon53][Tre07] to the initial and final space M_0 and M_f, such that:

$$M_0 = \{\mathbf{x} \quad s.t. \quad \phi(x, t_0) = 0\} \tag{5.15}$$

$$M_f = \{\mathbf{x} \quad s.t. \quad \psi(x, t_f) = 0\} \tag{5.16}$$

$$T_xM = \{\mathbf{x}, < \mathbf{x}, \nabla X > \geq 0\} \tag{5.17}$$

We have:

$$\lambda(t_0) \perp T_xM_0 \tag{5.18}$$

$$\lambda(t_f) - \frac{\partial \phi}{\partial x}(t_f) \perp T_xM_f \tag{5.19}$$

These equations explain the term *Transversality conditions*. With $\phi = 0$, and the linear dependancy of λ_0 and λ_f on M_0 and M_f, we write:

$$\lambda(t_0) \in \mathbb{R}^{n_x} \tag{5.20}$$

$$\lambda(t_f) = \frac{\partial \phi}{\partial \mathbf{x}_f} + \nu^T \frac{\partial \psi}{\partial \mathbf{x}_f} \tag{5.21}$$

After simplifications, in accordance with the Lagrangian differential, we define the constraint on the costate vector at terminal time:

$$\tau(\mathbf{x}, \lambda, \nu) = \frac{\partial m(t_f)}{\partial \mathbf{x}_f} + \nu^T \frac{\partial \psi}{\partial \mathbf{x}_f} - \lambda(t_f) \tag{5.22}$$

The TPBVP can be solved efficiently using a simple or multiple shooting algorithm [Ber01]. The roots are found using, for example, a simple Newton-Raphson's method. In this case, the convergence can be highly improved by providing the Jacobian.

5.3.3 Numerical derivatives

The costate vector gives the sensitivity of the cost wrt the current state vector. Up to first order, a deviation of the current extremal state $\mathbf{x}^*(t)$ of amount $\delta\mathbf{x}(t)$ produces the change:

$$\delta J^*(\mathbf{x}^*, \mathbf{u}^*; t) = \lambda(t)^T \delta\mathbf{x}(t) \tag{5.23}$$

This simple equation will be helpful to understand the limitations of indirect methods. Equation 5.23 also provides a mean to evaluate the derivatives.

The Jacobian of the problem can be evaluated quite precisely, and can improve convergence while finite differences fail[ZO05]. We use a sensitivity matrix and construct a state/costate transition matrix[BH75, Bat01]. Denoting $\Phi(t, t_0)$ the $(2n_x) \times (2n_x)$ transition matrix, we have:

$$\frac{d\Phi(t,t_0)}{dt} = \frac{\partial\mathbf{F}}{\partial\mathbf{S}}\Phi(t,t_0) \tag{5.24}$$

$$\Phi(t_0,t_0) = I \tag{5.25}$$

Where: $\mathbf{S} = [\mathbf{x}^T, \lambda^T]^T$.

Using equation 5.1, and the Gravity Gradient Matrix $G(\mu, \mathbf{R})$ [Bat01], we have:

$$\frac{\partial\mathbf{F}}{\partial\mathbf{S}} = \begin{bmatrix} 0 & I_{3\times3} & 0 & 0 & 0 & 0 \\ G(\mu,\mathbf{R}) & 0 & -\frac{F_{Th}}{m^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{\partial G\lambda_V}{\partial\mathbf{R}} & 0 & 0 & 0 & -G(\mu,\mathbf{R}) & 0 \\ 0 & 0 & 0 & -I_{3\times3} & 0 & 0 \\ 0 & 0 & -\frac{2F_{Th}\lambda_m}{m^3} & 0 & 0 & \frac{F_{Th}}{m^2} \end{bmatrix} \tag{5.26}$$

The transition matrix can be written:

$$\frac{\partial\mathbf{S}}{\partial\mathbf{S}_0} = \begin{bmatrix} \Phi_{R,R} & \Phi_{V,R} & \Phi_{m,R} & \Phi_{\lambda_R,R} & \Phi_{\lambda_V,R} & \Phi_{\lambda_m,R} \\ \Phi_{R,V} & \Phi_{V,V} & \Phi_{m,V} & \Phi_{\lambda_R,V} & \Phi_{\lambda_V,V} & \Phi_{\lambda_m,V} \\ \Phi_{R,m} & \Phi_{V,m} & \Phi_{m,m} & \Phi_{\lambda_R,m} & \Phi_{\lambda_V,m} & \Phi_{\lambda_m,m} \\ \Phi_{R,\lambda_R} & \Phi_{V,\lambda_R} & \Phi_{m,\lambda_R} & \Phi_{\lambda_R,\lambda_R} & \Phi_{\lambda_V,\lambda_R} & \Phi_{\lambda_m,\lambda_R} \\ \Phi_{R,\lambda_V} & \Phi_{V,\lambda_V} & \Phi_{m,\lambda_V} & \Phi_{\lambda_R,\lambda_V} & \Phi_{\lambda_V,\lambda_V} & \Phi_{\lambda_m,\lambda_V} \\ \Phi_{R,\lambda_m} & \Phi_{V,\lambda_m} & \Phi_{m,\lambda_m} & \Phi_{\lambda_R,\lambda_m} & \Phi_{\lambda_V,\lambda_m} & \Phi_{\lambda_m,\lambda_m} \end{bmatrix}_{(t,t_0)} \tag{5.27}$$

Thus, to evaluate the Jacobian of the final constraint on $X(t_f)$ with respect to the initial costate vector $\lambda(t_0)$, we use:

$$\frac{\partial\mathbf{S}_f}{\partial\mathbf{S}_0} = \Phi(t_f, t_0) \tag{5.28}$$

Then for the constraints we only need the left lower part of Φ:

$$\frac{\partial \mathbf{x}_f}{\partial \lambda_0} = \begin{bmatrix} \Phi_{R,\lambda_R} & \Phi_{R,\lambda_V} & \Phi_{R,\lambda_m} \\ \Phi_{V,\lambda_R} & \Phi_{V,\lambda_V} & \Phi_{V,\lambda_m} \\ \Phi_{m,\lambda_R} & \Phi_{m,\lambda_V} & \Phi_{m,\lambda_m} \end{bmatrix}_{(t_f,t_0)} \tag{5.29}$$

$$\frac{\partial \psi}{\partial \lambda_0} = \frac{\partial \psi}{\partial \mathbf{x}_f}\frac{\partial \mathbf{x}_f}{\partial \lambda_0} \tag{5.30}$$

For the transversality conditions:

$$\frac{\partial \lambda_f}{\partial \lambda_0} = \begin{bmatrix} \Phi_{\lambda_R,\lambda_R} & \Phi_{\lambda_R,\lambda_V} & \Phi_{\lambda_R,\lambda_m} \\ \Phi_{\lambda_V,\lambda_R} & \Phi_{\lambda_V,\lambda_V} & \Phi_{\lambda_V,\lambda_m} \\ \Phi_{\lambda_m,\lambda_R} & \Phi_{\lambda_m,\lambda_V} & \Phi_{\lambda_m,\lambda_m} \end{bmatrix}_{(t_f,t_0)} \tag{5.31}$$

$$\frac{\partial \tau}{\partial \lambda_0} = \frac{\partial \tau}{\partial \mathbf{x}_f}\frac{\partial \mathbf{x}_f}{\partial \lambda_0} + \frac{\partial \tau}{\partial \lambda_f}\frac{\partial \lambda_f}{\partial \lambda_0} \tag{5.32}$$

Equations 5.30 and 5.32 provide the gradient for the OCP problem.

This approach has been successfully used in [Oly07a], using the Primer Vector Theory.

5.3.4 Limitations

Sensitivity at Swing-bys

The sensitivity of the method is due to the Euler-Lagrange equations. Numerically, we can observe this difficulty by observing the costates dynamic.

Figure 5.3 depicts the costate vector dynamics computed along the Voyager 2 trajectory. To obtain this costate vector history, we integrated the Euler Lagrange equations, for each unit initial solution vector, along the state trajectory. As the spacecraft does not thrust along this trajectory, the value of the initial costate vector is not important to observe the behaviour of the costate. We can indeed consider the trajectory optimal for the maximum mass problem.

The sensitivity outside and inside the swing-by planet SOI, to account for the difference of change in the dynamics, is compared. In equation 5.26, the term $G(\mu_p, \mathbf{R}-\mathbf{R}_p)$ becomes predominant in front of $G(\mu, \mathbf{R})$. There is also a difference in scale between G and $\frac{\partial G}{\partial R}$ of the order of $\|\mathbf{R} - \mathbf{R_p}\|^2$. The small duration of the swing-by makes the costate dynamics change rapidly of many orders of magnitude.

The states are rather steady. This is not the case for the costate variables. Because any final cost function is very sensitive to a swing-by condition, the value of the costate variables at the swing-by gets high values (see equation 5.23). Figure 5.3 outlines the numerical difficulty appearing when the dynamics change.

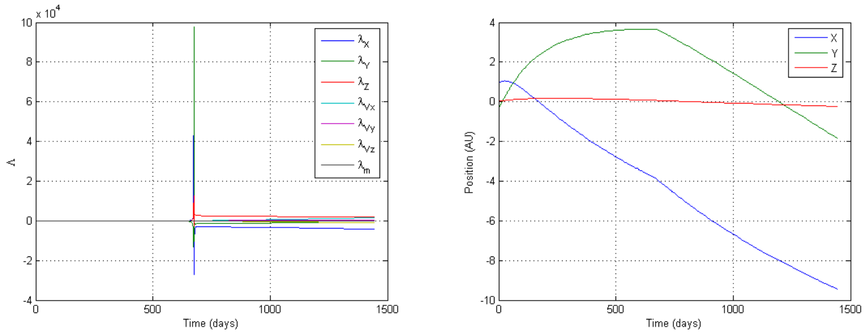

Figure 5.3: Costate dynamics along the Voyager 2 optimal trajectory

Note that with the patched conic approximation and a swing-by model[Ber01], we avoid any swing-by peak because the swing-by dynamics do not exist.

These different behaviours, between the state and the costate dynamics, make the resolution of the ODE related to the indirect TPBVP imprecise, and the state transition matrix $\phi(t, t_0)$ inaccurate. The Jacobian computed with the integration becomes ill-conditioned. A low condition number ensures that the computer has sufficient digits to handle the small differences in scale over a long time horizon. This is a source of discrepancy that prevents any solver from converging properly.

Robustness

The difficulty of this method is the convergence and robustness. A good initial estimation of the Lagrange multipliers is necessary. As initialisation methods are not readily available, finding an initial guess that would lie in the radius of convergence of a solution is a terrible task. Recently, Graichen and Petit[GP08] proposed a co-state initialisation approach using homotopy. The radius of convergence of an indirect method is considerably smaller than the radius of convergence of a direct method [Bet98]. Usually the solution found lies in the vicinity of the initial guess. The difficulty becomes stronger in more complex dynamics.

Recently, however an indirect method has been used in a N-body problem (GTOC 1). No swing-byes were performed, but the gravity field of Jupiter was used to perturb "optimally" the spacecraft dynamics.

Solution methods could have been:

- the use of homotopy methods. An initial solution of a simplified problem is used to initiate the algorithm. From iteration to iteration the problem is modified to match the original problem. Difficulties arise when the dynamics change, and very high precision is required to tackle the sensitivity of the Euler-Lagrange equations.

- a very good initial guess ...

Numerical Tools

The problem of the method comes also from the numerical tools used such as shooting methods. Usually, the efficiency of a method depends on the condition number of the Hessian. The rapidity of a method decreases with a bad condition number. The Hessian, unless provided, is often computed by finite differences using the Jacobian and quasi Newton approximation[GMS08],[KLW06].

A solution method could have been the use an augmented Lagrangian approach. Luenberger showed that the Hessian of the augmented Lagrangian tends toward the identity matrix. We thus have a good condition number, and a quicker convergence. But, the Hessian of the constraints does not change.

Numerical Accuracy

These solution methods are not sufficient, as they do not prevent the numerical inaccuracy of the state/costate during the swing-by[BH75, Bet98], and the implicit use of state transition matrices.

5.4 Summary, Conclusions

Methods	Constraint	Mathematical model	Numerical Difficulties
Non linear programming	Yes	continuous, discrete	high dimensionality, accuracy
Calculus of variation	No	continuous	TPBVP, stiffness
Maximum Principle	Yes	continuous, discrete	TPBVP, stiffness
Dynamic programming	Yes	continuous, discrete	curse of dimensionality

Table 5.1: Methods for optimal control, [MS62]

Table 5.1 sums up the properties of commonly used formulations for OCP.

Direct formulation and direct methods generally provide robust algorithms. They however lack accuracy. The pinpoint property of swing-bys, and their effective advantage make the interpolation step of co-location methods inadequate.

Looking for the control instead of the costates history is one of the reasons for the robustness. This point will be exploited for the algorithm of the next section.

Indirect methods are usually exact up to the integrator precision and w.r.t the problem model. However, because of the sensitivity of the Euler-Lagrange equations, it is often a challenging task to find a solution to the indirect problem formulation. A change in the dynamics induces high stiffness in the integration and the calculation of the gradient. We could not ensure that any indirect methods based on the Euler-Lagrange equations, and more particularly on the computation of a transition matrix, permits to solve swing-by optimal control problems.

It is important to note that today very few algorithms exist for the general optimisation of a trajectory in multi-body dynamics. We have now the basic directions for a solver capable of handling swing-bys. The ideal optimiser should encompass the following properties:

- accurate integration of the state equations, regardless of the costate dynamics.
- accurate integration of the costate equations, regardless of the state dynamics.
- important radius of convergence.
- robustness.

Integrating state and costate equations separately will provide more accurate and robust optimisation algorithms. This is the purpose of Chapter 6. Another optimisation method will be presented. It is a great solution method to handle the issue mentioned heretofore.

Chapter 6

Algorithm for Optimising Low-Thrust Interplanetary Transfers in Multi-Body Dynamics

6.1 Introduction

6.1.1 Issues and Objectives

The problem is that of transferring a spacecraft from one planet to another, taking into account their respective gravitational field, but also other massive bodies of the Solar System. Typically, the gravity acceleration of a planet is a second order perturbation, till the spacecraft reaches its sphere of influence. At this moment, the second order perturbation is no more a perturbation. This produces a change in the dynamics.

The difficulty of such transfers comes from the sensitivity of the Euler-Lagrange equations during the spiralling legs around the departing and target body. The Jacobian may also become ill-conditioned during the dynamical change from planetocentric to heliocentric, and from heliocentric to planetocentric.

This problem was chosen as it is a good benchmark for the swing-by transfer problem. When the planet sequence is unknown, there is no reason to expect an optimal trajectory including swing-bys. From this point, it would have been difficult to evaluate the adequacy of the approach.

Recently, Whiffen[Whi01] presented results and solutions to this transfer problem, although the method and algorithm is not described. This motivates us to seek an algorithm capable of finding such results.

6.2 Solution Methods

6.2.1 General considerations

Although the preceding developments (chapter 5) provide a good approach for solving the general two-body problem, it is not suitable for departing and arriving gravitational bodies or intermediate gravitational swing-by bodies.

The approach proposed here tackles the major numerical issues for the problem, such as:

- non linear objective
- non linear dynamics
- non convexity of the problem
- discontinuous control
- non autonomous state equation
- large difference of scale

Numerical discrepancies are avoided by using exact derivatives. To reduce the general sensitivity of the Euler-Lagrange equations, the difference of scale between the state and costate ODEs should be removed. The method, which is presented in this chapter, is based on gradient methods.

Gradient methods have been used in the past on simple applications. Jacobson[JM70] studies the case of the 2 dimensional orbital transfer of a spacecraft with full thrust. The Goddard or Dreyfus problem, considering the maximisation of the horizontal velocity of a launcher, has also been considered.

Indeed, gradient methods did not seem very popular, and this is understandable. Gradient methods, although robust, need to be adapted to each particular problem. Convergence often requires perfect tuning of convergence parameters.

6.2.2 Modified Gradient Method

Typically, a gradient method in optimal control theory relates the change in δu to the gradient of the Hamiltonian with respect to the control u.

We are trying to have a relation of the form of a feedback control law:

$$\delta u = \alpha + \beta \delta x$$

where α and β depend on the Hamiltonian H and its gradient H_u and H_x and possibly second order information H_{uu}. The problem is solved by a modified 2nd order gradient method [BH75]. The initial guess is provided by the sensitivity matrix procedure, described earlier, or the model described in chapter 3.

The general approach is an iterative process. The Lagrangian is expanded to second order around a reference state-costate trajectory. A forward - backward procedure is implemented to seek the optimal control. This is the key point about the method. Integrating separately the state and costate equations removes the numerical discrepancy when evaluating the Jacobian. As noted by Williamson and Tapley[WT72], such a method is more robust than the sensitivity approach. On the other side, we can state that the sensitivity approach is less stringent on the initial guess.

The required improvement on the final constraints gives updates on the control. The control follows a feedback law where the coefficient depends on the expected improvement in cost and final constraint reduction. At each iterative step, the second order development is respected and ensures weak displacements of the updates on the control.

Let's pose the general problem, as defined in section 5.1.1. We pose: $n_x = 7$ and $n_u = 3$ as the dimension of the state vector and the control vector for the space trajectory problem defined in Chapter 5. n_k is the dimension of the constraint vector ψ.

Again, all functions are continuously differentiable on their domain of definition for the variables $\mathbf{x}$ and $\mathbf{u}$. In addition, they have to be integrable with respect to time t. Thus: $f \in C^\infty(\mathbb{R}^{n_x} \times \mathbb{R}^{n_u})\backslash\mathbf{0})$, $\psi \in C^\infty \mathbb{R}^{n_x}$.

Constraints ψ also define consistent terminal constraints. This means that $\psi_x = \nabla_{x_f}\psi$ is of full rank.

Consider the cost function to minimise defined in equation 5.5. The Lagrangian is augmented with a quadratic term in the terminal constraints. The augmented Lagrangian[Ber82] is then:

$$\begin{aligned} L(\mathbf{x}, \mathbf{u}, \lambda, \nu, C_p) = -m(t_f) + \int_{t_0}^{t_f} \left(H(\mathbf{x}, \mathbf{u}, \lambda; t) - \lambda^T \frac{dX}{dt} \right) dt \\ + \nu^T \psi(\mathbf{x}(t_f), t_f) \\ + \psi(\mathbf{x}(t_f), t_f)^T C_p \psi(\mathbf{x}(t_f), t_f) \end{aligned} \tag{6.1}$$

With:

$$H(\mathbf{x}, \mathbf{u}, \lambda; t) = \lambda^T f(\mathbf{x}, \mathbf{u}; t) + \frac{1}{2}\Delta\mathbf{u}^T C_\delta \Delta\mathbf{u}$$

where $\mathbf{x}(t)$ is the state vector at time t, $\lambda(t) \in \mathbb{R}$ the costate vector at time t, $\mathbf{u}(t)$ the minimising control vector, $\nu \in \mathbb{R}^{n_k}$ is a static Lagrange multiplier assigned to the constraints, $C_p \in \mathcal{M}(\mathbb{R})_{n_k,n_k}$ is a penalty matrix for the constraints, and $C_\delta \in \mathcal{M}(\mathbb{R})_{n_u,n_u}$. The term $m(t_f)$ stands for the objective function to be minimised (equation 5.5).

Thus differentiating to second order Lagrangian equation 6.1 around a nominal trajectory $\{\bar{\mathbf{x}}(t), \bar{\lambda}(t), \bar{\mathbf{u}}(t)\}$, $t \in T = [t_0, t_f]$, with admissible perturbations $\{\delta\mathbf{x}, \delta\lambda, \delta\mathbf{u}\}$, and 2 integrations by parts give*:

$$\begin{aligned}
dL = &-\frac{dm(t_f)}{d\mathbf{x}_f}\delta\mathbf{x}_f \\
&+ \nu^T(\frac{1}{2}\delta\mathbf{x}_f^T\frac{d^2\psi}{dx_f^2} + \frac{d\psi}{dx_f})\delta\mathbf{x}_f + \delta\nu^T\psi \\
&+ (\frac{d\psi}{d\mathbf{x}}\delta\mathbf{x}_f + \frac{1}{2}\delta\mathbf{x}_f^T\frac{d^2\psi}{d\mathbf{x}^2}\delta\mathbf{x}_f)^T C_p\psi_f \\
&+ \psi_f^T C_p(\frac{d\psi}{d\mathbf{x}}\delta\mathbf{x}_f + \frac{1}{2}\delta\mathbf{x}_f^T\frac{d^2\psi}{d\mathbf{x}^2}\delta\mathbf{x}_f) \\
&- \lambda_f^T\delta\mathbf{x}_f + \lambda_0^T\delta\mathbf{x}_0 \\
&+ \int_T (H_x\delta\mathbf{x} + H_u\delta\mathbf{u} + H_\lambda\delta\lambda)dt \\
&+ \int_T \frac{d\lambda + \delta\lambda^T}{dt}\delta\mathbf{x}dt - \int_T \delta\lambda^T\frac{d\mathbf{x}}{dt}dt \\
&+ \frac{1}{2}\int_T (\delta\mathbf{x}^T H_{xx}\delta\mathbf{x} + \delta\mathbf{u}^T H_{uu}\delta\mathbf{u} + \delta\mathbf{x}^T H_{xu}\delta\mathbf{u} + \delta\mathbf{u}^T H_{ux}\delta\mathbf{x})dt \\
&+ \int_T \delta\lambda^T H_{\lambda x}\delta\mathbf{x} + \delta\mathbf{x}^T H_{x\lambda}\delta\lambda + \delta\lambda^T H_{\lambda u}\delta\mathbf{u} + \delta\mathbf{u}^T H_{u\lambda}\delta\lambda)dt \\
&+ \int_T o(\|\delta\mathbf{x}\|^2, \|\delta\lambda\|^2, \|\delta\mathbf{u}\|^2)dt
\end{aligned} \tag{6.2}$$

For the minimisation of L, the first order part should vanish, and the second order part should be positive[BH75].

The admissible perturbations $\delta x \in L^\infty(T, \mathbb{R}^{n_x})$, $\delta\lambda \in L^\infty(T, \mathbb{R}^{n_x})$ and $\delta u \in L^\infty(T, \mathbb{R}^{n_u})$ ensure that the Taylor development is correct. In other words, for all $(\mathbf{x}, \lambda, \mathbf{u}) \subset \mathcal{B}(\bar{\mathbf{x}}, o(\|\mathbf{x}\|)) \times \mathcal{B}(\bar{\lambda}, o(\|\lambda\|)) \times \mathcal{B}(\bar{\mathbf{u}}, o(\|\mathbf{u}\|))$, equation 6.2, without higher order terms, remains valid. This assumption should be respected and checked numerically when updating the control. This issue will be of concern later.

Because H_x, H_u, H_{xx}, H_{xu}, H_u can be computed analytically, all the derivatives of L and subsequent development are precise up to the ode solvers and machine precision.

*We use the mathematical conventions cited at the beginning of this dissertation, for the derivatives. Variables are omitted for conciseness, but all functions are evaluated for $\mathbf{x}$, λ and $\mathbf{u}$

With a fixed terminal time t_f, initial conditions $\mathbf{x}_0 = \mathbf{x}(t_0)$ considered fixed, we have the following conditions:

$$\delta \mathbf{x}_0 = 0 \tag{6.3}$$

$$\lambda_f^T = -\frac{dm(t_f)}{d\mathbf{x}_f} + \nu^T \frac{d\psi}{d\mathbf{x}_f} + \psi^T C_p \frac{d\psi}{d\mathbf{x}_f} \tag{6.4}$$

The second equation defines the transversality conditions for the problem considered.

6.2.3 Optimal Control

We are seeking the control that minimises the Lagrangian L given in equation 6.1.

The optimal control is given by the necessary conditions:

$$H_u + \delta\mathbf{x}^T H_{xu} + \delta\lambda^T H_{\lambda u} + \delta\mathbf{u}^T H_{uu} = 0 \tag{6.5}$$

Now assume H_{uu} has full rank, and is thus invertible. From this equation, we deduce the optimal perturbation policy for the nominal control:

$$\delta\mathbf{u} = (H_{uu})^{-1}(-H_u - H_{ux}\delta\mathbf{x} + H_{u\lambda}\delta\lambda) \tag{6.6}$$

Note that as the nominal control is not necessarily optimal, we can not directly use $H_u = 0$. The first part provides a descent direction to minimise the Hamiltonian H. The second and third terms provide feedback information on the state and costate trajectories, respectively $\bar{\mathbf{x}}(t)$ and $\bar{\lambda}(t)$.

To ensure that the perturbation $\delta\mathbf{u}$ leads to an improvement in the control, we should check that $H_{uu} > 0$ for the minimisation problem 5.5. This condition, with the Euler-Lagrange equation give us conditions for a strong minimum.

Besides, $\delta\mathbf{x}$ is given with the following equations, which should be integrated forward and concurrently with the state dynamic equations:

$$\frac{d\delta\mathbf{x}}{dt} = f_x(\bar{\mathbf{x}}, \bar{\mathbf{u}}; t)\delta\mathbf{x} + f_u(\bar{\mathbf{x}}, \bar{\mathbf{u}}; t)\delta\mathbf{u} \tag{6.7}$$

$$\delta\mathbf{u} = \bar{\mathbf{u}} - \mathbf{u} \tag{6.8}$$

$$\delta\mathbf{x}_0 = 0 \tag{6.9}$$

Proposition 6.2.1. *For the continuous formulation, the amplitude of* $\|\delta\mathbf{x}\|$ *is limited by the amplitude of* $\|\delta\mathbf{u}\|$. *If* $\|\delta\mathbf{u}\|$ *respects second order information, then* $\|\delta\mathbf{x}\|$ *should also respect second order information.*

We are seeking a law on the co-state vector and the final constraints perturbations.

Proposition 6.2.2. *Introducing a matrix $A(t) \in \mathcal{M}(\mathbb{R})_{nx,nx}$, we can write:*

$$\delta\lambda(t) = A(t)\delta\mathbf{x}(t) \tag{6.10}$$

Proof. The state transition matrix approach shows that we can relate the state vector $\mathbf{x}$ to the costate vector λ. We thus use:

$$\begin{aligned}\mathbf{x}(t_f) &= \phi_{11}(t_f, t)\mathbf{x}(t) + \phi_{12}(t_f, t)\lambda(t)\\ \lambda(t) &= \phi_{21}(t, t_f)\mathbf{x}(t_f) + \phi_{22}(t, t_f)\lambda(t_f)\end{aligned}$$

□

We get the dynamical equation on the costate vector:

$$\frac{d\lambda}{dt} = -H_x^T + H_u H_{uu}^{-1}(H_{ux} + f_u^T A) \tag{6.11}$$

Note that λ is different from the solution to the Maximum Principle. There is however an equivalence when $H_u(t) = 0$.

And also for the matrix A:

$$-\frac{dA}{dt} = H_{xx} + Af_x + f_x^T A - (H_{ux} + f_u^T A)^T H_{uu}^{-1}(H_{ux} + f_u^T A) \tag{6.12}$$

The matrix $A(t)$ is a symmetric matrix.

Eventually, supposing H_{uu} is positive definite, the optimal control perturbation, minimising the Lagrangian L, becomes:

$$\delta\mathbf{u} = -H_{uu}^{-1}H_u - H_{uu}^{-1}(H_{ux} + \mathbf{f}_u^T A)\delta\mathbf{x} \tag{6.13}$$

This has the form of a feedback control law. We can thus expect a property of robustness around a nominal trajectory regarding dynamical perturbations.

Proposition 6.2.3. *Around the nominal trajectory $\bar{\mathbf{x}}(t)$, and for the nominal control $\bar{\mathbf{u}}(t)$, we have the equivalence:*

$$H_u(\bar{\mathbf{x}}, \lambda, \bar{\mathbf{u}}; t) = 0 \Leftrightarrow \delta\mathbf{u}(t) = 0$$

Proof. For convenience, we omit $\mathbf{x}$, λ and $\mathbf{u}$ in the writing of H, H_u and H_{uu}. In the first direction (right to left), we have:

$$\delta\mathbf{u}(t) = H_{uu}(t)^{-1}(H_{ux}(t) + f_u^T(t)A(t))\delta\mathbf{x}(t)$$

As $\delta\mathbf{x}(t_0) = 0$, recursively we have also $\delta\mathbf{x}(t) = 0$ and $\delta\mathbf{u}(t) = 0$.

In the other direction, as $\delta\mathbf{u}(t) = 0$, no changes are made on the current trajectory compared to the nominal trajectory. Using equation 6.7, we get $\delta\mathbf{x} = 0$.

Thus we have: $H_{uu}^{-1}H_u = 0$. As H_{uu} is symmetric and positive definite, we conclude $H_u = 0$. □

We then pose:

$$I(t) = \int_{t_0}^{t} H_u^T H_{uu}^{-1} H_u dt \tag{6.14}$$

Proposition 6.2.4. *The function $I(t)$ is increasing and positive. For an optimal trajectory, $I(t_f) = 0$.*

Proof. As H_{uu} is supposed positive definite, the integrand $H_u^T H_{uu} H_u \geq 0$ is positive, continuous and integrable. Thus, if $I(t_f) = 0$, we have $H_u = 0$. Conversely, for $H_u = 0$, the integrand is identically null, thus $I(t) = 0$. □

6.2.4 Terminal State Constraints

Terminal equality constraints are taken into account using a weighting matrix C_p and a Lagrange vector λ. The use of Lagrange multipliers, as they have been introduced in equation 6.1, ensures the reduction of constraint violations at each iteration.

Proposition 6.2.5. *We have the following relationships:*

$$\delta\lambda = A\delta\mathbf{x} + Kd\nu \tag{6.15}$$

$$\delta\psi = K^T\delta\mathbf{x} + Qd\nu \tag{6.16}$$

where $K(t) \in \mathcal{M}(\mathbb{R})_{nx,nk}$ and $Q(t) \in \mathcal{M}(\mathbb{R})_{nk,nk}$.

Proof. With the state transition matrix, we write:

$$\begin{aligned} \mathbf{x}(t_f) &= \phi_{11}(t_f,t)\mathbf{x}(t) + \phi_{12}(t_f,t)\lambda(t) \\ \lambda(t) &= \phi_{21}(t,t_f)\mathbf{x}(t_f) + \phi_{22}(t,t_f)\lambda(t_f) \end{aligned}$$

The transversality conditions (equation 6.3) for the terminal constraints provide the value for $\lambda(t_f)$. We get:

$$\begin{aligned} (1 - \phi_{21}(t,t_f)\phi_{12}(t_f,t))\lambda(t) = {} & \phi_{21}(t,t_f)\phi_{11}(t_f,t)\mathbf{x}(t) \\ & + \phi_{22}(t,t_f)\psi_{\mathbf{x}_f}^T\nu \\ & + \phi_{22}(t,t_f)(-\frac{dm(t_f)}{d\mathbf{x}_f} + \psi_{\mathbf{x}_f}^T C_p \psi) \end{aligned}$$

A small perturbation of the variables $\mathbf{x}$ and ν gives:

$$\begin{aligned} \delta\lambda(t) = {} & (1 - \phi_{21}(t,t_f)\phi_{12}(t_f,t))^{-1}\phi_{21}(t,t_f)\phi_{11}(t_f,t)\delta\mathbf{x}(t) \\ & + (1 - \phi_{21}(t,t_f)\phi_{12}(t_f,t))^{-1}\phi_{22}(t,t_f)\psi_{\mathbf{x}_f}^T\delta\nu \end{aligned}$$

□

We have the equations:

$$\frac{dK}{dt} = ((f_x - f_u H_{uu}^{-1} H_{ux})^T + A(f_u H_{uu}^{-1} f_u^T)^T)K \tag{6.17}$$

and:

$$\frac{dQ}{dt} = K^T f_u H_{uu}^{-1} f_u^T K \tag{6.18}$$

These equations are integrated while on the nominal trajectory. We note immediately that directly computing A, K, Q with ODEs (equations 6.17 and 6.18) prevents the inversion of the transition matrix, and thus this approach is less prone to numerical errors. Particularly for sensitivity equations, where elements of the transition matrix can get high values (bad condition number).

Equation 6.2 gives the final conditions to solve the IVPs (6.12, 6.17, 6.18):

$$A(t_f) = \nu^T \frac{d^2\psi}{dx^2}(\mathbf{x}(t_f)) + \psi^T C_p \frac{d^2\psi}{dx^2} + \frac{d\psi}{dx}^T C_p \frac{d\psi}{dx} \tag{6.19}$$

$$K(t_f) = \frac{d\psi}{dx}(\mathbf{x}(t_f)) \tag{6.20}$$

$$Q(t_f) = 0 \tag{6.21}$$

The optimal perturbation, taking into account the Lagrange vector ν for the constraints, becomes:

$$\delta\mathbf{u} = -H_{uu}^{-1} H_u - H_{uu}^{-1}(H_{ux} + \mathbf{f}_u A)\delta\mathbf{x} - H_{uu}^{-1}\mathbf{f}_u K \delta\nu \tag{6.22}$$

We then improve at the same time the optimality of the current control and the reduction of the terminal constraints.

To guarantee a reduction of the terminal error (final constraints), we must have $Q(t_0)$ negative definite. Using equation 6.16, we would then ($Q(t_0) < 0$) get the Lagrange vector updates:

$$d\nu = -Q(t_0)^{-1}\delta\psi \tag{6.23}$$

Proposition 6.2.6. *For $Q(t_0)$ to be negative definite, we must have:*

- $H_{uu}(\bar{\mathbf{x}}, \mathbf{u}, \lambda; t) > 0$.
- *the linear system around* $(\bar{\mathbf{x}}, \bar{\mathbf{u}})$ *is fully controllable.*
- $\psi_x(\mathbf{x}; t_f)$ *has full rank.*

Proof. If the linear system (equation 6.7) is fully controllable, and $H_{uu} > 0$, $f_u H_{uu}^{-1} f_u^T$ is positive definite. Then, using $Q(t_f) = 0$, we have

$$Q(t_0) = -\int_{t_0}^{t_f} K^T f_u H_{uu}^{-1} f_u^T K dt$$

Note that equation 6.17 is a linear equation of the form:

$$\begin{aligned}\frac{dK}{dt} &= kK\\ k &= f_x + f_u(-H_{uu}^{-1}H_{ux}^T + H_{uu}^{-1}f_u)K\end{aligned}$$

k has also full rank because of the condition of controllability. As $K(t_f) = \frac{d\psi}{dx}(\mathbf{x}(t_f))$ has also full rank, $K(t)$ has full rank too[†] for all $t \in [t_0, t_f]$.

The integral is thus positive definite, and $Q(t_0) < 0$.

A similar proof can be found in [JM70], where a transition matrix is used, and $H_{uu}(\bar{\mathbf{x}}, \mathbf{u}^*, \lambda; t)$ is considered. □

This condition is however not sufficient. Nothing guarantee we have $\psi(\mathbf{x}; t_f) = 0$ for some control $\mathbf{u}(t)$. Let use the notations of [Tre07], and denote by $Acc(T)$ the set of accessible points for the solution of the optimal control problem (equations 5.1, 5.5, and 5.3) in transfer time T. The system is fully controllable if $Acc(t_f > t_0)$ contains $\mathbb{R}^{n_x}$. However, here we are only interested with $Acc(t_f - t_0 > T_{min}) \neq \emptyset$. The minimum transfer time T_{min} is generally found by studying the minimum time problem. If $t_f - t_0 > T_{min}$, we have at least one solution, and the condition $Q(t_0) < 0$ is sufficient to have $\psi(\mathbf{x}; t_f) = 0$.

After each step, the Lagrange vector is updated with $\delta\nu$. The vector ν can reach high values to satisfy the constraints. Compared to an approach where the terminal constraints are handled by a constant penalty matrix, this approach ensures that, for each iteration, both the cost function J and the terminal constraints violation ψ are reduced.

Intermediate state constraints (equality and inequality) such as minimum swing-by altitude, minimum distance to Sun, ... have not been explicitly considered in this study. The formulation is sufficiently well posed to easily handle intermediate constraints using a penalty approach.

Terminal inequality constraints can be handled using either penalty methods as above, or slack variables. The use of slack variables amounts to introducing unknown parameters.

It is possible to solve the problem with unknown parameters. Such parameters can be the departure orbital parameters, or the launch date. This has not been considered here. But, using the exact same development as in this section, using a Lagrangian approach, provides the equations for the update of the unknown parameters. Note however that any additional parameter needs additional ODE to solve.

[†] Every solution can be written using the unit basis solution, then by definition every solution has full rank.

As the purpose of this study is to find swing-bys, it is worth mentioning that besides not using intermediate constraints to force swing-by, we can place constraints on the minimum radius of a possible swing-by. This would require using penalty functions depending on the relative distances with planets, and over the entire spacecraft trajectory.

6.2.5 Constraint on the Control

As the thruster power is limited, the thrust force is itself bounded. We thus have the condition:

$$\|\mathbf{u(t)}\| \leq 1 \tag{6.24}$$

Here, we propose a method that removes this constraint (equation 6.24).

Given the space trajectory problem (equation 5.1), we seek the control functions $\alpha \subset \mathcal{C}^0(\mathbb{R})$, $\beta \subset \mathcal{C}^0(\mathbb{R})$ and $\delta \subset \mathcal{C}^0(\mathbb{R})$, and an application $s : \mathbb{R} \rightarrow [0, 1]$ such that the bounded control can be written:

$$\mathbf{u}(t) = s(\delta(t)) \begin{bmatrix} \cos\alpha(t)\cos\beta(t) \\ \cos\alpha(t)\sin\beta(t) \\ \sin\alpha(t) \end{bmatrix} \tag{6.25}$$

The control $\mathbf{u}$ is still constrained, but the variables $\alpha(t)$, $\beta(t)$ and $\delta(t)$ are unconstrained. Some restrictions apply:

Proposition 6.2.7. *For the minimisation problem, if the application s respects the following:*

- *being at least twice continuous differentiable.*
- *being bounded.*
- *having exactly one maximum and one minimum, possibly multiple.*

the amplitude $\|\mathbf{u}\|$ of the control can be bang-bang.

Proof. The first point is is due to the linear dependency of $\mathbf{u}$ in the Hamiltonian H, and the necessary computation of H_u, H_{uu} and H_{xu}.

For the solution to be bang-bang, H_u should vanish. Note then that:

$$\frac{\partial H}{\partial \delta} = \rho(\mathbf{x}, \lambda; t)\frac{\partial s}{\partial \delta}$$

where ρ is a switching function [Ber01]. Remember that, because of the maximum principle, $\rho < 0$ implies s at a maximum, and $\rho > 0$ implies s at a minimum.

The only way we can make H_u vanish, is to have $s(t) = 0$. The function ρ can only vanish if we are on a singular solution.

Now, we assume the second derivative of H w.r.t. δ is positive, for the minimisation problem, and:

$$\frac{\partial^2 H}{\partial \delta^2} = \rho(\mathbf{x}, \lambda; t) \frac{\partial^2 s}{\partial \delta^2} > 0$$

Thus:

- if $\rho(t) > 0$ and $s(t) = 0$, we have $\frac{\partial^2 s}{\partial \delta^2}(t) > 0$. This means that s is a minimum.

- if $\rho(t) < 0$ and $s(t) = 0$, we have $\frac{\partial^2 s}{\partial \delta^2}(t) < 0$. This means that s is a maximum.

By contradiction, we show that if s has more than one maximum and one minimum, we have different thrust levels, and the solution cannot be bang-bang. This maximum and minimum can however be multiple. □

To respect the range $[0, 1]$, s can be easily scaled or translated without changing this result.

Such applications can be for example: sin, $\sin^2$, $\cos^2$, ... or linear combinations of them.

Note however, that although this transformation can be appealing, in theory it introduces strong singularities in the control. The introduction of matrix C_δ circumvents this point in practice.

This approach can be compared to other solution methods. If the control structure is known a-priori, we can simplify the canonical system. The parametrisation of the instants of commutation of the control allows to remove the control amplitude, and the mass-related ODEs.

We can also use a penalty function in the Hamiltonian to limit the amplitude of $\mathbf{u}$. The major issue of this approach is that the control might exceed its allowed bounds in the process. Thus, during the process and for a bad initial guess, nothing prevents the control amplitude from being negative or exceeding unity! Even though this allows extra controllability for seeking optimal solutions, the solution might not be feasible. In addition, with a penalty approach, it would have been difficult to have $H_u = 0$, while trajectories can be very good.

Our approach guarantees feasible and optimal trajectories and controls.

6.3 Convergence Issues

The present development is valid and provides an optimal control under strict conditions:

1. the second order development is respected.
2. no conjugate points are encountered.
3. the problem is feasible.

6.3.1 Minimisation

One of the major issues when dealing with equation 6.12 is the possible unbounded matrix. The matrix A can rapidly increase and prevent the algorithm from converging. This appears when H_{uu} is near singular, or not sufficiently positive definite. As a result the feedback control law given by equations 6.6 or 6.22 does not exist.

Jacobson, Bullock and, Williamson and Tapley[WT72] introduced the definition of conjugate point .

Conjugate Point Along an extremal, the conjugate time t_c is defined as the instant where the determinant of the second derivative of H wrt $\mathbf{u}$ vanishes. The point $\mathbf{x}(t_c)$ is a conjugate point.

A conjugate point is defined as a point where many extremals pass. In other words, at a conjugate point, two neighboring extremals with the same performance index, meet. So if a conjugate point is encountered, we know in our case that a minimum extremal exists in the neighbourhood.

Detecting a conjugate point resume to test for the rank of a matrix. Prussing and Sandrik[PS02] provide a conjugate point test using a condition on the rank of a matrix. If the determinant of that matrix vanishes at some time t_c, then we may have a conjugate point at t_c. This is an interesting test, as often it is numerically difficult to detect the conjugate point. If the A matrix get an unbounded value at some point, the conjugate point is likely to be placed some time before.

Forcing $H_{uu}(t) > 0$ for all $t \in [t_0, t_f]$ around the nominal trajectory $\bar{\mathbf{x}}(t)$ ensures that we are always on a unique extremal and we can provide a feedback control (implicit function theorem). The most common approaches to make a given Hessian positive-definite, is with solving a Trust Region problem, or using a Levenberg algorithm.

Trust Region Problem [Kel99] Let's note q the quadratic model of a function $H(\mathbf{u})$ at $\mathbf{u} = \mathbf{u}_k$:

$$q(\mathbf{u}) = H_u(\mathbf{u} - \mathbf{u}_k) + \frac{1}{2}(\mathbf{u} - \mathbf{u}_k)^T H_{uu}(\mathbf{u} - \mathbf{u}_k)$$

We thus define the neighbourhood $D = \{\mathbf{u} \in \mathbb{R}^{nu} \quad s.t. \quad \|\mathbf{u} - \mathbf{u_k}\| < \Delta\}$. The standard Trust-Region problem is thus to solve:

$$\min_{\mathbf{u} \in D} q(\mathbf{u})$$

The positive definiteness of H_{uu} is controlled by both a Trust region algorithm, and the matrix C_δ. The Trust-Region problem then permits to find a positive-definite update of the Hessian matrix H_{uu}. It also limits the length of $\delta\mathbf{u}(t)$[‡]. C_δ is a penalty matrix which ensures the positive definiteness of H_{uu}, and slight variations in $\mathbf{u}(t)$ (see also [Jar83, Jar75]).

Another solution to get rid of this problem is to change the value of C_p. Changing C_p, provides a slightly different costate trajectory, thus a different control and a new neighboring extremal, which eventually gets away from the conjugate point.

6.3.2 Constraints reduction, and Problem Feasibility

Feasibility

We suppose proposition 6.2.6 is respected. Since the method proposed here is an iterative method, the constraint satisfaction is the primary concern, before the optimality of the control. If terminal constraints can be satisfied, an optimal control respecting the BVP (dynamic, terminal constraint, transversality conditions, cost) exists.

Thus we do not initially solve the optimal control problem, but first the constraint problem. We solve at first:

$$\begin{aligned} &\min_{\mathbf{u}} \rho J + \psi(\mathbf{x}_f)^T C_p \psi(\mathbf{x}_f) \\ &s.c. \quad \frac{d\mathbf{x}}{dt} = f(\mathbf{x}, \mathbf{u}; t) \\ &\mathbf{x}(t_0) = \mathbf{x}_0 \end{aligned} \tag{6.26}$$

with $\rho = 0$. That means that for the terminal cost problem the transversality conditions (equation6.3) are modified. Once the constraints have been significantly reduced, the problem cost (equation 5.5) is considered ($\rho = 1$) in the minimisation problem. It is however important that the constraints on the control be respected,

[‡] $|\alpha(t)| \leq \Delta$, $|\beta(t)| \leq \Delta$ and $|\delta(t)| \leq \Delta$

otherwise we have no real indication of the problem feasibility. This legitimates the approach on the control constraint explained above.

Indeed, there are results on the local controllability of nonlinear system, using Lie brackets[Tre07]. Proposition 6.2.6 only refers to local controllability. There are no results of *full controllability*, i.e. the existence of a control function that would bring any initial state to the desired final state, particularly when the dynamics change (from interplanetary to planetary and vice versa). It is thus important to assess as soon as possible the feasibility of the problem, to gain more time. To make the problem feasible, the time of flight is usually increased, as the minimum time problem provides a limit on controllability.

Constraint Reduction

In the Lagrangian 6.1, the matrix C_p controls the constraints feasibility. Setting C_p only, without the Lagrange multipliers ν, can lead to convergence. However, the convergence can be slow. High C_p does not ensure reduction of constraint. Low C_p leads to slow convergence, and might favour cost rather that constraints.

The use of both ν and C_p in the Lagrangian equation is indeed an augmented Lagrangian. The augmented Lagrangian comes from a regularisation (Yosida-Moreau). Due to Hestenes, Powell, Rockafellar and Bertsekas[Ber82], it permits to provide a poor initial guess while still converging (global convergence). The penalty parameter does not need to reach a very high value to get a satisfactory constraint feasibility.

It has the following advantages:

- it avoids the use of penalty parameters that are difficult to control correctly (penalty methods)
- it ensures the existence of a saddle point, even though a duality gap exists. The duality gap is the difference between the primal and dual objective values.

The duality gap is the difference between these 2 quantities.

$$\min_x \sup_\lambda J(x) + \lambda^T \psi(x)$$
$$\sup_\lambda \min_x J(x) + \lambda^T \psi(x)$$

It is not null when the primal problem is non convex. The role of C_p is to convexify the primal problem. In this case, C_p should not be seen as a penalty matrix for the constraint, but as a regularisation matrix of the dual problem. Furthermore, if ψ is chosen to have a sufficiently positive definite Hessian, for

C_p large enough, the Lagrangian $L(\mathbf{x}, \mathbf{u}, \lambda, \nu, C_p)$ becomes strictly convex[Ber95]. Thus the primal has an optimal solution. By the strong Duality Theorem[DT97], the dual problem has also an optimal solution, and there is no duality gap.

Remark Let's note J the objective function, and L the augmented Lagrangian with Lagrange variable ν, and assume these functions are sufficiently regular. The properties of the augmented Lagrangian are summarised below[LP]:

- $L(\nu) < J^*$, $\forall \nu \geq 0$
- $L(\nu^*) = \max_\nu L(\nu) = J^*$
- L is differentiable and concave for ν

$$\frac{\partial L}{\partial \nu} = \psi(\mathbf{x}_f) \tag{6.27}$$

To update C_p, we should use equation 6.27 to get ascent the step in L. However, opposite to most augmented Lagrangian approaches[LT98, BCM], we can compute the Lagrange vector ν. The value of C_p must be found to have the good ascent direction and to avoid ill-conditioning. Usually, a small value of C_p must be used initially. C_p must be increased in case of infeasibility.

6.3.3 Improvement in Optimality

In addition, if the new control $\mathbf{u} = \bar{\mathbf{u}} + \delta\mathbf{u}$ does not reduce the Lagrangian significantly, a line search is performed. The line search, with parameter ϵ, uses the control $\mathbf{u}_\epsilon = \bar{\mathbf{u}} + \delta\mathbf{u}_\epsilon$ with $\delta\mathbf{u}_\epsilon = \epsilon\alpha + \beta\delta\mathbf{x} + \epsilon d\nu$. The parameter ϵ is decreased from 1 to 0 until a satisfactory change is encountered.

For $\epsilon = 0$, we have obviously (proposition 6.2.3) an equivalence between the perturbed and the nominal control and trajectory:

$$\mathbf{u}_{\epsilon=0} = \bar{\mathbf{u}} \tag{6.28}$$

$$\mathbf{x}_{\epsilon=0} = \bar{\mathbf{x}} \tag{6.29}$$

The line search is supposed to correct the calculated improvement when the second order development does not hold strictly. For $\epsilon \in [0, 1]$, the line search procedure produces a control update $\delta\mathbf{u}_\epsilon$ in a descent direction.

Proposition 6.3.1. *For the line search algorithm, the change in cost, between the value of the functional $L(\mathbf{u}; t)$ evaluated for the nominal control $\mathbf{u}$, and the value of $L(\mathbf{u}_\epsilon; t)$ evaluated for the perturbed control $\mathbf{u}_\epsilon$ is:*

$$dL(\mathbf{u}_\epsilon) = -\epsilon(1 - \frac{\epsilon}{2})I(t_f) + o(\|\delta\mathbf{x}\|^2)$$

Proof. First, note that $I(t_f) > 0$, as was demonstrated earlier. The expression of L in equation 6.1, with the Euler equations and conditions 6.3, gives:

$$\begin{aligned} dL = &\int_{t_0}^{t_f} dH dt \\ &+ (\frac{1}{2}\delta x_f^T \frac{d^2\nu^T\psi}{dx_f^2})\delta x_f + \delta\nu^T\psi \\ &+ (\frac{d\psi}{dx}\delta x_f + \frac{1}{2}\delta x_f^T\frac{d^2\psi}{dx^2}\delta x_f)^T C_p\psi_f \\ &+ \psi_f^T C_p(\frac{1}{2}\delta x_f^T\frac{d^2\psi}{dx^2}\delta x_f) \end{aligned}$$

With:

$$dH = H_x\delta\mathbf{x} + H_u\delta\mathbf{u} + \frac{1}{2}\delta\mathbf{x}^T H_{xx}\delta\mathbf{x} + \frac{1}{2}\delta\mathbf{u}^T H_{uu}\delta\mathbf{u} + \delta\mathbf{u}^T H_{ux}\delta\mathbf{x} + \delta\mathbf{x}^T H_{xu}\delta\mathbf{u} + o(\|\delta\mathbf{x}\|^2, \|\delta\mathbf{u}\|^2)$$

Replacing with $\delta\mathbf{u}$, and collecting terms, gives:

$$\begin{aligned} dH = &-\epsilon H_u(H_{uu}^{-1}H_u) + \frac{1}{2}\epsilon^2(H_{uu}^{-1}H_u)^T H_{uu}(H_{uu}^{-1}H_u) \\ &+ (H_x + H_u H_{uu}^{-1}(H_{ux} + f_u A) - \frac{\epsilon}{2}(H_{uu}^{-1}H_u)^T H_{uu}(H_{ux} + f_u A) - \frac{\epsilon}{2}(H_{uu}^{-1}H_u)^T H_{ux})\delta\mathbf{x} \\ &- \frac{\epsilon}{2}\delta\mathbf{x}^T((H_{ux} + f_u A)^T H_u) \\ &+ \frac{1}{2}\delta\mathbf{x}^T(H_{xx} + (H_{ux} + f_u A)^T H_{uu}(H_{ux} + f_u A) \\ &+ (H_{uu}^{-1}H_u)^T H_{uu}(H_{ux} + f_u A) + (H_{ux} + f_u A)^T H_{ux})\delta\mathbf{x} \\ &+ H_u H_{uu}^{-1}\mathbf{f}_u K\delta\nu \\ &+ \frac{\epsilon}{2}H_u^T H_{uu}^{-1} f_u K\delta\nu \\ &+ \frac{1}{2}\delta\mathbf{x}^T(H_{ux} + f_u A)^T H_{uu}^{-1} f_u K\delta\nu \\ &- \delta\mathbf{x}^T H_{xu}H_{uu}^{-1}\mathbf{f}_u K\delta\nu \\ &- \delta\nu^T K\mathbf{f}_u H_{uu}^{-1}H_{ux}\delta\mathbf{x} \\ &+ o(\|\delta\mathbf{x}\|^2) + o(\|\delta\nu\|^2) \end{aligned}$$

$\delta\nu$ can be replaced with equation 6.23.

Or also, on a nominal trajectory, and after simplifications:

$$dH = (\frac{1}{2}\epsilon^2 - \epsilon)H_u^T H_{uu}^{-1}H_u + o(\|\delta\mathbf{x}\|)$$

Thus:

$$dL = (\frac{1}{2}\epsilon^2 - \epsilon)I(t_f) + o(\|\delta\mathbf{x_f}\|) \leq 0$$

□

This line search procedure differs from the Trust-Region problem. While the Trust-Region problem ensures a reduction of the Lagrangian by positivity, the line search procedure ensures the validity of the second order developments, regarding $\|\delta\mathbf{x}\|$ and $\|\delta\mathbf{u}\|$. However, if the Trust-Region problem is solved with a sufficiently small radius, the second order developments are respected and the line search on ϵ is generally useless.

In practice, a good ode solver should be used to avoid additional truncation errors.

6.4 Numerical approach of the Continuous Problem Control

6.4.1 Continuous control issue

As opposed to Direct methods (section 5.2), no approximations are made during propagation of both state $\mathbf{x}$ and costate λ. However, the control needs to be stored, because it is used for both backward and forward propagations, as opposed to indirect methods (section 5.3.2). It is difficult to store a continuous solution, without making approximations.

Discretisation of the control has an impact on the full controllability of the system, and the quality of the solution. This is a major issue, when the dynamics include different changes in scale.

Thus, we can wonder what is the best discrete approximation of the continuous control $\mathbf{u}(t)$, and in particular for bang-bang solutions. We thus seek the discretised control $\mathbf{u}(t)$ that is close, in a certain metric of $L^\infty(T,U)$, to the continuous optimal control $\mathbf{u}^*(t)$ of problem 5.5. Indeed, this encourages an adaptive mesh procedure that would:

1. increase or move points close to a bang-bang commutation. We can adapt the time mesh size in the neighbourhood of a bang-bang switching. The variation of the control will still be smooth, but the density of points allows for big variation on the time interval, and respect second order information.

2. adapt the meshing when the dynamics change (e.g. when going from the interplanetary phase to a planetocentric phase).

3. adapt the meshing when the optimality conditions are not satisfied.

For the bang-bang issue, two approaches can be considered, depending on the problem nature: autonomous (non dependant of time), and non autonomous systems. We can:

- add to each control thrust direction a date of application. This amounts to providing specific time interval to each thrust control.
- observe the control derivative wrt time. Strong variations in the control suggest placing a switch point.

6.4.2 Optimal placement for Autonomous Systems

The state is extended with the time variable t. The integration is done with a variable $s \in [s_0, s_f]$. A change of variable must be done appropriately with equations 6.17, 6.18 and 6.12.

The control is extended. It is described with:

$$\mathbf{u}_e = [\mathbf{u}, \Delta t] \tag{6.30}$$

Δt holds for the interval of application of the control $\mathbf{u}$. We must include the constraint:

$$t(s_f) = tof \tag{6.31}$$

As Δt is constant on each s-interval, this constraint is equivalent to:

$$\sum_i \Delta t = tof \tag{6.32}$$

The bounded interval $[s_0, s_f]$ can thus be chosen such that with $\Delta t \in [0, 1]$ and $s_0 = 0$, we have $s_f \geq tof$.

In addition, the Hamiltonian is extended with:

$$H_e(\mathbf{u}_e, \mathbf{x}, \lambda; s) = H(\mathbf{u}, \mathbf{x}, \lambda; s) + \Delta t \Delta u_\delta \tag{6.33}$$

The variable u_δ is the bang-bang part of the control $\mathbf{u}$, and $\Delta u_\delta(t) = u_\delta(t) - u_\delta(t^-)$ is the difference of control between two immediately successive instants of time. The process is then allowed to freely and optimally change the control duration to reduce the cost function value.

Proposition 6.4.1. *For the system, linear in the control* $\mathbf{u}$*, the extended Hamiltonian* H_e *is continuous on an extremal for a bang-bang solution.*

Proof. Consider $t^- = t_{i-1}$ and $t^+ = t_i$ the instant immediately before and after a bang-bang switch. Consider extended dynamics:

$$\begin{aligned}\mathbf{f}(\mathbf{x}, \mathbf{u}; t) &= \mathbf{f}_0(\mathbf{x}; t) + \mathbf{f}_1(\mathbf{x}; t)\delta(t)\mathbf{u}(t)\\ \frac{dt}{ds} &= \Delta t\end{aligned}$$

with the initial condition $t(s_0) = t_0$, under the constraints $t(s_f) = t_f$ and $\psi(\mathbf{x}; t_f) = 0$. The function $\delta(t)$ is the control amplitude, and $\|\mathbf{u}\|(t) = 1$.

The Hamiltonian is:

$$\begin{aligned} H(\mathbf{x}, \lambda, \mathbf{u}, \delta, \Delta t; s) = \lambda^T \mathbf{f}_0(\mathbf{x}; s) + \\ \lambda^T \mathbf{f}_1(\mathbf{x}; s)\delta\mathbf{u}) + \\ \lambda_t \Delta t + \Delta t(\delta - \delta^-) \end{aligned}$$

As $\mathbf{x}$, $\lambda(t)$, $\mathbf{u}$ and the dynamics are continuous, we have:

$$\begin{aligned} H(\mathbf{x}, \lambda, \mathbf{u}, \delta, \Delta t; t^+) - H(\mathbf{x}, \lambda, \mathbf{u}, \delta, \Delta t; t^-) &= \lambda^T \mathbf{f}_1(\mathbf{x}; t)(\delta^+\mathbf{u}^+ - \delta^-\mathbf{u}^-) \\ &+ \Delta t^+(\delta_i - \delta_i^-) \\ &- \Delta t^-(\delta_{i-1} - \delta_{i-1}^-) \\ &+ \lambda_t(\Delta t^+ - \Delta t^-) \end{aligned}$$

With: $\Delta t^+ = t_{i+1} - t^+$ and $\Delta t^- = t_i - t^-$, and $t = t^+ = t^-$. On an extremal the control is bang-bang, and $\Delta t^+(\delta_i - \delta_i^-)$ and $\Delta t^-(\delta_{i-1} - \delta_{i-1}^-)$ vanish. This simplifies to:

$$H(\mathbf{x}, \lambda, \mathbf{u}, \delta, \Delta t; t^+) - H(\mathbf{x}, \lambda, \mathbf{u}, \delta, \Delta t; t^-) = -\lambda_t \Delta t^-$$

The necessary condition of optimality for Δt, on the segment $[t_{i-2}, t_{i-1}]$ concludes: $\lambda_t \Delta t^- = 0$. □

This shows there is no numerical issue that can appear because of bang-bang.

For fixed time transfer, with non gravitating bodies (other than the Sun), this approach is easily applied.

It is possible to turn the problem into an autonomous one. However, this task may not be easy, or may require approximations.

For example, in the space trajectory problem, the dynamics depend on the planet position. We thus need to evaluate the 1st and 2nd time derivatives of the planets position $\mathbf{R}_p$ wrt the Sun. We have then:

$$\frac{d\mathbf{R}_p}{dt} = \mathbf{V}_p \tag{6.34}$$

$$\frac{d\mathbf{V}_p}{dt} = -\mu_{SUN} \frac{\mathbf{R}_p}{\|\mathbf{R_p}\|^3} + \gamma \tag{6.35}$$

$$\frac{d^2\mathbf{V}_p}{dt^2} = -\mu_{SUN} \frac{d}{dt} \frac{\mathbf{R}_p}{\|\mathbf{R_p}\|^3} + \frac{d\gamma}{dt} \tag{6.36}$$

Usually R_p is given using an ephemeris database. We need to differentiate the ephemeris function with respect to time. This reduces to calculating the perturbation function $\gamma(t)$. This perturbation function depends on the influence of other planets and bodies in the Solar System.

Of course, it is possible to assume the planet's dynamics to be exactly Keplerian ($\gamma(t) = 0$). But this limits the accuracy. Trajectory scenarii considering multiple swing-bys over long time horizon might result in approximate solutions.

6.4.3 Mesh Placement Strategies

In the general case, for non autonomous systems, we consider different strategies to update as best as possible the mesh.

1. use information on the smoothness of the dynamics.
2. use the size of steps $\delta\mathbf{u}$ or $\delta\mathbf{x}$.
3. use information on the trajectory solution. The solution mesh provided by a RK solver is used for the control mesh.

The first case has already somewhat been treated in numerous publications about direct methods.

For the second strategy, consider the size of $\delta\mathbf{u}$ or $\delta\mathbf{x}$. Even though $\delta\mathbf{u}$ and $\delta\mathbf{x}$ might respect second order developments, their size is also limited by adding intermediate control points. This approach is equivalent to studying H_u. It also provides information on the steepness of the optimal control, for the reference trajectory.

The third strategy considers a constant ratio of the number of control points over the number of state points. The control is then updated every given number of points. This limits the error.

Both strategies 2 and 3 can solve global controllability issues, where the dynamics are highly nonlinear, to ensure a reduction of the Lagrangian. However, strategy 3 can be quite expensive in memory usage for rapidly changing dynamics.

Strategy 2: Using derivatives

When the control amplitude undergoes a rapid change, it is likely that we can have a bang-bang commutation. More specifically, H_u provides the necessary information on the rapid change in the control.

Proposition 6.4.2. *Bang-bang switches occur when the part of $H_u(t)$ related to the control amplitude changes sign.*

Proof. Simply note that $H_u(\mathbf{x}, \lambda; t)$ is related to the switching function $\rho(\mathbf{x}, \lambda; t)$.

$$H_\delta(\mathbf{x}, \lambda, \mathbf{u}; t) = F(\frac{\lambda_v}{m}\mathbf{u} - \frac{\lambda_m}{g_0 I_{Sp}})$$

For the optimal direction, $\mathbf{u}^*$ is aligned with the primer vector λ_V. □

This proposition is of a very strong practical interest as H_u is computed when constructing the control 6.6, 6.22.

Basically, we give thresholds for:

- the minimum segment length Δt_{min}

- the maximum segment length Δt_{max}
- the maximum slope γ_{max}

For each of these thresholds, points can be added or removed. The general algorithm is presented in algorithm 2.

```
Input: Initial mesh points $\{t_i\}_{i=1..N}$, threshold $h$
Output: Adapted mesh $\{t_i\}_{i=1..M}$
$M = N$;
Measure the slope of each component of $\Delta H_u(t_i) = H_u(t_{i+1}) - H_u(t_i)$;
foreach mesh points $i$ s.t. $\Delta H_u(t_i) > h$ do
    $t = t_i$;
    $n = \Delta H_u(t_i)/h$;
    $dt = (t_{i+1} - t_i)/n$;
    Add $n - 1$ points between $t_i$ and $t_{i+1}$, with step $dt$;
    copy the control $\mathbf{u}_i$ at $t_i$ at the new control points.;
    $M \leftarrow M + n$;
end
```

Algorithm 2: Step Update

We get the following proposition:

Proposition 6.4.3. *Algorithm 2 converges with a finite number of mesh points.*

On convergence, when $H_u = 0$ the algorithm does not add any new mesh points. When adding mesh points, because of the conservative law on the new mesh points placement, the trajectory is not modified. Consequently, when adding points the convergence of the algorithm is not degraded.

6.5 Algorithm and discussions

6.5.1 Presentation of the algorithm

For the first step, a nominal control $\mathbf{u}(t)$ and ν are provided. The sensitivity equations are integrated backward. We then construct a feedback control law. This control law is used in the forward propagation to compute an improved trajectory. The nominal control is updated after each successful iteration. The process is repeated until we get the desired accuracy in the initial and final state, and gets close to zero.

A line search procedure seeks the value of ϵ such that the merit function decreases. If the error measure has not significantly decreased, the penalty parameter C_p is increased.

```
Input: initial control $\bar{\mathbf{u}}$, initial trajectory $\bar{\mathbf{x}}$
Output: optimal control
$\psi_{old} \leftarrow \psi$;
select $\eta_1 > 0$ and $\eta_2 > 0$;
while $\|\psi\| \geq \eta_1$ and $I(t_f) \geq \eta_2$ do
    Backward integration of eq. 6.12;
    Backward integration of eqs. 6.17 and 6.18;
    $\epsilon \leftarrow 1$;
    while $\|\psi\| \geq \|\psi_{\mathbf{old}}\|$ do
        Compute new control $\mathbf{u}_\epsilon$;
        Compute new trajectory $\mathbf{x}$;
        $\epsilon \leftarrow \epsilon/2$;
    end
    $\bar{\mathbf{u}} \leftarrow \mathbf{u}$;
    $\bar{\mathbf{x}} \leftarrow \mathbf{x}$;
end
```

Algorithm 3: Modified Gradient Algorithm (simple form)

6.6 Academic Examples

6.6.1 Goddard's Problem

The Goddard's problem is the classical 2-D rocket problem.

The dynamics are:

$$\frac{d}{dt}\begin{bmatrix} X \\ Y \\ V_x \\ V_y \end{bmatrix} = \begin{bmatrix} V_x \\ V_y \\ F\cos\alpha \\ F\sin\alpha - g \end{bmatrix}$$

X, Y are respectively the rocket horizontal position and altitude. V_x and V_y are respectively the rocket horizontal and vertical velocity. We suppose the thrust to be $F = 2g$, and $g = 1$. This accounts for a constant thrust. The mass flow rate is not taken into account. $\alpha(t)$ is the 1-D steering control law.

We want to maximise the final horizontal velocity $J = V_x(t_f)$ with the final constraints:

$$Y(t_f) = 1$$
$$V_y(t_f) = 0$$

The final constraints and the objective function impose the rocket to convert all of its velocity into a horizontal component.

The initial guess is $u(t) = \pi/2 \quad \forall t \in [0, t_f]$. The algorithm converged in 27 iterations, with a tolerance of 1e-6 on the final constraints. The optimal cost found is $V_x(t_f) = 1.915$. Figure 6.1 displays the trajectory.

The problem can also simply be solved using the Maximum Principle. Assigning the Lagrange variables $[\lambda_X, \lambda_Y, \lambda_{Vx}, \lambda_{Vy}]$ to the respective state variables

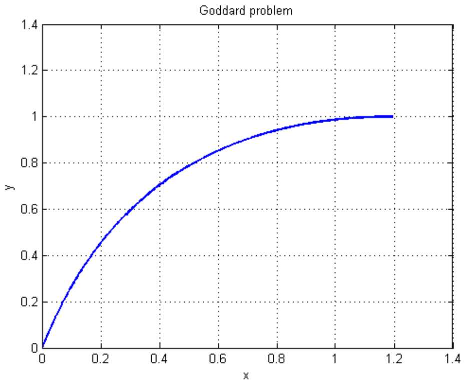

Figure 6.1: Goddard's problem: state trajectory

$[X, Y, V_x, V_y]$, we have the following Euler-Lagrange equations:

$$\frac{d}{dt}\begin{bmatrix}\lambda_X\\ \lambda_Y\\ \lambda_{V_x}\\ \lambda_{V_y}\end{bmatrix} = \begin{bmatrix}0\\ 0\\ -\lambda_X\\ \lambda_Y\end{bmatrix}$$

and the transversality conditions:

$$\lambda_x(t_f) = 0 \quad \mathit{and} \quad \lambda_{V_x}(t_f) = 1$$

With a simple Newton type solver, we converge in 7 iterations and the optimal cost found is $V_x(t_f) = 1.96$. The trajectory and the optimal control are presented on figure 6.2.

The difference in cost (2.5%) comes from the discretisation scheme that was used with our solver.

6.6.2 Orbital Transfer

This example deals with the transfer from an orbit to another. We start from an initial circular Low Earth Orbit ($R_{LEO} = 20000km$), and target a circular Earth orbit at $R_{MEO} = 42000km$ [MC64].

The transfer is coplanar. The transfer time is fixed at 4 days. The control is given by the steering angle α and the throttle variable δ. We have the limited thrust constraint: $0 \leq \delta \leq 1$.

The spacecraft initial mass is $m_0 = 1000kg$, and is equipped with a $F = 5N$, $Isp = 2000s$ thruster. This example is for the sake of illustration, and may not be of interest in practice. The objective is to maximise the final mass.

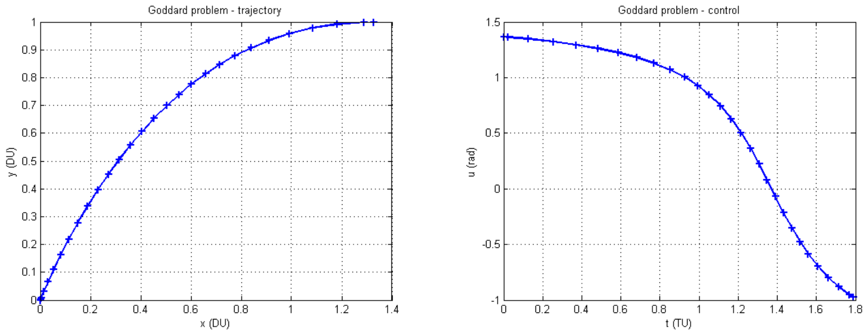

Figure 6.2: Goddard's problem solution with Maximum Principle: state trajectory(left), control (right)

The dynamics are:

$$\frac{d}{dt}\begin{bmatrix} r \\ V_r \\ V_\theta \\ m \end{bmatrix} = \begin{bmatrix} V_r \\ \frac{V_\theta^2}{r} - \frac{\mu_{Earth}}{r^2} + \frac{F}{m}\sin\alpha \\ -\frac{V_r V_\theta}{r} + \frac{F}{m}\delta\cos\alpha \\ -q\delta \end{bmatrix}$$

r is the radial distance. V_r and V_θ are respectively the radial and ortho-radial velocities norms. m is the spacecraft mass. The Earth gravitation effect is taken into account through the constant μ_{Earth}.

The final constraints can be written:

$$\begin{aligned} (r(t_f) - R_{MEO})^2 &= 0 \\ V_r(t_f) &= 0 \\ V_\theta(t_f) - \sqrt{\frac{\mu_{Earth}}{r(t_f)}} &= 0 \end{aligned}$$

These constraints impose the spacecraft to be on a final circular orbit, with radius R_{MEO}.

Variable	Our tool (gradient)	T3D (indirect)
Time of flight Initial mass m_0	4 days 1000 kg	
Final Mass	932.60 kg	932.02 kg

Table 6.1: Mars-Earth validation case comparisons

Table 6.6.2 compares the final result with T3D. They are close, but slight differences the relative accuracy of the integration and the stopping criterion for the optimality norm.

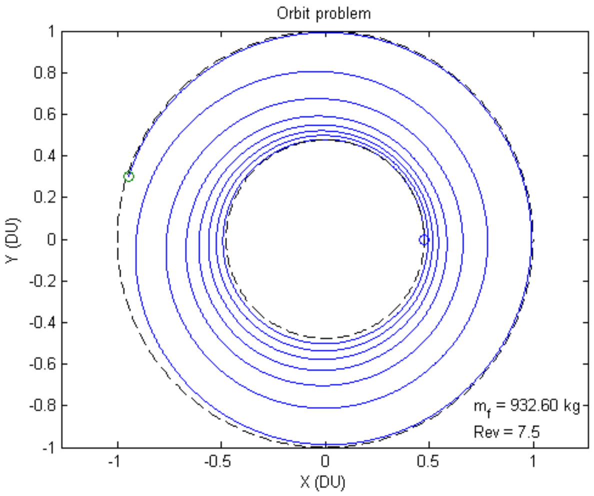

Figure 6.3: Optimal Low-Thrust Orbital Transfer Trajectory from R_{LEO} to R_{MEO}

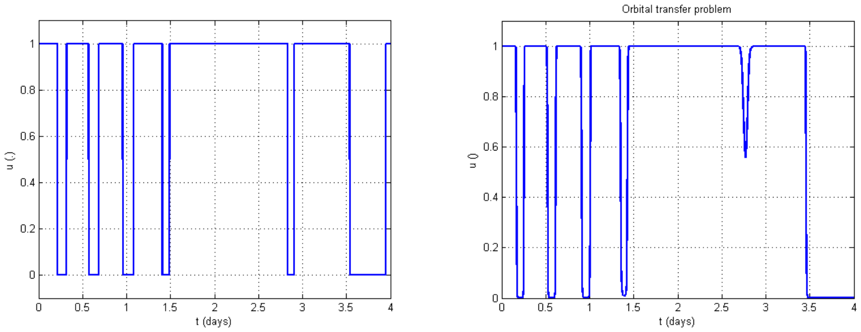

Figure 6.4: Optimal Low-Thrust Orbital Transfer Control from R_{LEO} to R_{MEO}. Comparison of T3D control (left) and the method control (right).

The transfer includes 7 revolutions (figure 6.3).

Looking at figure 6.4, the control given by T3D[DM04] is quite similar to ours. The control in our method is not bang-bang on the final segment. This is due to the limited optimality that we requested (10^{-5}). From a practical point of view, convergence was much longer than for T3D, but somewhat easier once we find the appropriate iteration scheme and parameters ($C_p, C_\delta, ...$). The terminal constraints are easily validated, while the optimality condition for the control might not be met. It is then important to put a stringent condition on the optimality (H_u) to expect the bang-bang structure.

6.7 Conclusion, discussions

We derive equations to solve the optimal control problem 5.5, 5.1, 5.3, 5.4. We get a feedback control law that allows computing a robust trajectory regarding dynamical perturbations. Most of these equations are not new, although these equations are different from DDP, or the Successive-Sweep or Second Order Weak Variation algorithms.

We focus on understanding why such an approach is relevant to our problem. Decoupling Euler-Lagrange equations during the integration adds robustness. Backward integration makes the integration stable numerically. The sensitivity of the Euler-Lagrange equations is spread along the state and costate trajectory. As long as we are able to solve properly the ODEs (equations 6.12, 6.17, 6.18) we are able to find a better or optimal solution.

The main advantage of the gradient method is thus its ability to handle complex dynamics. Compared to a transition matrix approach, a natural explanation of the method would be to say that the more equations we have, the more sensitivity we remove.

Convergence is relatively slow compared to usual algorithms (e.g. shooting). The accuracy of the solution requires many iterations, even though the process might converge quite easily. In the end, for complex dynamics, this method provides a high fidelity trajectory

Often in the literature, authors prefer talking about "better solution" rather than optimal solution. The use of the augmented Lagrangian, the control transformation and the continuous control approximation should prevent such conclusions. In our examples, we always get bang-bang control. To my knowledge, no implicit bang-bang control results have been published for space trajectory problems.

Chapter 7

Numerical Examples

7.1 Mars - Earth rendezvous transfer

This case is a validation case. We compare the results with those of T3D[DM04] Consider a Mars-Earth transfer. We minimise the consumption, and the time of flight is fixed. The terminal constraints impose a rendezvous with the arrival planet (Earth). The planets gravity is not considered. Launch and Rendezvous are performed with massless planets.

The dynamics are those described in chapter 5. The rendezvous terminal constraints, respecting the rank condition, are:

$$\psi(\mathbf{x}; t_f) = \begin{bmatrix} \mathbf{r}_f - \mathbf{r}(t_f) \\ \mathbf{v}_f - \mathbf{v}(t_f) \end{bmatrix} \tag{7.1}$$

We consider a launch date on September 1st, 2009. The time of flight is 350 days. The spacecraft has a 0.215N thruster with 3500s Isp.

Table 7.1 displays the results of the optimisation, and a comparison with a solution provided by T3D. The trajectory computed with our algorithm is displayed on figures 7.1 and 7.2.

Variable	Our tool (gradient)	T3D (indirect)
Launch date	01/09/2009	
Time of flight	350 days	
Initial mass m_0	500 kg	
Final Mass	359.08 kg	359.11 kg

Table 7.1: Mars-Earth validation case comparisons

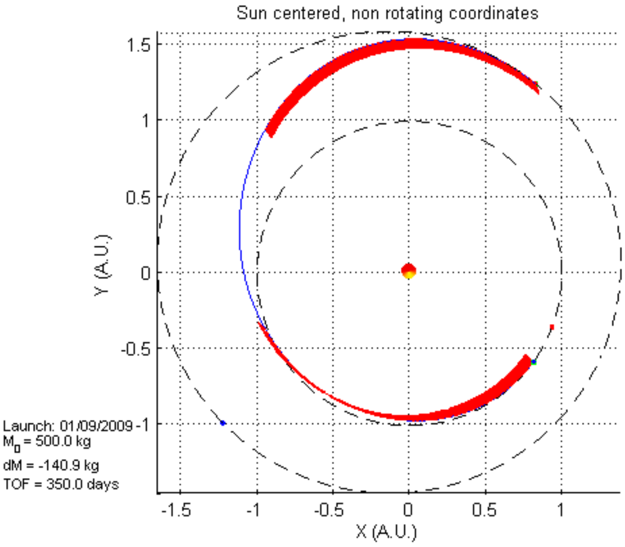

Figure 7.1: Mars - Earth two-body transfer

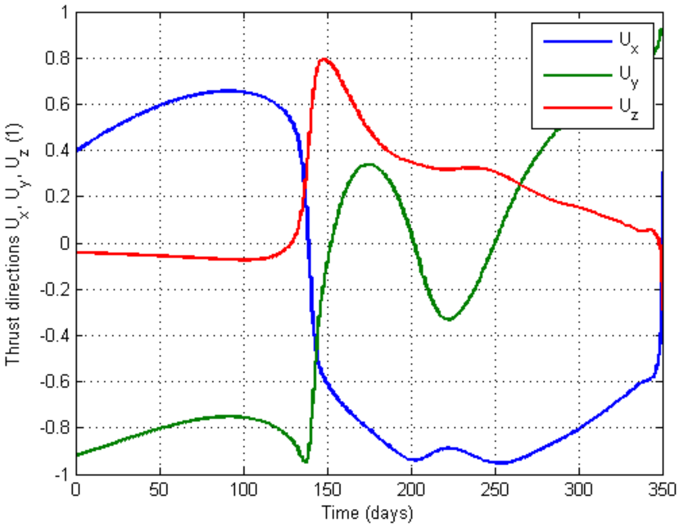

Figure 7.2: Mars - Earth two-body transfer control

7.2 Earth - Mars with capture and escape phases

These examples can be found in [Oly08].

We study direct transfers considering the departure and arrival planets gravity. The spacecraft is initially on a high orbit around the departure planet. For the capture cases, the terminal constraint (eq. 7.2) is a circular orbit of radius R_{HEO}.

$$\psi(\mathbf{x}; t_f) = \begin{bmatrix} \|\mathbf{R(t_f)}\| - r_f \\ \|\mathbf{V(t_f)}\| - \sqrt{\frac{\mu_p}{r_f}} \\ \mathbf{R}(t_f)^T \mathbf{V}(t_f) \end{bmatrix} \tag{7.2}$$

where $\|\mathbf{R(t_f)}\|$ and $\|\mathbf{V(t_f)}\|$ are respectively the relative position and velocity vectors wrt the arrival planet.

The planet positions are propagated from a given epoch date. This avoids numerical discrepancies while evaluating gradients. However, we assume that over a long time period the orbital elements do not change.

The initial guess for escape phases is a somewhat tangential thrust law, until the relative energy reaches zero. With a quasi-tangential thrust law we increase largely the orbit energy to escape rapidly. The thrust amplitude is fixed. If there are too many revolutions, we either increase the thrust amplitude, or let the solver remove them.

In all other cases, the initial guess was the un-converged solutions provided by the indirect approach.

Earth to Mars transfer

The spacecraft uses a $3000s$ Isp thruster. The initial mass is $m_0 = 500kg$. The launch date has been chosen to have a short transfer time, and it is on 15/01/2016. The time of flight is fixed to 350 days.

We compare the different cases, with or without Earth or Mars gravity field.

Phases are characterized by the zero energy condition (figure 7.3) relative to the central body of interest. Among this point, we have hyperbolic conditions relative to that same central body.

In Tables 7.2, 7.3 and 7.4, we compare the different possible cases, with the capture phase, the escape phase, both or none. For the escape phase, the initial orbit is a High Earth Orbit (HEO), circular with a radius $R_{HEO} = 100000km$. For the capture phase, the final orbit is High Mars Orbit (HMO), circular with radius $R_{HMO} = 100000km$.

To limit infeasible cases we assign a high radius for R_{HMO} around Mars. This also limits the influence of the multi-body dynamics so that we can rigorously compare two-body capture with three-body capture. Indeed, when setting a small

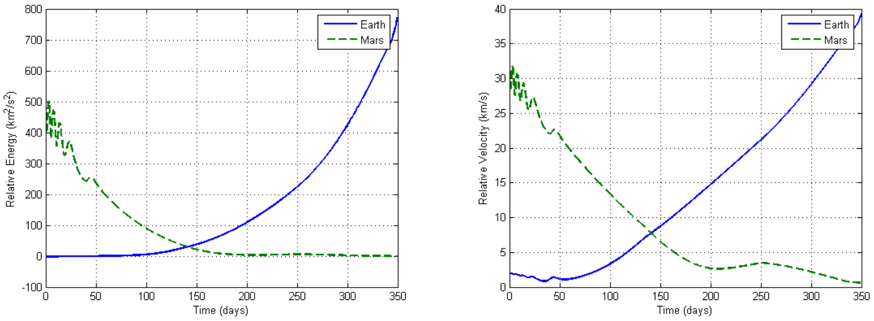

Figure 7.3: Earth HEO escape - Mars HMO insertion, spacecraft total energy and relative velocity with respect to Earth and Mars.

radius, much time and thus consumption is spent for correcting the final radius, which is of no interest for this first comparison (although we can wonder whether the radius R_{HMO} has a real influence on the capture phenomenon).

In Table 7.2, we consider capture problems. The initial guesses for the capture cases were quite poor, as they did not include the capture phase.

Case 2a is a two-body escape to a Mars HMO insertion. Case 2b is a direct result from case 2a as the time of flight is the duration of two-body segment of case 2a. There is thus no third body capture phenomenon, nor rendezvous. Case 2c, take the information from case 2b, time of flight, to rendezvous Mars. The 2-body capture mass is computed from the capture solution. It is the mass when the spacecraft is captured by the arrival planet. This value could then be compared with the rendezvous solution. Thus cases 2b and 2c can be somewhat compared.

Cases	1 Earth Mars RdV	2a Earth Mars HMO	2b Earth Mars Capture	2c Earth 2-body Mars Capture
Initial mass m_0	500 kg			
Time of flight (days)	350	350	342.0	342.0
Final V_{rel} (km/s)	0	0.674	0.597	0
Final Mass m_f (kg)	407.7	402.1	408.2	407.7

Table 7.2: Earth-Mars transfer comparisons, capture cases. ($\|\psi\| \leq 10^{-6}$)

Because the duration of the capture phase is not very long, we have a direct indicator of the influence of the planet gravity over the performance. Comparisons are difficult, as the problems are different, but the case studied here slightly demonstrates a trend. In fact, capture phases tend to reduce the consumption. The stronger the gravity acceleration of the arrival planet, or the closer to the planet is the final orbit, the more benefit we should have on the consumption.

Figure 7.4 depicts the Earth - Mars capture. The spacecraft is captured on a High Mars orbit (HMO).

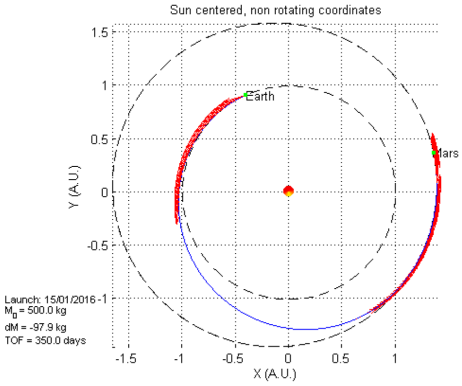

Figure 7.4: Earth - Mars HMO capture trajectory.

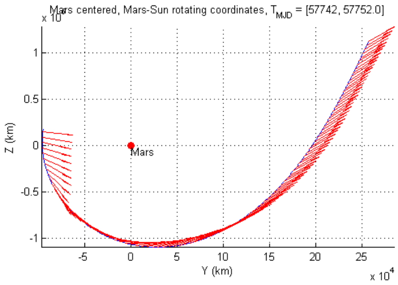

Figure 7.5: Earth - Mars HMO capture, close view of the capture phase.

In Table 7.3, we consider escape phases. Case 3a considers a departure from a HEO. Case 3b takes as initial conditions, the conditions at the end of the escape phase of case 3a. We change the launch date, the initial mass, and introduce a hyperbolic excess velocity.

According to this example, a low-thrust escapes under-performs a V_∞ launch. The escape date is $T_0 + 51.7$ days, and $V_\infty = 1.07$ km/s. Because V_∞ is not null (fig. 7.3), the escape phase has indeed an influence on the interplanetary travel. The optimality principle dictates that both trajectory are optimal from the point of escape to Mars. Thus, it is likely that the difference of performance comes

Cases	1 Earth Mars RdV	3a Earth HEO escape Mars RdV	3b Earth V_∞ Mars RdV
Time of flight (days)	350	350	298.3
Departure from t_0	0	0	51.7
Initial mass m_0 (kg)	500	500	476.6
$V_\infty (km/s)$	0	-	1.068
Final Mass m_f (kg)	407.7	374.5	385.7

Table 7.3: Earth-Mars transfer comparisons, with escape phase. ($\|\psi\| \leq 10^{-7}$)

from the bad orientation of the velocity when escaping, for case 3a. In addition, in the multi-body case, when escaped we are still under the influence of Earth's gravity, such that we tend to increase gravity losses. Although, the position of the spacecraft on the circular HEO might be of importance for the orientation of the velocity vector. This position has not been optimized here.

On figure 7.6, we take the same initial conditions, but we rendezvous Mars, as for the usual two-body problem.

Figure 7.7 depicts a close view of the escape phase. It includes coast arc. This show the method to be able to correctly identify the switching of the optimal control on the entire trajectory, despite the sensitivity of the problem and the difference of scale in the dynamics.

Figure 7.7 depicts 4 coast phases. Trajectories have not many revolutions around Mars (fig. 7.4) because of the high altitude of 100000 km and the low gravitational parameter μ.

In table 7.4, we study the escape and capture problem. Case 4a considers an HEO escape to an HMO insertion. Case 4b is a direct consequence of case 4a, as the time of flight and the launch date give the two-body transfer leg of case 4a. Case 4c, is the two-body transfer problem taking as initial conditions the outgoing conditions of the escape phase of case 4a, and considers a two-body rendezvous with Mars.

The difference between cases 4b and 4c is significant. Although, we did not perform further check, as the solution found for case 4c might as well be a local solution. There are many degree of freedom due to the un-specified direction of $\mathbf{V}_\infty$.

Figure 7.8 gives the Earth escape to Mars capture, considering both Earth and Mars gravity field. The trajectory is spiraling out to escape Earth gravity field, then enter the interplanetary phase. Close to Mars, the spacecraft is captured and spirals in to reach the desired Mars altitude as on fig. 7.5.

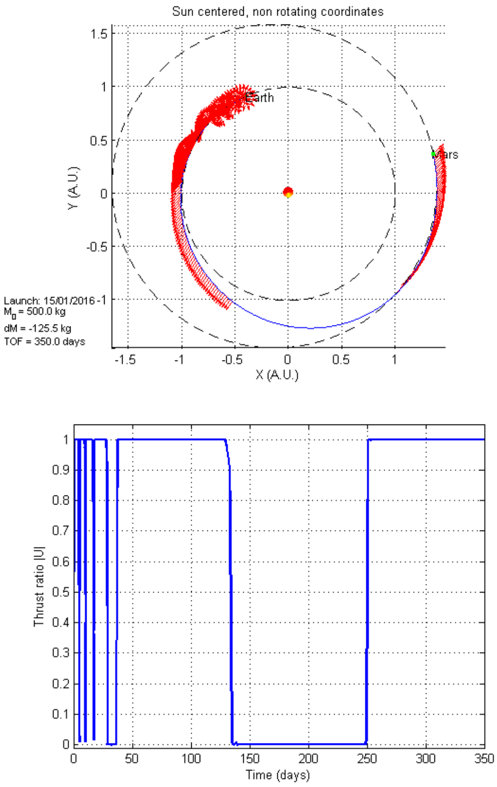

Figure 7.6: Earth HEO escape - Mars trajectory and control.

Back to Earth

The spacecraft is the same as in the previous example, but the initial mass is set to $m_0 = 300kg$. The launch date is on 15/01/2018. We study the capture phase with planet Earth.

We seek a return trajectory, from Mars to Earth, departing from Mars on the 15/01/2018.

The spacecraft is placed on a HEO around the Earth. Figure 7.9 depicts the spiraling movement of the spacecraft around the Earth, for the different circular radius targets. We assume the spiral begins when the spacecraft is captured. Compared to the Mars capture, the number of revolution around Earth is more important than around Mars, for the same altitude. This is due to a stronger gravity acceleration. It appears that different classes of solutions exist, seeing at the final inclination of the orbits. Indeed, the difference from a trajectory point of view can be seen on figure fig:earthcaptureclose.

A bad initial guess for this problem can result in many iterations.

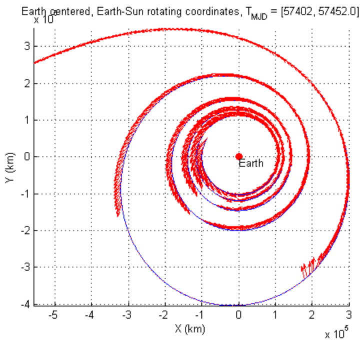

Figure 7.7: Earth HEO escape, close view of the escape phase.

Cases	1 Earth Mars RdV	4a Earth HEO esc. Mars HMO capt.	4b Earth HEO esc. 2-body Mars capt.	4c Earth V_∞ 2-body Mars capt.
Time of flight (days)	350	350	292.1	292.1
$V_\infty(km/s)$	0	-	1.088	1.088
Initial mass m_0 (kg)	500	500	476.6	476.6
Final Mass m_f (kg)	407.7	375.4	378.7	322.8

Table 7.4: Earth-Mars transfer comparisons, with escape and capture phases. ($\|\psi\| \leq 10^{-6}$)

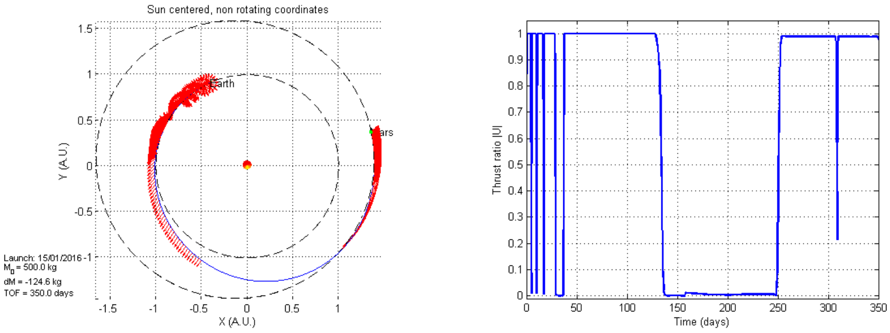

Figure 7.8: Complete Earth escape - Mars capture trajectory and control.

Comments

The optimization of such trajectory is more difficult than it appears to be. Most of the time, it is difficult to say whether the solver simply fails to converge, or the problem simply has no solution. The feasibility procedure is of great help in practice.

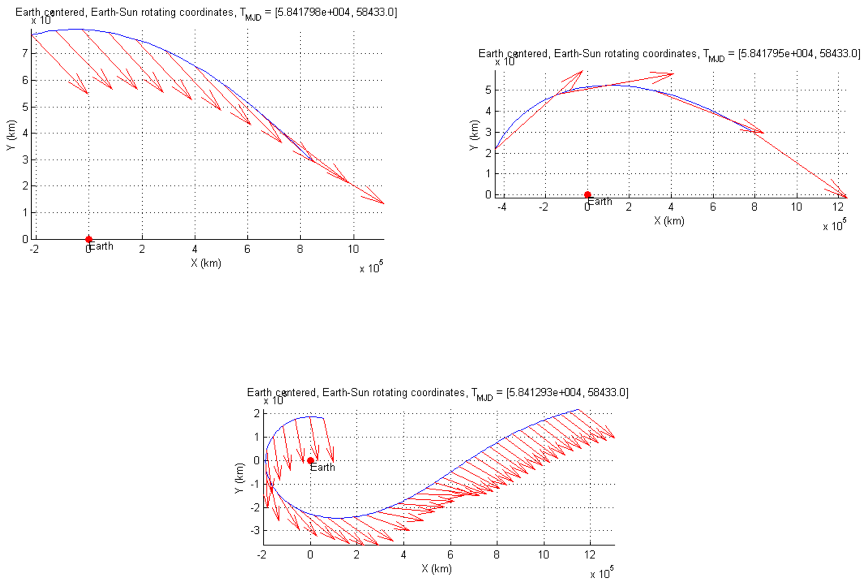

Figure 7.9: Close look at the Earth capture phase, Mars - Earth transfer, for respectively 800 10^3 km, 500 10^3 km, 200 10^3 km radius

R_{HEO} (km)	Time of Flight (days)	N_{REV}	Final mass (kg)
two-body	300	-	247.82
800	300	< 1	247.26
500	300	< 1	247.42
200	300	1+	233.73

Table 7.5: Mars-Earth with capture phase, continuation on R_{HEO}, departing from Mars on 15/01/2018. Number of revolutions around Earth.

Our algorithm does not seem to suffer from a problem of sensitivity, as being captured on a negative energy orbit around a planet is a simple task. The difficulty arises when we are looking for a specific orbit. Indeed, besides the interplanetary transfer time, we should allow sufficient time for the spacecraft to make the correction to be on the correct planetocentric orbit. This additional time, regarding the two-body transfer problem, can be important for low planet orbits. Seeing at the examples, the spacecraft needs about 30 days to escape the earth influence (SOI of about 1e6km), while it needs about 250 days to travel the interplanetary phase. Planetocentric phases are thus far from being negligible.

This problem does not appear though for swing-by, as swing-by only lasts but a few days.

7.3 GTOC3 Asteroid to Asteroid leg, with Automatic Swingby Design

The problem is the one of transferring a spacecraft from an asteroid to another, as specified in the GTOC 3 (see 3.1.2). The initial and final asteroids are given. The time of flight is fixed, and the dynamics include only the Sun and the Earth gravity.

In the reference cases, Earth swing-bys are used to reach the asteroid. Here, as our program allows, we do not impose any swing-by. But the gravitational potential of the Sun and the Earth are both considered.

The thruster has a 3000s Isp, and 0.15 N of thrust.

Table 7.6 is the definition of the transfer. It is a sub-trajectory of our solution trajectory.

Variable	Our tool (gradient)	T3D + Patched Conic Approx.
Launch date t_0	01/07/2020	
Time of flight T	551.5 days	
Departure Body	2001 GP2	
Arrival Body	1991 VG	
Initial mass m_0	1907.7 kg	
Final Mass m_f	1760.2 kg	1760.0 kg
Swing-by date	$\approx t_0 + 100 days$	$t_0 + 101.8 days$
Swing-by radius R	$\approx 199649 km$	176604.6 km

Table 7.6: GTOC3 asteroid - Earth - asteroid, Swing-by case comparison

On this example the initial guess used has no particular property. We solve the problem from scratch. Indeed, it was not possible to use the initial guess provided by T3D. The patched conic approximation provides an initial guess to our algorithm that crashes the spacecraft on Earth. This example proves the good convergence of our algorithm.

Our algorithm manages to find an Earth swing-by. This allows comparisons between our solution, and the solution we submitted for GTOC3[*]. They are summarised in table 7.6.

For these examples, and all examples where swing-bys can be expected, as the feasibility procedure of section 6.3.2 dictates, we should seek the constraint feasibility first. If the constraint-satisfied trajectory manages to get swing-byes, then the optimal trajectory necessarily keeps the swing-byes, if those are of benefit.

[*] Swing-by radiuses are estimated with the total gravity acceleration graph.

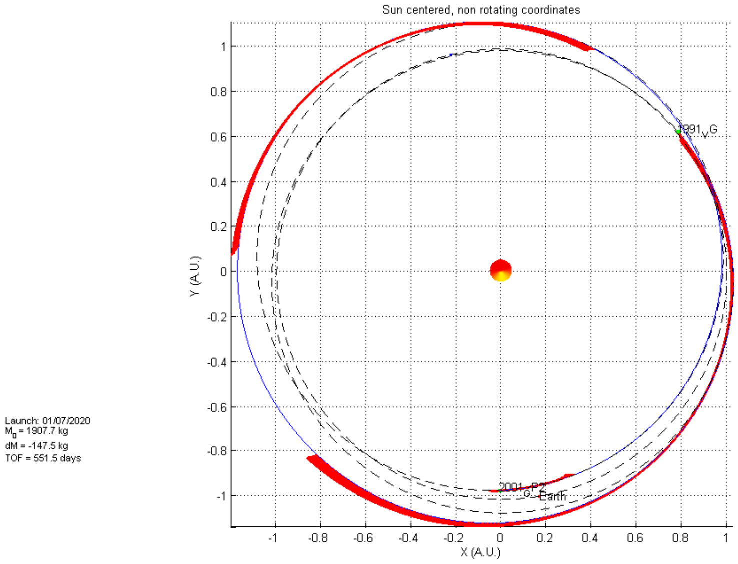

Figure 7.10: GTOC3: Asteroid to Asteroid transfer trajectory, with intermediate Earth Swing By

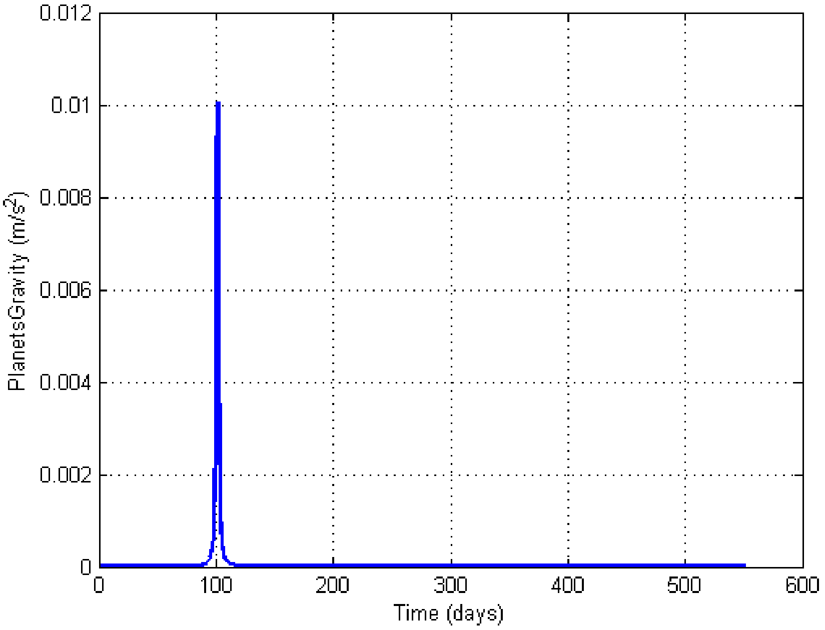

Figure 7.11: GTOC3: Asteroid to Asteroid transfer gravity acceleration, with intermediate Earth Swing By

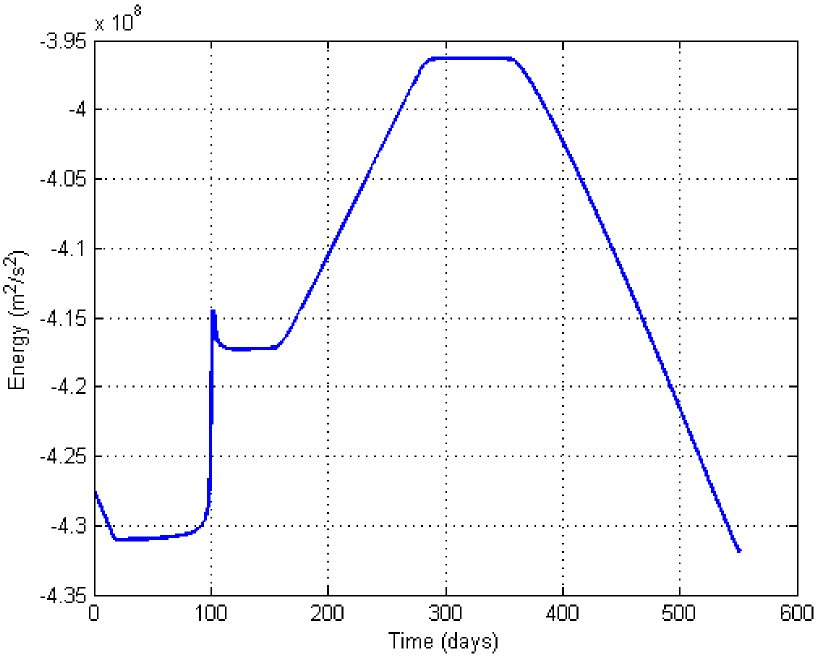

Figure 7.12: GTOC3: Asteroid to Asteroid transfer energy, with intermediate Earth Swing By

Figure 7.12 and 7.11 show the transfer energy and the gravity field measured by the spacecraft during the transfer. The gravity acceleration graph shows a very steep peek that matches the instant of Earth swing-by. In accordance, the energy graph shows a rapid increase in energy. These observations are typical of swing-by transfers, as they have been introduced in section 1.3.

Figure 7.14 displays the close approach to Earth. Because of the high altitude, and the hyperbolic excess velocity, the deviation is not very important (see equation 1.19).

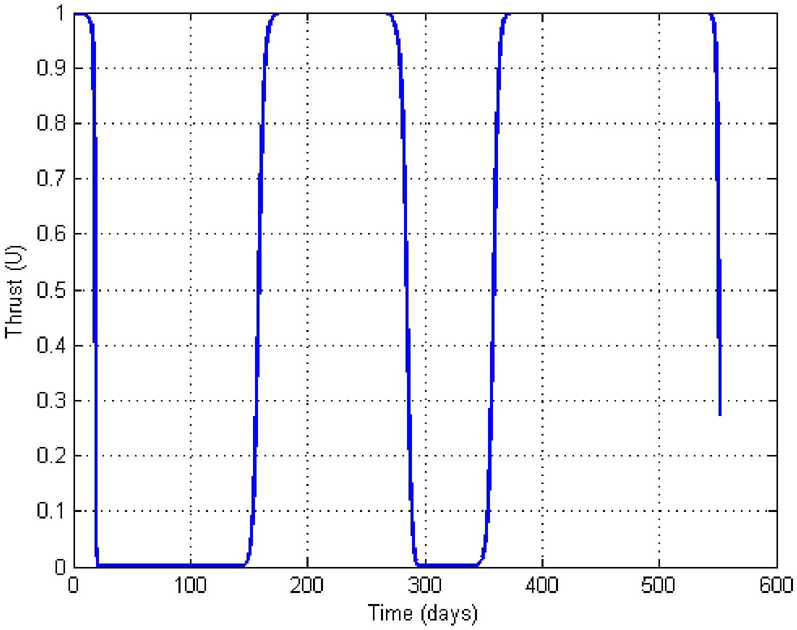

Figure 7.13: GTOC3: Asteroid to Asteroid transfer thrust, with intermediate Earth Swing By

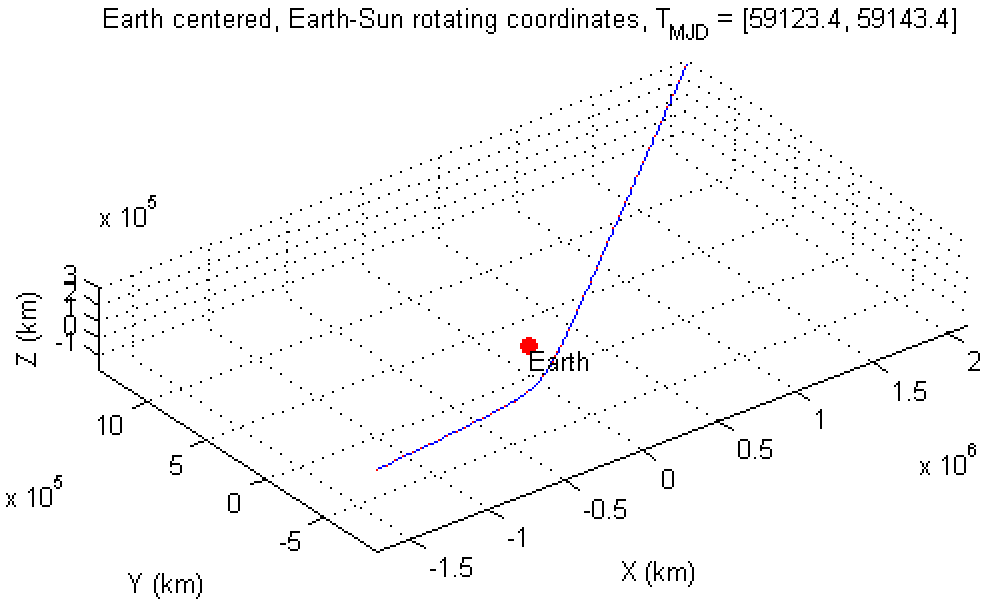

Figure 7.14: GTOC3: Earth Swing By

Conclusions

Le vrai point d'honneur n'est pas d'être toujours
dans le vrai. Il est d'oser, de proposer des idées
neuves, et ensuite de les vérifier. Il est aussi, bien
sûr, de savoir reconnaître publiquement ses
erreurs (...). L'honneur du scientifique est (...)
d'accepter de perdre la face

Pierre-Gilles de Gennes, Prix Nobel de Physique
1991

Thesis Summary

We provide 2 different, but somewhat complementary, approaches.

The first approach considers the patched conic approximation, and a low-thrust model to evaluate as quickly as possible different scenarii. We introduce a new low-thrust shape-based model, which includes a Keplerian parameterized function. This model improves on the current models, as the coast segments limit the control approximation regarding the optimal control. As a result, investigating a search space is quick and close to optimal possibilities.

We introduce a simplified computational procedure that permits, along with a pruning approach, to reduce the computational cost of evaluating different multi-gravity assist trajectories. We show that the complexity of using any parametrised model is polynomial with the number of phases. The approach can be easily parallelled.

Search for and evaluating different scenarii can then be done easily. However, besides the energetic tools presented in chapter 2, we did not find any new approaches that would help evaluating a scenario. Systematic approaches should be preferred to heuristics. Energetic tool and experience help in reducing the number of possibilities.

Often it is considered that determining the scenario demands to be able to find the global optimum of the problem. Indeed, among a perusal of literature locally optimal scenarii are never explicitely sought, while for "simple" direct transfer cases, the global optimality of solution is still an open problem. The second approach addresses then local optimal control methods, to seek locally optimal scenario.

We investigate current tools and methods, and try to demonstrate the limitation of direct and indirect methods for complex dynamical problems. A formulation is introduced to tackle difficult dynamical problems.

We try to solve problem with big variations and changes in the dynamics. This was a good benchmark to gravity assist trajectories. We successfully manage to optimise multi-gravity assist trajectories. The solver is able to automatically find the necessary swing-bys, however it is still a matter of luck and good initial guess.

Perspectives

The optimal control method is difficult in practice, even with robust optimisers.

The differences in scale introduce error of integration, as the mantissa of double numbers is limited. Further investigations should be made using more precise computer architectures to limit truncations error.

Also, efficient and more precise integrators should be studied for multi-scale multi-body dynamics. Some significant work seems to have been done in this field in molecular dynamics.

Any initial guess does not necessarily provide a solution with swing-by, if such solutions exist. In that scope, it should be possible to encourage the search of swing-by by modifying the dynamics. Some future work can address this problem, either by deforming the physics with a homotopy approach, or by imposing swing-bys as an initial guess and testing if the optimised transfer preserves them.

Bibliography

[AB07] K. Alemany and R.D. Braun. Survey of global optimization methods for low-thrust, multiple asteroid tour missions. In *AAS/AIAA Space Flight Mechanics Meeting*, 2007. [cited at p. 42]

[AG90] E. Allgower and K. Georg. *Numerical Continuation Methods.* Springer Verlag, Berlin-New York, 1990. [cited at p. 31]

[Bat01] R.H. Battin. *An Introduction to the Mathematics and Methods of Astrodynamics.* AIAA Education Series, 1 edition, 2001. [cited at p. 10, 13, 16, 26, 49, 50, 86, 146]

[BBS$^+$06] E.A Bering, M. Brukardt, J.P. Squire, T.W. Glover, V. Jacobson, and G. McCaskill. Recent improvements in ionization costs and ion cyclotron heating efficiency in the vasimr engine. In *44th AIAA Aerospace Sciences Meeting and Exhibit.* AIAA, AIAA, January 2006. [cited at p. 7]

[BC95] J.T. Betts and E.J. Cramer. Application of direct transcription to commercial aircraft trajectory optimisation. *Journal of Guidance, Control and Dynamics*, 18(1), 1995. [cited at p. 80]

[BCM] E. G. Birgin, R. A. Castillo, and J. M. Martínez. Numerical comparison of augmented lagrangian algorithms for nonconvex problems. [cited at p. 105]

[BE02] R. Bertrand and R. Epenoy. New smoothing techniques for solving bang-bang optimal control problems - numerical results and statistical interpretation. *Optimal Control: Applications and Methods*, pages 171 – 197, 2002. [cited at p. 83]

[Ber82] D.P Bertsekas. *Constrained optimization and Lagrange multiplier methods.* Academic press, New-York, 1982. [cited at p. 93, 104]

[Ber95] D.P Bertsekas. *Nonlinear Programming.* Athena Scientific, Massachusetts, 1995. [cited at p. 105]

[Ber01] R. Bertrand. *Optimisation de trajectoires interplanétaires sous hypothèses de faible poussée.* PhD thesis, Université Paul Sabatier, 2001. [cited at p. iii, 10, 41, 83, 85, 86, 88, 100]

[Bet] J.T Betts. *SOCS*. Boeing. [cited at p. 82]

[Bet98] J.T. Betts. Survey of numerical methods for trajectory optimization. *Journal of Guidance, Control and Dynamics*, 21(2), 1998. [cited at p. 79, 80, 88, 89]

[BGRC04] E. Barrabes, G. Gomez, and J. Rodrigez-Canabal. Gravitationnal assisted trajectories. Lecture notes, University of Barcelona, July 2004. [cited at p. 14, 16]

[BH75] A Bryson and Y.C. Ho. *Applied Optimal Control*. Hemisphere Publishing Corporation, new york edition, 1975. [cited at p. 86, 89, 93, 94]

[BH98] J. Betts and W. Huffman. Mesh refinement in direct transcription methods for optimal control. *Optimal Control Applications and Methods*, 19(6), 1998. [cited at p. 82]

[BHTR06] D. A. Benson, G. T. Huntington, T. P. Thorvaldsen, and A. V. Rao. Direct trajectory optimization and costate estimation via an orthogonal collocation method. *Journal of Guidance, Control and Dynamics*, 29(6), 2006. [cited at p. 81]

[BLV03] C. H. Bischof, B. Lang, and A. Vehreschild. Automatic differentiation for matlab programs. In *Proc. Appl. Math. Mech.* RWTH Aachen University, 2003. [cited at p. 60]

[BMN$^+$05] V.M. Becerra, D.R. Myatt, S.J. Nasuto, J.M. Bishop, and D. Izzo. An efficient pruning technique for the global optimisation of multiple gravity assist trajectories. In *Proceedings of Global Optimisation*, 2005. [cited at p. 37]

[Bol33] O. Bolza. *Vorlesungen Uber VariationsRechtung*. Druck und Verlag Von B.G.Teubner, 1933. [cited at p. 78]

[Bol91] F.W. Boltz. Orbital motion under continuous radial thrust. *Journal of Guidance Control and Dynamics*, 14(3):667–670, 1991. [cited at p. 49]

[Bol92] F.W. Boltz. Orbital motion under continuous tangential thrust. *Journal of Guidance Control and Dynamics*, 15(6):1503–1507, 1992. [cited at p. 49]

[CCW94] V. Coverstone-Carroll and S.N. Williams. Optimal low thrust trajectories using differential inclusion concepts. *Journal of the Astronautical Science*, 42(4):379 – 393, 1994. [cited at p. 81]

[Ceo98] T. Ceolin. *Optimisation of Impulsive Thrusts for the Calculation of Heliocentric Trajectories and Planetary Station Acquisition*. PhD thesis, Université Paul Sabatier, 1998. [cited at p. 28]

[DHV01] A. L. Dontchev, William W. Hager, and Vladimir M. Veliov. Second-order Runge–Kutta approximations in control constrained optimal control. *SIAM Journal on Numerical Analysis*, 38(1):202–226, 2001. [cited at p. 82]

[DM04] T. Dargent and V. Martinot. An integrated tool for low-thrust optimal control orbit transfers in interplanetary trajectories. In *Proceedings of the 18th International Symposium on Space Flight Dynamics*, October 2004. [cited at p. 115, 117]

[DT97] G.B. Dantzig and M.N Thapa. *Linear Programming: Theory and Extensions*. Springer Series in Operations Research. Springer edition, 1997. [cited at p. 105]

[EC92] P.J. Enright and B.A. Conway. Discrete approximations to optimal trajectories using direct transcription and nonlinear programming. *Journal of Guidance, Control, and Dynamics*, 15(4), 1992. [cited at p. 80]

[EFP04] F.W. Elliot, J.E. Foster, and M.J. Patterson. An overview of the high power electric propulsion (hipep) project. In *40th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*. AIAA, AIAA, July 2004. [cited at p. 7]

[Eng06] A. Engelsone. *Direct Transcription Methods in Optimal Control: Theory and Practice*. PhD thesis, North Carolina State University, May 2006. [cited at p. 82]

[FR01] F. Fahroo and I.M. Ross. Costate estimation by a legendre pseudospectral method. *Journal of Guidance, Control and Dynamics*, 24(2), 2001. [cited at p. 81]

[FR02] F. Fahroo and I.M. Ross. Direct trajectory optimization by a chebyshev pseudospectral method. *Journal of Guidance, Control and Dynamics*, 25(1), 2002. [cited at p. 81]

[Gao07] Y. Gao. Low-thrust interplanetary transfers, including escape and capture trajectories. *Journal of Guidance, Control and Dynamics*, 30(6), 2007. [cited at p. 79]

[Gay88] D. Gaylor. *Optimal Low-Thrust Trajectories for Planetary Capture*. PhD thesis, Air Force Institute of Technology, Dec 1988. [cited at p. 79]

[GFR08] Q. Gong, F. Fahroo, and I.M. Ross. Spectral algorithm for pseudospectral methods in optimal control. *Journal of Guidance, Control and Dynamics*, 31(3), 2008. [cited at p. 82]

[GMS08] P.E. Gill, W. Murray, and M. Saunders. *User's guide for SNOPT Version 7: Software for Large-Scale Nonlinear Programming*. Standford Optimization Laboratory, June 2008. [cited at p. 60, 89, 146]

[GP08] K. Graichen and N. Petit. Constructive methods for initialization and handling mixed state-input constraints in optimal control. *Journal of Guidance, Control, and Dynamics*, 31(5):1334–1343, 2008. [cited at p. 88]

[HC96] A.L. Herman and B.A. Conway. Direct optimization using collocation based on high-order gauss-lobatto quadrature rules. *Journal of Guidance, Control and Dynamics*, 19(3), 1996. [cited at p. 81]

[HMG04] T. Haberkorn, P. Martinon, and J. Gergaud. Low-thrust minimum-fuel orbital transfer: A homotopic aproach. *Journal of Guidance, Control and Dynamics*, 27(6), November-December 2004. [cited at p. 31]

[Hul97] D.G. Hull. Conversion of optimal control problems into parameter optimization problems. *Journal of Guidance, Control and Dynamics*, 20(1), 1997. [cited at p. 80]

[Hul03] D. G. Hull. *Optimal Control Theory for Applications.* Mechanical engineering Seris. Springer, 2003. [cited at p. 81, 82]

[Hun07] G.T. Huntington. *Advancement and Analysis of a Gauss Pseudospectral Transcripton for Optimal Control Problems.* PhD thesis, Massachusetts Institute of Technology, June 2007. [cited at p. 81]

[IBM$^+$06] D. Izzo, V.M. Becerra, D.R. Myatt, S.J. Nasuto, and J.M. Bishop. Search space pruning and global optimisation of multiple gravity assist spacecraft trajectories. *Journal of Global Optimisation*, 38(2):283–296, June 2006. [cited at p. 37]

[Izz06] D. Izzo. Lambert's problem for exponential sinusoids. *Journal of Guidance Control and Dynamics*, 29(5):1242–1245, 2006. [cited at p. 45]

[Jar75] B. A Jarmark. Convergence control in diffenretial dynamic programming applied to air-to-air combat. *AIAA Journal*, 14(1), 1975. [cited at p. 103]

[Jar83] B. A Jarmark. Missile duel between two aircraft. *Journal of Guidance*, 8(4), 1983. [cited at p. 103]

[JC02] G. Robert Jahn and Edgar Y. Choueiri. *Encyclopedia of Physical Science and Technology*, volume 5, chapter Electrical Propulsion. Academic Press, 2002. [cited at p. 5]

[JCGK01] R. Jehn, S. Campagnola, D. Garcia, and S. Kemble. Low thrust approach and gravitational capture at mercury. In *18th international Symposium on Space Flight Dynamics*, 2001. [cited at p. 79]

[JM70] D.H. Jacobson and D.Q. Mayne. *Differential Dynamic Programming.* Elsevier, New York, 1970. [cited at p. 92, 99]

[Joh69] F.T. Johnson. Approximate finite thrust trajectory optimization. *AIAA Journal*, 7(6):993 – 997, 1969. [cited at p. 10]

[KE95] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proc. IEEE Intl. Conf. On Neural Networks*, 1995. [cited at p. 29]

[Kea97] R.B. Kearfott. Interval computations: Introduction, uses, and ressources. 1997. [cited at p. 31]

[Kel99] C.T Kelley. *Iterative Methods for Optimization.* SIAM, 1999. [cited at p. 103]

[KLMR01] W. S. Koon, M.W. Lo, J. E. Marsden, and S.D. Ross. Low energy transfer to the moon. *Celestial Mechanics and Dynamical Astronomy*, 81(1):63–73, 2001. [cited at p. 16]

[Klu02] C.A. Kluever. Optimal earth-capture trajectories using electric propulsion. *Journal of Guidance, Control and Dynamics*, 25(3), 2002. [cited at p. 79]

[KLW06] Y. Kawajir, C. Laird, and A. Wachter. *Introduction to IPOPT: A tutorial for downloading, installing, and using IPOPT*, November 2006. [cited at p. 89]

[KMLR02] W.S. Koon, J.E. Marsden M.W. Lo, and S.D. Ross. Constructing a low energy transfer between jovian moons. *Contemporary Mathematics*, 292:129–145, 2002. [cited at p. 16]

[KP95] C.A. Kluever and B.L. Pierson. Optimal low-thrust three-dimensional earth-moon trajectories. *Journal of Guidance, Control and Dynamics*, 18(4), July 1995. [cited at p. 79]

[Law92] D.F. Lawden. Calculation of singular extremal rocket trajectories. *Journal of Guidance, Control and Dynamics*, 15(6), 1992. [cited at p. 28]

[LP] Torbjörn Larsson and Michael Patriksson. An augmented lagrangean scheme for capacitated traffic assignment problems. [cited at p. 105]

[LT98] R.M Lewis and V. Torczon. A globally convergent augmented lagrangian pattern search algorithm for optimization with general constraints and simple bounds. ICASE 98-31, NASA, August 1998. [cited at p. 105]

[LVBBT07] J. Laurent-Varin, J.F. Bonnans, N. Berend, and C. Talbot. Interior-point approach to trajectory optimization. *Journal of Guidance, Control and Dynamics*, 30(5), 2007. [cited at p. 82]

[MC64] J.E. McIntyre and L. Crocco. Optimal transfer between close cirbular orbits using the pontryaguin maximum principle. January 1964. [cited at p. 113]

[MC95] N. Markopoulos and A.J. Calise. Analytical investigations in aircraft and spacecraft trajectory optimization and optimal guidance. Technical Report 4672, NASA, May 1995. [cited at p. 45, 56]

[MS62] R. Murray Smith. *Switching and Learning in Feedback Systems, State of the Art Survey*, volume 3355 of *Lecture Notes in Computer Science*. Springer,, September 1962. European Summer School on Multi-Agent Control, Maynooth, Ireland, September 8-10, 2003, Revised Lectures and Selected Papers. [cited at p. xiv, 89]

[MS63] W.G Melbourne and C.G Sauer. Optimum interplanetary rendezvous trajectories with power limited vehicles. *AIAA Journal*, 1(1), 1963. [cited at p. 10]

[MS65] W.G Melbourne and C.G. Sauer. Performance computations with pieced solutions of planetocentric and heliocentric trajectories for low thrust missions. *JPL Space Programs Summary*, 4, 1965. Jet Propulsion Lab, No. 37-36. [cited at p. 10]

[MW02] J.K. Miller and C.J. Weeks. Application of tisserand's criterion to the design of gravity assist trajectories. In *AIAA/AAS Astrodynamics Specialist Conference*, Aug 2002. [cited at p. 16, 24]

[Mya03] D. R. Myatt. Advanced global optimisation for mission analysis and design. ARIADNA study 04/4101, ESA, 2003. [cited at p. 28]

[Neu04] A. Neumaier. Complete search in continuous global optimisation and constraint satisfaction. *Acta Numerica*, 2004. [cited at p. 29]

[NVB01] R. Nah, S.R. Vadali, and E. Braden. Fuel-optimal, low-thrust, three-dimensional earth-mars trajectories. *Journal of Guidance, Control and Dynamics*, 24(6), 2001. [cited at p. 79]

[Oly07a] J. T. Olympio. Global trajectory optimisation: Can we prune the solution space when considering deep space maneuvers? ARIADNA study 06/4101, ESA, 2007. [cited at p. v, 28, 32, 87]

[Oly07b] J. T. Olympio. Mixed thrust - coast arc model for interplanetary gravity assist trajectory. In *58th IAC*, 24-28 sept. 2007. [cited at p. v, 46]

[Oly08] J. T. Olympio. Algorithm for low-thrust optima interplanetary transfers with escape and capture phases. In *AIAA Astrodynamic Specialist Conference*, 18-21 Aug. 2008. [cited at p. v, 119]

[Pet02] A.E. Petropoulos. A review of some exact solutions to the planar equations of motion of a thrusting spacecraft. In *2nd International Symposium Low Thrust Trajectories*, Toulouse, France, 2002. [cited at p. 45, 46]

[PK94] B.L Pierson and C.A. Kluever. Three stage approach to optimal low-thrust earth moon trajectories. *Journal of Guidance Control and Dynamics*, 17(6):1275 – 1282, 1994. [cited at p. 10]

[PL01] A.E. Petropoulos and J.M. Longuski. A shape-based algorithm for the automated design of low thrust, gravity assist trajectories. In *AAS/AIAA Astrodynamics Specialists Conference*, 2001. [cited at p. 45, 46]

[PLB00] A.E. Petropoulos, J.M. Longuski, and E.P. Bonfiglio. Trajectories to jupiter via gravity assists from venus, earth, and mars. *Journal of Spacecraft and Rockets*, 37(6), 2000. [cited at p. 23, 46]

[Pon53] L.S. Pontryaguin. *Mathematical Theory of Optimal Process*. 1 edition, 1953. [cited at p. 84, 85]

[Pru79] J.E. Prussing. Geometrical interpretation of the angles α and β in the lambert's problem. *Journal of Guidance*, 2(5), 1979. [cited at p. 26]

[PS02] J.E. Prussing and Suzannah L. Sandrik. Second-order necessary and sufficient conditions applied to continuous-thrust trajectories. Monterey, CA, Aug 2002. [cited at p. 102]

[PV95] K.E. Parsopoulos and M.V. Vrahatis. Particle swarm optimization method for constrained optimization problems. 1995. [cited at p. 29]

[RFRR04] M.D. Rayman, T.C. Fraschetti, C.T. Russel, and C.A. Raymond. Dawn: A mission in development for exploration of main belt asteroids vesta and ceres. In *55th International Astronautical Congress*, 2004. [cited at p. 6, 67]

[RFRR05] M.D. Rayman, T.C. Fraschetti, C.A. Raymond, and C.T. Russel. Preparing for the dawn mission to vesta and ceres. In *56th International Astronautical Congress*, October 2005. [cited at p. 6]

[RO08] C. L. Ranieri and C. A. Ocampo. Indirect optimization of two-dimensional finite burning interplanetary transfers including spiral dynamics. *Journal of Guidance, Control and Dynamics*, 31(3), July 2008. [cited at p. 79]

[Ros03] I.M. Ross. *User's Manual for DIDO: A MATLAB Application Package for Solving Optimal Problems.* Naval Postgraduate School, September 2003. [cited at p. 82]

[Rus07] R.P. Russel. Primer vector theory applied to global low-thrust trade studies. *Journal of Guidance, Control and Dynamics*, 30(2), 2007. [cited at p. 79, 83]

[Sau97] C.G. Sauer. Solar electric performance for medlite and delta class planetary missions. In *AAS/AIAA Astrodynamics Specialist Conference*, 1997. [cited at p. 67]

[SC94] W.A. Scheel and B.A. Conway. Optimization of very low-thrust, many revolution spacecraft trajectories. *Journal of Guidance, Control, and Dynamics*, 17(6), 1994. [cited at p. 80]

[Sey94] H. Seywald. Trajectory optimisation based on differential inclusion. *Journal of Guidance, Control and Dynamics*, 17(3), 1994. [cited at p. 81]

[SF99] J.A. Sims and S. Flanagan. Preliminary design of low thrust interplanetary missions. In *AAS/AIAA Astrodynamics Specialist Conference*, 1999. [cited at p. 46]

[SKL$^+$02] R. Serban, W.S. Koon, M.W. Lo, J.E. Marsden, L.R. Petzold, S.D. Ross, and R.S. Wilson. Halo orbit mission correction maneuvers using optimal control. *Automatica*, 38(4):571–583, April 2002. [cited at p. 3]

[SL02] N.J. Strange and J.M. Longuski. Graphical method for gravity assist trajectory design. *Journal of Spacecraft and Rockets*, 39, "" 2002. [cited at p. 24]

[SPC97] A. Schwartz, E. Polak, and Y. Chen. *RIOTS95: A Matlab. Toolbox for Solving Optimal Control Problems, Version. 1.0*, May 1997. [cited at p. 82]

[Tad04] S. Tadashi. *A Study of Variable Thrust, Variable Specific Impulse trajectories for Solar System Exploration.* PhD thesis, Georgia Institute of Technology, December 2004. [cited at p. xi, 13, 83]

[TC95] S. Tang and B.A. Conway. Optimization of low-thrust interplanetary trajectories using collocation and nonlinear programming. *Journal of Guidance, Control and Dynamics*, 18(3), 1995. [cited at p. xii, 80, 81]

[Tre07] E. Trelat. *Commande Optimale, notes de cours.* 2007. [cited at p. 85, 99, 104]

[TZ88] A. Torn and A. Zilinskas. *Global Optimisation.* 1 edition, 1988. [cited at p. 29]

[VA03] J. A. Van Allen. Gravitational assist in celestial mechanics - a tutorial. *American Journal of Physics*, 71(5), may 2003. [cited at p. 11]

[VD06] M. Vasile and P. DePascale. Preliminary design of low thrust multiple gravity assist trajectories. *AIAA, Journal of Spacecraft and Rockets*, 43(5):1065–1076, 2006. [cited at p. 45]

[VM97] D.A. Vallado and W.D. McClain. *Fundamentals of Astrodynamics and Applications*. McGraw-Hill, 1 edition, 1997. [cited at p. 26]

[VN00] S.R. Vadali and R. Nah. Fuel optimal planer earth-mars trajectories using low-thrust exhaust modulated propulsion. *Journal of Guidance, Control and Dynamics*, 23(3), 2000. [cited at p. 79]

[VSD05] M. Vasile, L. Summerer, and P. DePascale. Design of earth-mars transfer trajectories using evolutionary branching technique. *Acta Astronautica*, 56:705 – 720, 2005. [cited at p. 39]

[VSJ05] M. Vasile, O. Schütze, and O. Junge. Spiral trajectories in global optimisation of interplanetary and orbital transfers. Technical Report ARIADNA AO05/4106, ESA, 2005. [cited at p. 45]

[WB06] A. Wächter and L. T. Biegler. On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25 – 57, 2006. [cited at p. 81]

[Whi01] G.J. Whiffen. Application of a novel optimal control algorithm to low thrust trajectory optimization. In *AAS/AIAA Space Flight Mechanics Meeting*, 2001. [cited at p. 79, 91]

[WS02] G.J. Whiffen and J.A. Sims. Application of the sdc optimal control algorithm to low thrust escape and capture including fourth body effects. In *2nd International Symposium on Low Thrust Trajectories*, Toulouse, France, June 18–20 2002. [cited at p. 79]

[WT72] W.E. Williamson and B.D. Tapley. Ricatti transformations for control optomization usiing second variation. *IEEE Transactions on Automatic Control*, 17(3), june 1972. [cited at p. 93, 102]

[ZO05] S. Zimmer and C. Ocampo. Analytical gradients for gravity assist trajectories using constant specific impulse engines. *Journal of Guidance, Control and Dynamics*, 28(4), July 2005. [cited at p. 86]

Appendices

Planets' numerical data

Name	Radius (km)	Excentricity	Period (days)	Avg. Dist. from SUN (AU)	μ (10^{14} $m^3 s^{-2}$)
Mercury ☿	2439.7	0.20563	87.969	0.38	0.22032
Venus ♀	6051.8	0.00677	224.701	0.72	3.24858
Earth ♁	6378.14	0.01671	365.25	1.0	3.98600
Mars ♂	3402.45	0.09341233	779.96	1.52	0.428283
Jupiter ♃	71492	0.0483926	4335.35	5.2	1266.86537
Saturn ♄	60268	0.0541506	10757.74	9.53	37931284.5
Uranus ⛢	25656	0.0471677	30708.16	19.2	57.93947
Neptun ♆	24961	0.0085858	60224.9	30.07	68.35107
Pluto ♇	2300	0.25	90613.3	39.48	0.00870

Planets' position is given by DE405 JPL ephemeris database.

$AU = 1.49597870691\ 10^{11}\ m$
$\mu_{SUN} = 1.32712440018\ 1^{20}\ m^3/s^2$
$G = 6.67259e - 11\ m^3/s^2/kg$
$T_{sideral} = 365.25636\ days$

List of Symbols and Abbreviations

Symbol Name	Description	Unit
a	semi major axis	m
c	Exhaust velocity	m/s
e	eccentricity	
E	Orbital Energy	m^2/s^2
F	Thrust force	N
f	dynamic function	
H	Hamiltonian	
H	angular momentum vector	m^2/s
h	angular momentum	m^2/s
I_{Sp}	Specific Impulse	s
J	Performance index	m/s
M	mass	kg
m	spacecraft mass	kg
P	power	W
p	semi latus rectum	m
q	mass flow rate	kg/s
r	radius	m
t	Time, Epoch(MJD2000)	
u	control vector	
V_∞	Hyperbolic excess velocity	km/s
Vx,Vy,Vz	Cartesian velocity components	m/s
x	state vector	
x,y,z	Cartesian position components	m
μ	gravitational constant	m^3/s^2
δ	swing-by angular deviation	rad
ΔV	Velocity increase	m/s
Λ	Co-state vector	
ν	Lagrange multiplier for constraints	

Constant Name	Description	Value
AU	Astronomical Unit	$1.49597870691 10^{11}$ m
g_0	Gravity acceleration constant	9.80665 m/s^2
G	Universal gravitational constant	6.67259 10^{-11} $m^3/s^2/kg$
μ_{SUN}	Sun gravitational constant	1.32712440018 10^{20} m^3/s^2

Abbreviation	Description	Definition
AIAA	American Institute of Aeronautics and Astronautics	
BC	Boundary Condition	
BVP	Boundary Value Problem	page 103
CNES	Centre National d'Etudes Spatiales (France)	
COV	Calculus of Variation	
DE405	JPL ephemeris data	
DRO	Distant Retrograde Orbit	
DSM	Deep Space Manoeuvre	page 27
ESA	European Space Agency	
JPL	Jet Propulsion Laboratory	
MGA	Multi Gravity Assist	page 28
MJD	Modified Julien Date	
MJD2000	Modified Julien Date with reference epoch Jan 1st, 2000 at 12:00:00	
NASA	National Aeronautics and Space Administration (USA)	
NEA	Near Earth Asteroids	
ODE	Ordinary Differential Equation	
OCP	Optimal Control Problem	page 79
POP	Parameter Optimisation Problem	page 25
R3BP	Restricted 3 Body Problem	page 79
SNOPT	Sparse Nonlinear OPTimizer [GMS08]	
SOI	Sphere Of Influence [Bat01]	page 10
SQP	Sequential Quadratic Programming	page 61
TPBVP	Two Point BVP	page 83

Index