OPTIMIZATION OF LOW-THRUST GRAVITY-ASSIST TRAJECTORIES UNDER APPLICATION OF TISSERAND'S CRITERION

Vom Fachbereich Produktionstechnik

der

UNIVERSITÄT BREMEN

zur Erlangung des Grades
Doktor-Ingenieur
genehmigte

Dissertation

von

Dipl.-Ing. Volker Maiwald

Gutachter: Prof. Dr.-Ing. Andreas Rittweger
Fachbereich Produktionstechnik,
Universität Bremen

Prof. Dr.-Ing. Bernd Dachwald
Fachbereich Luft- und Raumfahrttechnik,
Fachhochschule Aachen

Tag der mündlichen Prüfung: 23. März 2018

I think we are going… because it's in the nature of the human being to face challenges.

- Neil Armstrong

Acknowledgements

My gratitude goes towards my colleagues and friends from the System Analysis Space Segment Department for their support and fellowship. I thank especially Dominik Quantius, Andy Braukhane, Daniel Schubert and Conrad Zeidler for their discussions, support and proof reading concerning my dissertation. I also thank Dr. Marco Sharringhausen, who always helped me whenever I had questions about mathematics and Dr. Wolfgang Seboldt for stepping up on a short notice during the final stages of this thesis and providing much appreciated insight and advice.
I thank my parents, Dr. Ursula and Werner Maiwald for enabling me to follow my dreams – I did, thank you! – and my brothers Rüdiger and Tobias for their encouragement.
And of course, I thank my incredible wife Inga who patiently endured my struggles with my work and supported me so much, despite the bumpy road far off an optimal path.

Revisions for Publication

This is a revised edition of the original dissertation. The changes have been made to allow a more self-enclosed reading of the dissertation. Next to editorial changes to address language and typing errors, the following changes have been made to the originally submitted version:

1) The paper by Chen, Kloster and Longuski (2008) has been included in the discussion of Chapters 2.4.2, 6 and 7.11.4.
2) The content about Simulated Annealing in Chapter 5 and its application in Chapter 7 have been expanded for clarification.
3) A thorough explanation about the shape-based trajectory model has been included in Chapter 5, to erase the need for lecture of the respective paper by Wall and Conway (2009).
4) A depiction of an example trajectory (as given in Fig. 5-4) in conjunction with Tisserand Graphs has been added, including a discussion, and can be found in Chapter 7.12

Abstract

Exploring the solar system is becoming more and more challenging and the associated space missions are becoming similarly more demanding concerning Δv, targeted at distant bodies. Recent examples for such missions are *New Horizons*, which visited the dwarf planet Pluto and *Rosetta's* rendezvous with the comet 67P/Churyumov-Gerasimenko. These missions would not have been possible without the application of gravity assists, a technique already used for *Pioneer 10* and *11* and the *Voyager* missions. It is based on transferring energy between a planet or other flyby partner and a spacecraft, which results in trajectory changes without associated propellant consumption.

Low-thrust propulsion, which is usually operating continuously for significant amounts of time during a mission, is very efficient and thus often acts as mission enabler. The effort - expressed in used propellant mass fraction – to achieve a certain mission goal is usually smaller for low-thrust propulsion than for a similar mission applying chemical propulsion. This is due to the large specific impulse (typically >3000s) associated with low-thrust propulsion. To further enhance our ability to explore the solar system it is therefore desirable to combine these two techniques.

Currently, a number of methods exists to optimize low-thrust trajectories and these have been applied to gravity-assist scenarios as well. However, up to now a method for optimization of the actual sequence of gravity- assist partners is not publically available. Instead, the respective mission analyst has provided the sequence, based on assumptions, estimates and experience.

The major goal of this dissertation is to close this gap and investigate if it is possible to reduce the amount of expert information needed and for optimization and thus increase the degree to which the search space is searched in completeness. Reducing the amount of involved experience and trajectory resp. sequence information, improves the chance of finding unforseen and worthwhile new mission sequences. This dissertation provides an analysis of how to incorporate the gravity-assist sequence into the optimization process and thus allow a complete, thorough and effective search of useful mission scenarios.

First, it is analyzed how gravity-assist sequences are obtained for impulsive missions and whether the same technique can be similarly applied to low-thrust scenarios. A typical approach for gravity-assist planning of impulsive mission scenarios is the application of Tisserand's Criterion. It is an energy relation, derived for observations of comets in the 19th century within the context of the circular restricted three-body system. This criterion can be represented visually in the form of so-called Tisserand graphs, which are used to map possible gravity-assist sequences. Furthermore, two sources of errors are analyzed: First, the error is analysed, which is caused by applying this relation to the actual solar system and thus diverting from the circular restricted three-body system. Second, the error is analysed, which is caused by adding thrust into the equations of motion. It is shown that the thrust force is negligible in the balance of forces for realistic mission designs. However, the thrust's effect on the specific orbit energy cannot be neglected.

Next, a correction term to apply Tisserand's Criterion in a low-thrust situation is derived. This term however, does not allow an a priori evaluation of the mission sequence and therefore the modified Tisserand's Criterion cannot be used to map out possible paths of gravity assists similarly as the unmodified Tisserad's Criteron for impulsive missions.

A method, based on a heuristic search, a shape-based trajectory model and the inclusion of possible gravity-assist partners as variable, is set up. The heuristic search circumvents the usage of Tisserand's Criterion. As a next step to establish the method, the respective variable structure is analyzed. Some of the relevant trajectory variables are shown to be interdepent. For instance, the flight times of the individual trajectory legs between all encountered bodies have to add up to the

total mission flight time. Consequently, two types of variables are defined: global and local. The first are independent and thus can be used by an evolutionary algorithm to evolve the mission candidates into better (based on solution fitness) solutions during the search. The latter are interdependent.

A number of example calculations are conducted to assess the success and usefulness of applying this search method. It is shown that for a single gravity assist in an Earth to Jupiter mission the obtained results are reliable concerning the gravity-assist partner and date and successfully improve the non-gravity assist benchmark by more than 20%. Performance for multi-gravity-assist missions is worse, but still solution improvement and a dominance of the global variables can be observed. Further calculations introduce constraints based on the variable regions of maximum Δv change and show that the solution quality and reliability can thus be improved. Furthermore, the limitations of reproducing existing trajectories are shown and discussed.

Finally, the drawbacks of the applied shape-based trajectory model are expressed in a discussion concerning its relation to Tisserand graph diagrams. It is recommended to use a different trajectory model to address these drawbacks in future work.

Zusammenfassung

Die Herausforderung das Sonnensystem zu erforschen wird immer größer und die damit verbundene Planung und Durchführung von Raumfahrtmissionen entsprechend anspruchsvoller. Dies ist auch dadurch bedingt, dass die Ziele solcher Missionen immer schwerer zu erreichen sind. *New Horizons*, die einen Vorbeiflug am Zwergplaneten Pluto durchgeführt hat und *Rosetta*s Rendezvous mit dem Kometen 67P/Tschurjumow-Gerasimenko sind zwei aktuelle Beispiele für solche anspruchsvollen Missionen. Ohne die Anwendung von sogenannten Schwungholmanövern wären sie nicht möglich gewesen. Diese Methode ist bereits bei den berühmten Missionen *Pioneer 10* und *11* angewendet worden und ebenso während des *Voyager*-Programms. Sie beruht auf Energieaustauch zwischen Planet und Raumfahrzeug. Dieser Energieaustausch ermöglicht Bahnänderungen ohne Treibstoffaufwand und kann so dazu beitragen, dass eine Mission überhaupt durchführbar wird.

Sogenannte Niedrigschubtriebwerke, die üblicherweise über lange Zeitabschnitte einer Mission kontinuierlich betrieben werden, stellen eine effiziente Möglichkeit zur Bahnänderung dar. Um eine bestimmte Mission durchführen zu können, ist der Treibstoffaufwand üblicherweise kleiner als für vergleichbare Mission, die mit chemischen, d.h. impulsiven Triebwerken geflogen werden. Dies ergibt sich aus dem hohen spezifischen Impuls (typischerweise > 3000s) von Niedrigschubtriebwerken.

Um also die Erforschung des Sonnensystems voranzutreiben, ist es ratsam den Einsatz von Niedrigschubtriebwerken mit Schwungholmanövern zu kombinieren. Zur Optimierung von Niedrigschubbahnen existieren verschiedene Methoden und diese wurden auch bereits für Schwungholszenarien bei solchen Missionen eingesetzt. Allerdings ist bisher keine Methode veröffentlicht, bei der die Sequenz der Schwungholpartner ebenfalls Teil der Optimierung ist. Die entsprechenden Himmelskörper sind üblicherweise eine Vorgabe für die Optimierung der jeweiligen Bahn, basierend auf Annahmen und Erfahrung der Missionsanalysten.

Das grundlegende Ziel dieser Dissertation ist es, diese Lücke zu schließen und zu untersuchen, ob es möglich ist, die Menge an Experteninformationen und Annahmen, die in die Optimierung einfließen, zu reduzieren und so den Suchraum umfassend zu durchsuchen. Dadurch sollte es möglich sein, unvorhergesehene, neue und nützliche Missionssequenzen aufzuspüren. Daher wird in dieser Dissertation analysiert, wie die Schwungholsequenz Teil der Optimierung werden kann. Als erster Schritt wird dafür überprüft, wie solche Sequenzen üblicherweise für impulsive Missionen bestimmt werden und ob dieser Ansatz auch im Niedrigschubfall angewendet werden kann.

Ein typisches Verfahren für die Planung von Schwungholmanövern ist die Anwendung des Tisserandkriteriums. Diese Energieformulierung, die ursprünglich für die Nutzung bei Kometenbeobachtungen im 19. Jahrhundert im Kontext des eingeschränkten Dreikörperproblems entwickelt wurde, wird häufig in Form von sogenannten Tisserandgraphen umgesetzt. Mit diesen lässt sich eine Sequenz grafisch planen. In dieser Arbeit wird analysiert, in welchem Verhältnis die mathematischen Abweichungen bei Anwendung des Tisserandkriteriums im realen Sonnensystem zu denen bei Anwendung im eingeschränkten Dreikörperproblem stehen. Ebenso werden diese Abweichungen mit denen verglichen, die durch die Einführung einer Schubbeschleunigung hervorgerufen werden. Dabei wird gezeigt, dass der Einfluss des Schubs auf die Kräftebilanz vernachlässigbar ist, seine Auswirkungen auf die spezifische Bahnenergie des Systems aber nicht.

Folglich wird ein Korrekturterm hergeleitet, der die Energieauswirkungen bei der Anwendung im Niedrigschubfall berücksichtigt. Dieser Term verhindert allerdings mögliche Sequenzen a priori zu

bewerten und ist daher nicht geeignet diese im Vorfeld in gleicher Weise zu planen, wie es für impulsive Missionen der Fall ist.

Mit diesen Erkenntnissen wird eine Methode aufgestellt, die auf einer heuristischen Suche, einem formbasierten Trajektorienmodell und der Berücksichtigung von möglichen Schwungholpartnern als Optimierungsvariable fußt. Die entsprechende Variablenstruktur wird analysiert und Abhängigkeiten der Variablen untereinander herausgestellt. So müssen z.B. die Flugzeiten für die einzelnen Segmente der Mission zusammen die Gesamtmissionsdauer ergeben. Diese Abhängigkeiten machen es notwendig zwei Arten von Variablen zu definieren: globale und lokale. Erstere sind unabhängig voneinander und daher geeignet, Lösungskandidaten während der Suche mit Evolutionären Algorithmen weiterzuentwickeln, um bessere (basierend auf der Fitness-Funktion) Lösungen zu erhalten.

Anhand von Beispielrechnungen wird gezeigt, dass die vorgeschlagene Methode geeignet ist, um ein einzelnes Schwungholmanöver für Missionen von der Erde zum Jupiter zu optimieren und Zeitpunkt, Ort und Partner zu definieren. Die Verbesserungen im Vergleich zu einer Mission ohne Schwungholmanöver sind größer als 20%. Zwar ist die Lösungsqualität für mehrfache Schwungholmanöver unzureichend, allerdings kann dennoch eine Verbesserung der Lösungen im Laufe der Optimierung festgestellt werden. Daraus lässt sich eine gewisse Dominanz der globalen Variablen ableiten. In weiteren Rechnungen wird der Einfluss von Beschränkungen des Suchraums auf die Lösungsqualität untersucht. Dabei wird eine leichte Verbesserung der Qualität und vor allem Zuverlässigkeit nachgewiesen. Außerdem werden die Einschränkungen bzgl. der Reproduktion von existierenden Flugbahnen gezeigt und diskutiert.

Abschließend wird anhand einer Darstellung der jeweiligen Trajektorien in Tisserand-Graphen-Diagrammen diskutiert welche Nachteile das verwendete formbasierte Trajektorienmodell aufweist und empfohlen für mögliche zukünftige Arbeiten ein anderes Trajektorienmodell zu verwenden.

Table of Contents

List of Figures

List of Tables

List of Symbols, Constants and Abbreviations

Latin Symbols

a	Semi-major axis	M	Mass of the primary body
E	Energy	n	Angular velocity
e	Eccentricity	P	Linear momentum
f	Function	$\vec{R}$	Position vector
H	Hamiltonian	r	Radius of the position
h	Specific angular momentum	T	Orbital period; flight time; Thrust
i	Inclination; number of iteration	t	Element of time interval; time variable
J	Objective function	U	Control strategy; Pseudo Potential
L	general function of the instantaneous state, control and time; Misc. abbreviation for the derivation of formulas	v	Velocity
		$\vec{x}$	State vector
m	Mass		

Greek Symbols

α	Thrust angle direction in shape-based model	δ	Deflection angle of a gravity-assist maneuver
Δ	Difference (e.g. in energy)	ε	Specific orbit energy

Υ	Universal Gravitational Constant (see Constants); flight path angle for shape-based model
μ	Gravitational parameter
μ^*	Reduced mass
Φ	Function of the final state and time
τ	expected mean value of control variable for Simulated Annealing
Θ	The angle swept over in the course of the trajectory in the shape-based model
ω	Angular velocity
ξ	Angle of direction of the outgoing heliocentric velocity
ζ	Angle of direction of the planetary, heliocentric velocity

Abbreviations

DE	Differential Evolution
EA	Evolutionary Algorithm
ENC	Evolutionary Neurocontroller
GA	Gravity Assist; Genetic Algorithm
GTOC	Global Trajectory Optimization Competition
ICRF	International Celestial Reference Frame
MP	Match Point (between trajectory segments)
MJD	Modified Julian Date
SOI	Sphere of Influence
SQP	Sequential Quadratic Programming

Constants and Units

AU	Astronomical Unit = 149,597,870,700 m (IAU, 2012)
Υ	Universal Gravitational Constant = $6.67408 \cdot 10^{-11}$ $m^3/(kg \cdot s^2)$ (NIST, 2015)
g_0	Standard gravity acceleration at sea leve = 9.80665 m/s (BIPM, 2006)
m_S	Mass of the Sun = 1.98911×10^{30} kg (Murray and Dermott, 2008)
m_J	Mass of Jupiter = 1.8986×10^{27} kg (Murray and Dermott, 2008)
μ_S	Gravitational parameter of the Sun = $2.958718364 \times 10^{-4}$ AU^3/d^2
μ_{Me}	Gravitational parameter of Mercury = $4.882888633 \times 10^{-11}$ AU^3/d^2
μ_V	Gravitational parameter of Venus = $7.246295917 \times 10^{-10}$ AU^3/d^2
μ_E	Gravitational parameter of Earth = $8.873925462 \times 10^{-10}$ AU^3/d^2
μ_M	Gravitational parameter of Mars = $9.542814315 \times 10^{-11}$ AU^3/d^2
μ_J	Gravitational parameter of Jupiter = $2.831629481 \times 10^{-7}$ AU^3/d^2
μ_{Sa}	Gravitational parameter of Saturn = $8.472592149 \times 10^{-8}$ AU^3/d^2
μ_U	Gravitational parameter of Uranus = $1.290955488 \times 10^{-8}$ AU^3/d^2

μ_N	Gravitational parameter of Neptune = $1.531755475 \times 10^{-8}$ AU^3/d^2
$r_{SOI,Me}$	Sphere of influence of Mercury = 7.553476×10^{-4} AU
$r_{SOI,V}$	Sphere of influence of Venus = 4.200136×10^{-3} AU
$r_{SOI,E}$	Sphere of influence of Earth = 6.183155×10^{-3} AU
$r_{SOI,M}$	Sphere of influence of Mars = 3.836898×10^{-3} AU
$r_{SOI,J}$	Sphere of influence of Jupiter = 3.228610×10^{-1} AU
$r_{SOI,Sa}$	Sphere of influence of Saturn = 3.649733×10^{-1} AU
$r_{SOI,U}$	Sphere of influence of Uranus = 3.462567×10^{-1} AU
$r_{SOI,N}$	Sphere of influence of Neptune = 5.795455×10^{-1} AU

1 Introduction

The exploration of the solar system is becoming more and more ambitious and demanding: One of the more recent missions has been *New Horizons*, the first spacecraft to reach the dwarf planet Pluto in mid-2015, after a ten-year journey through the solar system. Likewise ambitious have been the comet mission *Rosetta,* equipped with chemical propulsion, and the low-thrust mission *Dawn*, visiting the dwarf planet Ceres and the asteroid Vesta respectively.

All three spacecraft utilized gravity assists during their missions and *Rosetta* could not have accomplished its mission without this maneuver (Accomazzo et al., 2012). The asteroid sample mission *Hayabusa* and its successor *Hayabusa 2* are further examples for low-thrust missions utilizing gravity-assist maneuvers. *Hayabusa* was targeted at the near-Earth asteroid Itokawa, its successor is targeted at the near-Earth asteroid Ryugu.

Utilization of gravity-assist maneuvers increases the Δv a spacecraft can potentially achieve with a specified propellant mass by exchanging energy with a planetary body (Prussing and Conway, 1993) and has been used already for the famous *Pioneer 10* and *11* as well as both *Voyager* missions. *Voyager 2* conducted four gravity assists, one at each giant planet in the solar system, completing the so-called 'grand tour'. The flight paths of these missions are depicted in Figure 1-1 with gravity assists visible as abrupt course changes at the point of planetary encounter, e.g. for *Pioneer 11* during flyby at Saturn (note that this is a planar representation, omitting inclination changes due to gravity assists).

Low-thrust propulsion is more efficient in terms of mission performance as it requires less propellant to reach a specific mission target than chemical, i.e. impulsive, propulsion. This is owed to the larger (about factor eight to ten) specific impulse associated with low-thrust propulsion, typically 3000 s or more (Sutton and Bilbarz, 2001), enabling missions with large Δv requirements otherwise not possible or only with a more limited payload mass.

To gain advantage of both these effects it is desirable to combine low-thrust propulsion with gravity-assist maneuvers, as it was done for *Dawn*. Equipped with three electrical thrusters (operating one at a time), *Dawn* required a total of about 2000 days of thruster operation with a thrust of up to 91 mN. About one and a half years after launch, the probe reached the planet Mars, conducted a gravity-assist maneuver and continued its way to the asteroid main belt for its encounter with *Vesta*. (NASA, 2007)

Dawn is one of the few flying missions that have combined gravity assists with low-thrust propulsion; further examples are *Hayabusa* and *Hayabusa 2* (Tsuda et al, 2012). There are also examples for not (yet) flown mission designs utilizing both, e.g. a Trojan mission applying a gravity-assist maneuver at Jupiter (Diniega et al., 2013), ESA's *BepiColombo* targeted at Mercury (Jehn et al., 2008) or the Global Trajectory Optimization Competition (GTOC) for finding gravity-assist maneuver trajectories for asteroid deflection missions (Petropoulos et al., 2007).

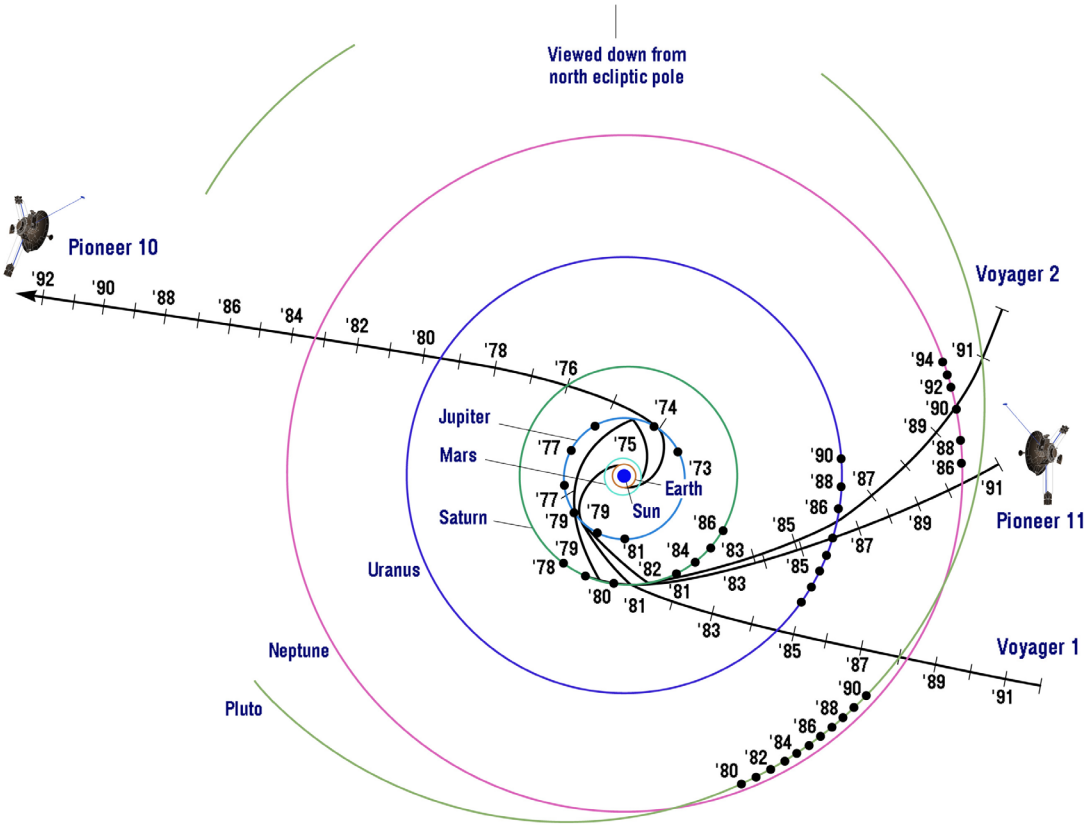

Figure 1-1: The flight paths of *Pioneer 10* and *11* as well as *Voyager 1* and *2* in the solar system. Gravity assists occur at Jupiter (*Pioneer 10* and *11*, *Voyager 1* and *2*), Saturn (*Pioneer 11*, *Voyager 1* and *2*), Uranus (*Voyager 2*) and Neptune (*Voyager 2*). They are visible as changes (note: inclination changes are not visible in this two dimensional representation). Image: NASA

Dawn has been designed to be feasible even without the gravity assist (and has been carrying excessive Xenon fuel) but used the Δv change of the gravity assist for extra margin (Rayman et al., 2006). Its original velocity vector has been turned by 78° due to the Mars gravity assist and the spacecraft trajectory's inclination was changed by 4.4°. In total, the maneuver provided a Δv of 2.6 km/s, i.e. about 20% of the total mission Δv. (Rayman and Mase, 2010)

Considering a growing importance of low-thrust propulsion and its potential as mission enabler, it is therefore beneficial to have access to a tool that allows gravity-assist optimization of low-thrust trajectories. Establishing a method and tool for the purpose of preliminary mission analysis and trajectory optimization is the goal of this thesis. First results of the presented work have been published in (Maiwald, 2015, Maiwald, 2016, Maiwald, 2017a and Maiwald, 2017b).

1.1 Low-Thrust and Gravity-Assist Optimization

Low-thrust missions require extensive optimization to achieve the largest effect because of the theoretically infinite solution space of the trajectory's control variables history (Kim, 2005), whereas impulsive maneuvers usually have a solution space limited to three dimensions (e.g. two thrust angles (Dachwald, 2003) and the thrust impulse's magnitude).

Due to this nature of the problem, simplifications have to be made and the fact that the function behind the optimization problem is unknown calls for heuristic solution approaches, e.g. evolutionary algorithms (Dachwald, 2003).

For optimization of gravity-assist trajectories the sequence is usually determined by a mission analyst and not part of the actual optimization. Various approximation methods are used for the actual trajectory calculations, whereas the sequence of the gravity-assist partners is simply fed into the respective optimization tool or method (e.g. Crain et al., 2000, Debban et al., 2002, Jehn and Langevin, 2016). For sequences not associated with low-thrust propulsion, but impulsive, Strange and Longuski (Strange and Longuski, 2002) and Labunsky et al. (Labunsky et al., 1998) have used a method based on Tisserand's Criterion (Miller and Weeks, 2002). They translated this energy-based function of orbit parameters, originally used to derive gravity-assist effects on comets passing by Jupiter, into graphs that enable sequencing of gravity-assist partners. However, the conditions for usage of Tisserand's Criterion, prescribe that gravitation is the sole force acting on the small partner of a gravity assist, which would violate the idea of using low-thrust propulsion in combination.

1.2 Thesis Topic

The goal of this thesis is to establish a method for optimizing low-thrust missions utilizing gravity-assist maneuvers to improve the overall mission performance. The frame are low fidelity trajectories, allowing quick assessment of mission feasibility. In difference to publically available methods of optimization, the sequence of gravity-assist partners is intended to be part of the optimization process, allowing to find new useful sequences without relying on previous experience and in general provide a more thorough and extensive analysis of the search space. To accomplish this, it is first investigated how gravity assists are optimized for impulsive missions and if and how the same method can be applied for low-thrust. Following this evaluation, a new method is derived and is evaluated regarding its suitability for the purpose of finding gravity assists and regarding its performance.

1.3 Content Overview

This first chapter serves as an introduction to the topic and gives an overview over the idea of combining low-thrust trajectory optimization with gravity-assist sequence optimization.

Chapter 2 focuses on the mathematical basics and the state of the art in both disciplines. It explains the astrodynamical basis for an energy exchange by gravity assists and how such maneuvers are usually modeled for calculation. It introduces the parameters of hyperbolic excess velocity and deflection, which define the gravity assist. In the next subchapters, the method of Tisserand graphs is introduced as is their usage. Afterwards, low-thrust optimization and its special challenges are presented. These challenges are caused by the fact that the search space for such an optimization task is infinite, due to the required continuous thrust. Objective functions and constraints are explained and an overview over the methods used to solve such optimization problems is provided. At the end of the chapter a review is given of previous attempts to combine low-thrust and gravity-assist optimization and how that is usually conducted with a preset sequence of gravity-assist partners as opposed to making the sequence part of the optimization. Furthermore, the challenges caused by using Tisserand's Criterion for low-thrust application are introduced.

Chapter 3 describes the scientific goals of this work and explains the path by which they are sought to be reached.

As a first step on that path, Chapter 4 investigates the applicability of Tisserand's Criterion, and in extension Tisserand graphs for missions involving low-thrust propulsion, by first explaining its derivation via the Jacobi-Integral. With the purpose of relating inaccuracies caused by the introduction of thrust into the problem – as opposed to a gravitation-only situation, which is the basis for Tisserand's Criterion – inaccuracies caused by diverting from the premise of a circular restricted three-body system are assessed in their magnitude. Violated assumptions are e.g. a non-constant mass of the spacecraft and non-circular orbits. As the associated deviation in energy caused by the continuous thrust is found to be not negligible, a modified Tisserand's Criterion is derived, leading to a correction term to be used when applying it on low-thrust trajectories. This is discussed in relation to the previously mentioned violated assumptions and a recommendation on the usage of Tisserand's Criterion is given.

Chapter 5 contains various aspects concerning the optimization of the desired low-thrust gravity-assist trajectories. First, possible optimization algorithms are described and algorithms for usage are selected. Furthermore, it is explained which optimization parameters need to be used for the different parts of the optimization – namely the gravity-assist sequence and the low-thrust trajectory. Additionally, using an investigation of possible areas of maximum Δv change, constraints on the solution space are presented that have the potential to ease the optimization process. At the same time, constraining the solution space based on likely gravity-assist partners (e.g. excluding Venus as gravity-assist partner once the spacecraft has already reached Jupiter) is explained. In addition the chapter presents the objective function to be used, based on Δv change, and the trajectory model in the form of a shape-based method.

In Chapter 6, finally all steps of the optimization method are summarized, including the benchmarking of the resulting sequence with non-gravity-assist trajectories.

In Chapter 7, the numerical tool for experimentally evaluating the method is presented along with the tests for evaluation. It is shown how the trajectory design is undertaken with gravity-assist sequencing as part of the optimization. It is further investigated which of the previously selected algorithms performs best and a case study in the form of a transfer from Earth to Jupiter is used to proof the concept of the method. Comparison calculations of existing trajectories are also included. The obtained results are discussed by reviewing the success rate of the optimization and the effect of constraining the solution space by the maximum Δv change and of the algorithm selection. Finally, the usefulness of the optimization method derived in this work is evaluated and open issues are presented.

The final chapter, Chapter 8, contains the summary and conclusion of the thesis.

2 State of the Art for Gravity-Assist and Low-Thrust Mission Design

This chapter presents the mathematical description of gravity assists with the formulas relevant for the remaining thesis. A general overview over application and history of gravity-assist maneuvers can be found in (Prussing and Conway, 1993) and (Messerschmid and Fasoulas, 2005). It is further reviewed what the state of the art of low-thrust trajectory optimization and gravity-assist sequencing is and explained that up to now no method has been published that combines both in the sense that the gravity-assist sequence is also part of the optimization effort and not input for the optimization.

As a basis for further considerations, the concept of the energy relation called Tisserand's Criterion is explained. This parameter is conserved during a gravity assist assuming that only gravitation is acting on the spacecraft (or any other body, e.g. a comet) during the flyby. It is therefore used for planning gravity-assist maneuvers for impulsive missions. Finally, it is discussed why its application on low-thrust missions is not as straightforward as for impulsive missions.

2.1 Δv Change via Gravity Assist

The possibility to transfer energy from one body to another body in the solar system – usually a comet, passing asteroid or spacecraft – is a consequence of the multi-body system. The simple two-body view does not allow exchange of energy between the system's bodies. Nonetheless, a common simplification of gravity-assist maneuvers is to assume a two-body system for the spacecraft's actual flyby of planetary body. Further, usual assumptions are (Prussing and Conway, 1993, p. 130ff):

- The heliocentric, planetary velocity remains constant for the duration of the transfer (due to a short fly-by time), and thus
- the spacecraft velocities inbound to the planet and outbound are equal in magnitude (i.e. relative to the planet, energy is conserved).

Figure 2-1 shows a schematic view of a gravity-assist situation with the involved velocities. The velocity vector $\vec{v}_\infty$ of a spacecraft or other body is turned during the close encounter with the planet as it is subject to the planet's gravitation acceleration. As the magnitude of the entry- and exit velocities, v_∞, called hyperbolic excess velocity, in the planetcentric view remains constant, the gravity-assist effect can be described as a rotation of the entry-velocity vector by an angle δ (see Figure 2-1), called deflection. It is depending on the Δv change resp. the geometry of the flyby (Prussing and Conway, 1993, p. 130ff):

$$\Delta v_{GA} = \frac{2\, v_\infty}{1 + \frac{{v_\infty}^2 \cdot r_{per}}{\mu_{pl}}} \tag{2-1}$$

$$\Delta v_{GA} = 2\, v_\infty \, \sin\frac{\delta}{2} \tag{2-2}$$

$$\Rightarrow \delta = 2 \cdot \arcsin\left(\frac{1}{1+\frac{{v_\infty}^2 \cdot r_{per}}{\mu_{pl}}}\right), \tag{2-3}$$

where r_{per} is the spacecraft radius at the hyperbola's pericentre and μ_{pl} the planet's standard gravitational parameter. A change in velocity is directly linked to a change in energy, via the kinetic energy (Prussing and Conway, 1993):

$$\varepsilon = \frac{v^2}{2} - \frac{\mu}{r} = -\frac{\mu}{2\,a}\,, \tag{2-4}$$

where ε is the specific orbital energy, v the velocity, r the position of the spacecraft on the trajectory, a the semi-major axis and μ the central body's gravitational parameter. Equation (2-4) describes the orbital energy of an elliptical orbit and links the change in energy directly with a change in the orbit's geometry, due to its influence on the semi-major axis. A hyperbolic trajectory has a positive value $\frac{\mu}{2\,a}$ as orbital energy; a parabolic trajectory has an orbital energy of 0.

In Figure 2-1 it is also apparent that the direction of the vector rotation depends on where the spacecraft passes the planetary body, in this example whether it passes in front or behind the planet (actual flybys are less clearly distinguished). The first of which results in a reduction of heliocentric velocity, the latter in an increase. In the above example, this can be estimated by the rotation direction of angle δ. Rotation in a positive sense (left) results in a positive Δ*v* change in Eq. (2-2). A negative rotation (right) of the deflection δ means also a negative Δ*v*. This can be more clearly formulated by vector addition; an example is given in Figure 2-4 and is explained in the coming sections. For a detailed derivation the reader is pointed towards (Prussing and Conway, 1993) or (Messerschmid and Fasoulas, 2005).

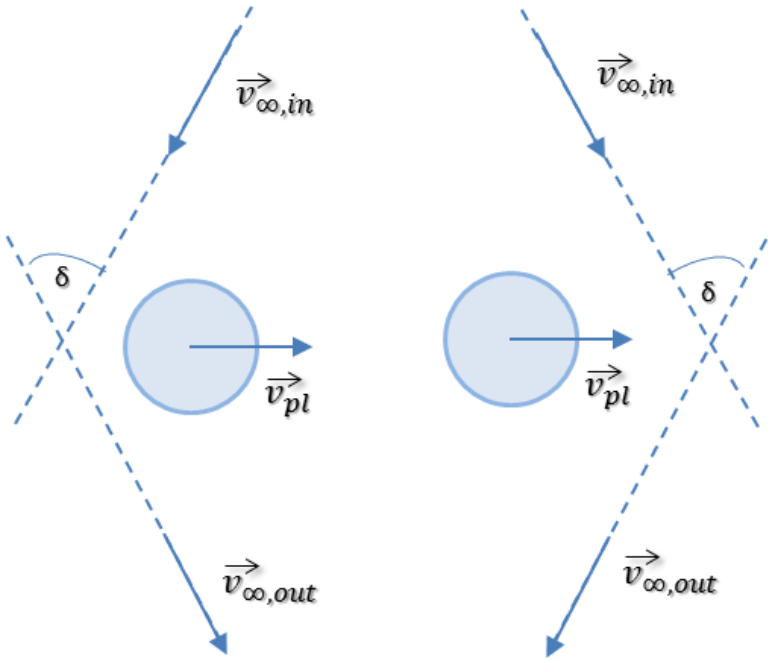

Figure 2-1: Velocity vector relations in a gravity assist example passing behind the planet (left) and in front (right).

2.2 Powered Gravity-Assist Maneuvers

The effect of a gravity-assist maneuver can be increased by support from an onboard propulsion system – either by breaking or accelerating the spacecraft. The operation of the propulsion system subjects the spacecraft to an additional Δv in addition to the non-powered gravity assist. Generally its kinetic energy is changed (expressed as ΔE), proportional to its initial velocity and the Δv ($\Delta E = v_1 \cdot \Delta v + \frac{\Delta v^2}{2}$ for tangential thrust) (Yam et al. 2004), i.e. the larger the velocity at the beginning of the maneuver (v_1), the larger the change by this maneuver. The rationale behind this can be derived in the following way, based on Figure 2-2 ($\vec{v}_1$ as velocity before the gravity assist, $\vec{v}_2$ after the gravity assist and β as angle between $\vec{v}_1$ and the gravity assist's $\Delta\vec{v}$):

$$v_2{}^2 = v_1{}^2 + \Delta v^2 - 2v_1\Delta v \cdot \cos(\pi - \beta) = v_1{}^2 + \Delta v^2 + 2v_1\Delta v \cdot \cos(\beta) \tag{2-5}$$

Be ΔE the difference in mass specific kinetic energy between the conditions 1 and 2, it can be written as:

$$\Delta E = E_2 - E_1 = \frac{v_2{}^2}{2} - \frac{v_1{}^2}{2} = \frac{1}{2}(v_2{}^2 - v_1{}^2) \tag{2-6}$$

Inserting Eq. (2-5) and with β = 0 for tangential thrust in the apside, this becomes:

$$\Delta E = v_1 \cdot \Delta v + \frac{\Delta v^2}{2} \tag{2-7}$$

It can be seen that the kinetic energy change caused by a maneuver is directly depending on the initial velocity v_1 besides the Δv. This means it is best to time the maneuver in a way that it coincides with the largest velocity, which is the point of periapsis passage for a gravity-assist maneuver. This fact is also called Oberth-Effect as Oberth (1970) was the first to describe it. To allow orbit determination during flybys of low-thrust spacecraft, this currently cannot be exploited as coast arcs need to be planned to begin several weeks before such an encounter for navigation purposes (García Yarnoz et al., 2006).

2.3 Sequence Finding

The energy transfer between spacecraft and a gravity-assist partner has been described in the previous sections. This energy transfer is a consequence of the multi-body system. However, the actual implementation of gravity assists and finding the corresponding sequence of gravity-assist partners is challenging. So called Tisserand graphs are a graphical method to determine possible sequences for impulsive missions. They are based on the energy relation called Tisserand's Criterion (introduced in the next subsection) first published in *Traité de Mécanique Céleste* in the late 19th century (Kloster et al., 2011).

Tisserand graphs are graphical representations of orbit parameters for a given planet (or other

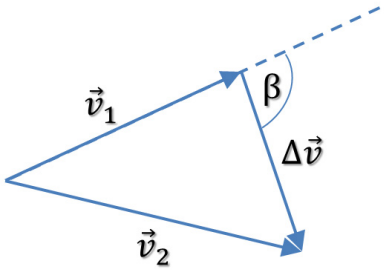

Figure 2-2: Velocity vector relations for any maneuver causing a $\Delta\vec{v}$, with the angle β describing the angle between the initial velocity and the $\Delta\vec{v}$.

flyby partner) and an affected spacecraft and have been developed independently by two research groups headed by Labunsky (Labunsky et al., 1998, pp. 101 - 197) and Strange and Longuski (Strange and Longuski, 2002), resp. Heaton, Strange and Longuski (Heaton et al., 2002). They link a spacecraft's orbital period and pericentre distance in a primary body coordinate system with its energy relative to the secondary body.

Heaton et al. describe how the trial and error design of a Europa Orbiter tour through the Jupiter system eventually led to a graphical approach for planning after they began identifying valuable tours by analyzing the spacecraft's orbital period and periapsis around the primary body, in this case Jupiter (Heaton et al., 2002). This finally led to the application of graphs, which are by now referred to as Tisserand graphs (Strange and Longuski, 2002). Generally, it has to be noted that this method assumes circular and coplanar orbits of the planetary bodies and a coplanar trajectory for the spacecraft around the primary body (Heaton et al., 2002).

An example set of graphs is shown in Figure 2-3. Each point on the graphs within this diagram, represents a Kepler ellipse with a shape set by its orbital period, which is a function of the semi-major axis, and the pericentre distance. Its angular tilt towards the planet is not specified, but the spacecraft passes by the given planet with a certain v_∞ (marked in the diagram) on this ellipse. The two extreme positions at both ends of the respective graph represent ideal Hohmann-transfer ellipses between the given pericentre distance and the respective outer radius (designated by the semi-major axis a, here implicitely stated via the orbital period ($T = f(a)$). The lower ends of the graphs represent Hohmann-transfer ellipses to inner orbits, i.e. their velocity changes are opposite to their flight direction; the upper ends represent those to outer orbits, i.e. the velocity changes are in flight direction. In both cases the Δv is applied tangentially and corresponds to the respective v_∞. The creation and usage of such graphs are described in the following subsections.

2.3.1 Tisserand's Criterion

This subsection serves only as introduction and a more thorough derivation and analysis is presented in Chapter 4. Tisserand's Criterion is a utilization of the Jacobi integral derived by Francois Felix Tisserand and originally applied to orbit perturbations of comets caused by Jupiter. Its application is now more general and not restricted to Jupiter. (Miller and Weeks, 2002)

Equation (2-8) describes the Jacobi-Integral (C_J is the integral's constant, referred to as Jacobi constant), an energy function where conservation applies, in inertial coordinates. It is based on the circular restricted three-body system (Miller and Weeks, 2002):

$$\dot{x}^2 + \dot{y}^2 + \dot{z}^2 = 2\omega\,(x\dot{y} - y\dot{x}) + 2\frac{\Upsilon M_a}{R_a} + 2\frac{\Upsilon M_b}{R_b} - C_J \tag{2-8}$$

Here x, y, and z are the coordinates of the point mass, ω the angular velocity of the primary and secondary bodies around their common barycenter, Υ the universal gravitational constant, M the respective mass of those two bodies (a and b), and R their respective distances to the point mass (Miller and Weeks, 2002).

With the help of some simplifications and assumptions (see Section 4.2.4), Eq. (2-8) can be expressed as (Miller and Weeks, 2002):

$$\frac{1}{a_1} + 2\sqrt{\frac{a_1(1-{e_1}^2)}{{R_{pl}}^3}}\cos i_1 \approx \frac{1}{a_2} + 2\sqrt{\frac{a_2(1-{e_2}^2)}{{R_{pl}}^3}}\cos i_2 \tag{2-9}$$

with the parameters denoted as before and R_{pl} describing the distance of the secondary body around the primary body, e.g. a planet around the sun or a moon around a planet, and the indeces 1 resp. 2 denoting the conditions pre and post encounter.

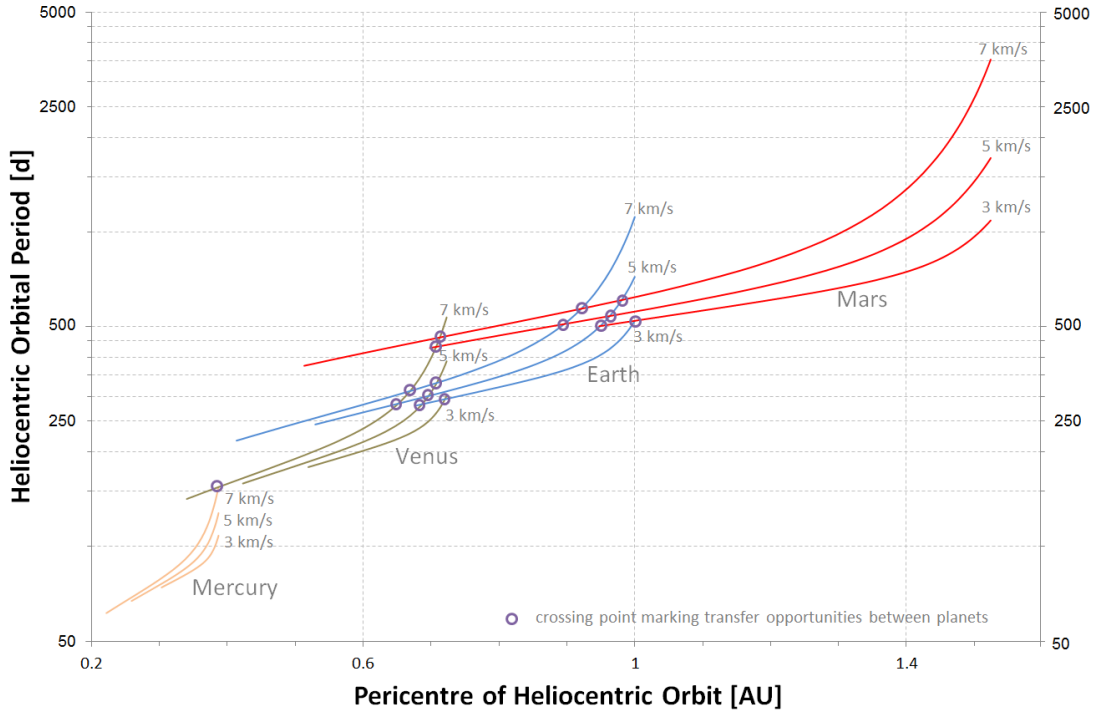

Figure 2-3: Example Tisserand graphs for hyperbolic excess velocities (v_∞) of 3 to 7 km/s and inner solar system planets, Mercury (orange), Venus (brown), Earth (blue) and Mars (red). The heliocentric orbital period (in days) is depicted as a function of a heliocentric pericenter (in AU).

For Tisserand himself, the criterion's application was to measure the orbital parameters of two comets observed at different points in time and with the help of this criterion determine whether it was in fact the same object. If the function's value, also called Tisserand Parameter, is approximately the same for both comets, they are likely the same object and the change in orbital parameters has just been caused by perturbations due to a close fly-by at Jupiter (or in theory any other body larger than the comet). Confirmation can then be gained by propagating the orbits back (for the second observation) or forward (for the first observation) in time to identify a close approach to such a body. (Miller and Weeks, 2002)

For designing gravity-assist maneuvers, Tisserand's Criterion applies as well, i.e. a spacecraft's orbit before and after the encounter with its gravity-assist partner has to fulfill this rule. It sets the limits in which the orbital parameters can be changed with a given body. (Miller and Weeks, 2002)

Equation (2-9) can be modified by multiplying both sides of the equation with R_{pl} obtaining the formulation, which is used by Strange and Longuski (2002):

$$\frac{R_{pl}}{a_1} + 2\sqrt{\frac{a_1(1-{e_1}^2)}{R_{pl}}}\cos i_1 \approx \frac{R_{pl}}{a_2} + 2\sqrt{\frac{a_2(1-{e_2}^2)}{R_{pl}}}\cos i_2 \tag{2-10}$$

Both formulations are used in the literature.

2.3.2 Deriving Tisserand Graphs from Orbital Parameters

As depicted in Figure 2-4 the maximum heliocentric velocity change obtainable through a gravity-assist maneuver depends on the heliocentric, planetary (or any other satellite's) velocity, the planetcentric, hyperbolic excess velocity of the spacecraft and the angle α between these two (Strange and Longuski, 2002).

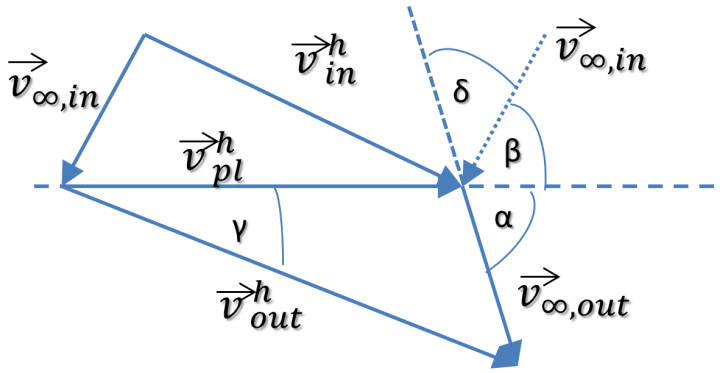

Figure 2-4: Example situation of velocity vectors as applicable during a gravity assist. The index *h* denotes heliocentric velocities; non-indexed velocities are planetcentric.

The maximum velocity can be derived with the help of the diagram in Figure 2-4. It occurs for $\alpha = 0$ and equals $v^h_{pl} + v_{\infty,out}$. In case of $\alpha = 180°$, the heliocentric velocity reaches its minimum with $v^h_{pl} - v_{\infty,out}$. These two extreme values are the starting resp. end points of the graphs as shown in Figure 2-3 (Strange and Longuski, 2002). The remaining values can be obtained by varying the angle α, which then allows creating graphs like shown in Figure 2-3 for constant values of v_∞.

To create the desired graphs, it is mandatory to first derive the heliocentric velocity of the spacecraft after the encounter and from that the orbital parameters of the spacecraft. These are the semi-major axis a (and thus implicitly the orbital period) and the heliocentric pericentre distance r_p (which in combination with a provides the shape and is connected with it via the eccentricity e, as $r_p = a\ (1\text{-}e)$) (Strange, 2012). As mentioned before and shown later, the orbital period T is a function of the semi-major axis (also known as the third Keplerian law).

To determine an orbit six independent values are needed, which can be provided by the three-dimensional position and velocity vectors at any given time (this is reduced for the planar problem). For further considerations the heliocentric position is assumed to be identical for spacecraft and planet and the planet is assumed to be on a circular orbit. The knowledge of this position already supplies three values (resp. two for a planar problem).

From Figure 2-4 mainly two formulations can be derived. First, it allows finding a formulation for the magnitude of the heliocentric, outward velocity of the spacecraft (i.e. after the flyby, note the magnitude of $v_{\infty,out} = v_{\infty,in} = v_\infty$) with the help of the cosine law:

$$\begin{aligned} {v^h_{out}}^2 &= {v_\infty}^2 + {v^h_{pl}}^2 - 2\, v_\infty v^h_{pl} \cos(\pi - \alpha) \\ &= {v_\infty}^2 + {v^h_{pl}}^2 + 2\, v_\infty v^h_{pl} \cos(\alpha). \end{aligned} \tag{2-11}$$

Second, the angle γ between the planet's velocity and the heliocentric, outward velocity of the spacecraft can be formulated as (again via the cosine law):

$$\cos\gamma = \frac{{v^h_{out}}^2 + {v^h_{pl}}^2 - {v_\infty}^2}{2\, v^h_{out} v^h_{pl}} \tag{2-12}$$

Combining Eq. (2-11) and (2-12) leads to:

$$\cos\gamma = \frac{v_{pl}^{h} + v_{\infty}\cos(\alpha)}{\sqrt{{v_{\infty}}^{2} + {v_{pl}^{h}}^{2} + 2\, v_{\infty} v_{pl}^{h}\cos(\alpha)}}. \tag{2-13}$$

In Eq. (2-13) the angle γ only depends on the angle α (referred to as the "pump angle" by Kloster et al. (Kloster et al., 2011)) and the velocities of the spacecraft and planet. The latter is given by the selected planetary body, the first of which is variable (see Figure 2-3) as is the angle α. For each gravity assist the pump angle is changed via the rotation of the velocity vectors by the angle δ, see Figure 2-1.

Both these angles are connected via the relative position of the velocity vectors in this example named β, assuming that the planet's velocity coincides with the reference plane (if this is not the case, the planet's velocity vector angle towards the reference plane is known as well, still allowing to relate the vectors to each other). Varying the pump angle α from 0 to 180° allows the creation of the graphs.

However, at this time only the magnitude and direction (γ) of the spacecraft's heliocentric, outward velocity is known. Via the angle γ the actual vector can be determined (assuming a planar orbit as mentioned before, as Tisserand graphs only consider a planar view). Figure 2-5 shows an arbitrary positioning of the heliocentric spacecraft and planetary velocities and the angles describing the relative orientation of the two vectors (the angle γ is known at this point) and their absolute orientation in the coordinate frame. The angle ζ is practically known via the planetary velocity vector, as it can be derived that:

$$\cos\zeta = \frac{v_{pl,x}^{h}}{v_{pl}^{h}} \tag{2-14}$$

Furthermore, for the angle ξ the following relation can be found:

$$\xi = \zeta - \gamma \tag{2-15}$$

With both angles known, this leads to the vector description of the heliocentric outward spacecraft velocity (it has to be noted that Tisserand graphs are planar representation, i.e. the z-component = 0):

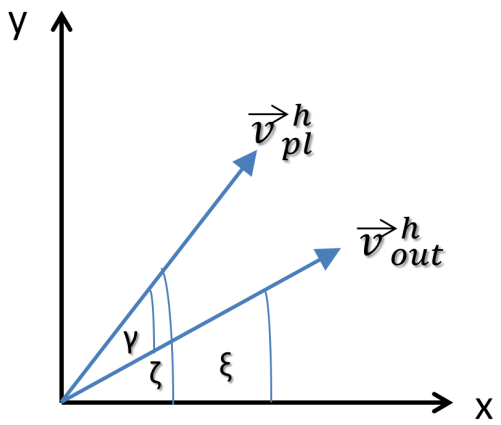

Figure 2-5: Heliocentric planetary and spacecraft velocities in an arbitrary coordinate frame.

$$\vec{v}_{out}^{\,h} = \begin{pmatrix} v_{out,x}^h \\ v_{out,y}^h \\ v_{out,z}^h \end{pmatrix} = \begin{pmatrix} \cos(\xi) \cdot v_{out}^h \\ \sin(\xi) \cdot v_{out}^h \\ 0 \end{pmatrix} = \begin{pmatrix} \cos(\xi) \cdot \sqrt{{v_\infty}^2 + {v_{pl}}^2 + 2\, v_\infty v_{pl} \cos(\alpha)} \\ \sin(\xi) \cdot \sqrt{{v_\infty}^2 + {v_{pl}}^2 + 2\, v_\infty v_{pl} \cos(\alpha)} \\ 0 \end{pmatrix} \tag{2-16}$$

As mentioned before, the position vector of the spacecraft matching the point of time of the heliocentric outward velocity is identical to the planetary position (in a first approximation), therefore the position and velocity vectors are known and can be used to determine the orbit of the spacecraft.

The mass specific angular momentum is (Prussing and Conway, 1993, p. 13):

$$\vec{h} = \vec{r} \times \vec{v} \tag{2-17}$$

Applied to the problem at hand, Eq. (2-17) yields:

$$\vec{h} = \vec{r}_{pl} \times \vec{v}_{out}^{\,h} \tag{2-18}$$

$$\Rightarrow \vec{h} = \begin{pmatrix} r_{pl,x} \\ r_{pl,y} \\ 0 \end{pmatrix} \times \begin{pmatrix} v_{out,x}^h \\ v_{out,y}^h \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ r_{pl,x} \cdot v_{out,y}^h - r_{pl,y} \cdot v_{out,x}^h \end{pmatrix}$$

$$\Rightarrow h = r_{pl,x} \cdot v_{out,y}^h - r_{pl,y} \cdot v_{out,x}^h$$

The orbit's shape, i.e. its eccentricity, can now be derived via (Prussing and Conway, 1993, p. 52):

$$e = |\vec{e}| = \left| \frac{\vec{v}_{out}^{\,h} \times \vec{h}}{\mu} - \frac{\vec{r}_{pl}}{r_{pl}} \right|, \tag{2-19}$$

where μ is the standard gravitational parameter of the primary body. The vector product is formulated as:

$$\vec{v}_{out}^{\,h} \times \vec{h} = \begin{pmatrix} v_{out,x}^h \\ v_{out,y}^h \\ 0 \end{pmatrix} \times \begin{pmatrix} 0 \\ 0 \\ r_{pl,x} \cdot v_{out,y}^h - r_{pl,y} \cdot v_{out,x}^h \end{pmatrix} = \begin{pmatrix} h \cdot v_{out,y}^h \\ -h \cdot v_{out,x}^h \\ 0 \end{pmatrix} \tag{2-20}$$

Combined with Eq. (2-19) this becomes:

$$e = \left| \frac{\vec{v}_{out}^{\,h} \times \vec{h}}{\mu} - \frac{\vec{r}_{pl}}{r_{pl}} \right| = \left| \begin{pmatrix} \frac{h}{\mu} \cdot v_{out,y}^h \\ -\frac{h}{\mu} \cdot v_{out,x}^h \\ 0 \end{pmatrix} - \frac{1}{r_{pl}} \begin{pmatrix} r_{pl,x} \\ r_{pl,y} \\ 0 \end{pmatrix} \right| = \left| \begin{pmatrix} \frac{h}{\mu} \cdot v_{out,y}^h - \frac{r_{pl,x}}{r_{pl}} \\ -\frac{h}{\mu} \cdot v_{out,x}^h - \frac{r_{pl,y}}{r_{pl}} \\ 0 \end{pmatrix} \right| \tag{2-21}$$

$$= \sqrt{\left(\frac{h}{\mu} \cdot v_{out,y}^h - \frac{r_{pl,x}}{r_{pl}} \right)^2 + \left(-\frac{h}{\mu} \cdot v_{out,x}^h - \frac{r_{pl,y}}{r_{pl}} \right)^2}$$

With the angular momentum and eccentricity known, the orbit can be determined via the semi-major axis (resp. pericentre distance r_{per}) and the orbital period (Prussing and Conway, 1993, p. 16 and p. 18):

$$a = \frac{\frac{h^2}{\mu}}{1 - e^2} \tag{2-22}$$

$$r_{per} = a(1 - e) \tag{2-23}$$

$$T = 2\pi \cdot \sqrt{\frac{a^3}{\mu}} = 2\pi \cdot \sqrt{\frac{{r_{per}}^3}{\mu(1 - e)^3}} \tag{2-24}$$

It can be seen that the orbital period now is only a function of the eccentricity and the pericentre distance. The first of which is only depending on the position of the spacecraft, the planetary velocity and the hyperbolic excess velocity of the spacecraft to the planetary body. Thus, the graph as shown in Figure 2-3 can be derived. As the complete term including all formulations would not add to the clarity of the solution, it is omitted here and only a simplification is shown.

The actual function values are then calculated by varying α and thus $\vec{v}_{out}^{\,h}$ after selecting v_∞ and a planetary body (v_{pl}).

2.3.3 Application on Gravity-Assist Sequencing

Figure 2-6 shows an excerpt of Figure 2-3 focused on Tisserand graphs for Venus and Earth, each for hyperbolic excess velocities of 3, 5 and 7 km/s. It can be seen that the perihelion distance of the respective orbit options cannot exceed the orbital distance of the respective planet (Kemble, 2006). The maximum orbital period, depending on the semi-major axis which at the same time is an indicator for the specific orbit energy ($|\varepsilon| = \mu/(2 \cdot a)$), is achieved by the maximum heliocentric velocity, i.e. for α = 0. Vice versa the minimum orbital period concurs with the minimum heliocentric orbital velocity, i.e. α = 180°. (Strange and Longuski, 2002)

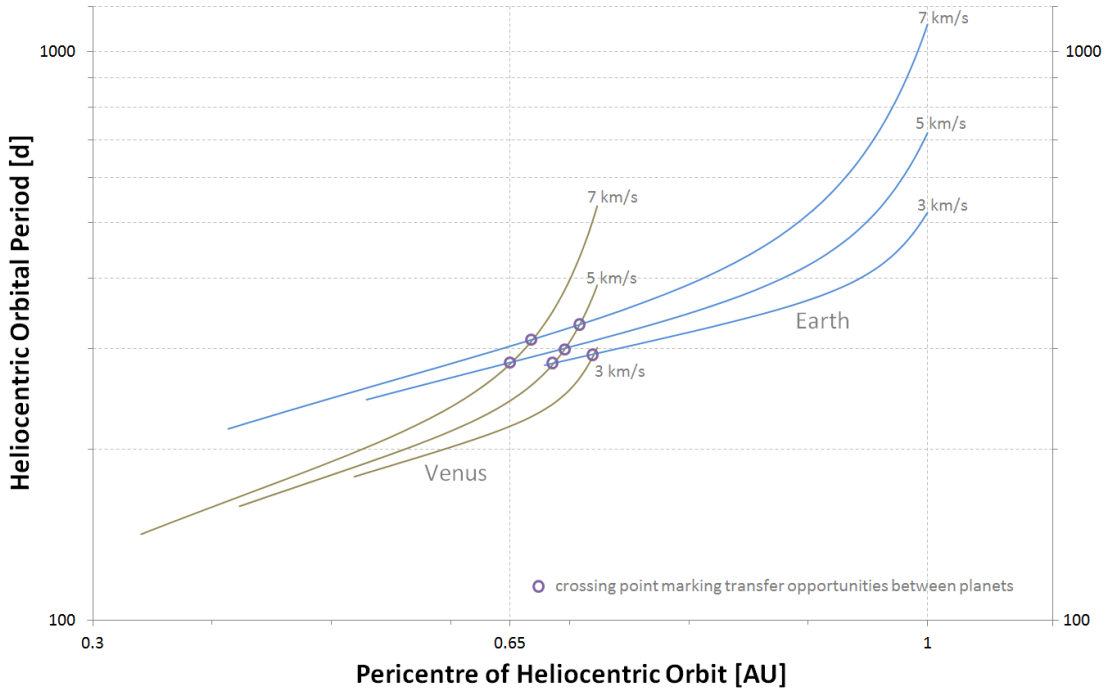

Figure 2-6: Example Tisserand graphs for hyperbolic excess velocities (v_∞) of 3 to 7 km/s and the planets Venus (brown) and Earth (blue). The heliocentric orbital period (in days) is depicted as a function of a heliocentric pericenter (in AU).

For each graph the hyperbolic excess velocity, i.e. the energy relative to the flyby body, remains constant. Contrary, the energy to the central body varies (indicated by the variable orbital period). Omitting the inclination due to the planar view, the graph provides a relationship between the shape of the orbit (via relating the pericentre distance to the semi-major axis in form of the orbital period) and its size (semi-major axis via orbital period), which concurs with Tisserand's Criterion.

Transfer from one planet to the other can be achieved at crossing points of two graphs (Heaton et al., 2002) and it can be seen that this is not possible at every orbit obtainable by a given v_∞. Transfer is possible once the pericentre of the orbit drops to a region where it crosses the orbit of a given target planet. In case of Earth and Venus, flybys have to occur in a way to drop the orbit pericentre radius to at least a distance of ca. 0.73 AU so that the spacecraft orbit touches the planetary orbit.

Starting with a certain v_∞ and angle α (depending on the selected escape trajectory), a spacecraft could leave e.g. Earth on a given heliocentric trajectory with a matching condition in the orbital period (vertical axis) and pericentre (horizontal axis). This condition can be changed with the help of a gravity assist, which is expressed by an instantaneous change in condition from one point on the graph to another. If timed correctly, i.e. with a certain resonance in orbit periods of Earth and the spacecraft, the flyby can re-occur at a later point in time, causing another change in the heliocentric velocity vector (Kemble, 2006). With this technique, the orbit can be adapted to e.g. reach Venus. With the presented hyperbolic excess velocities, it can however be seen in Figure 2-3 that reaching Mercury is not possible with flybys at Earth alone. Mars however is reachable from Venus and Earth alike.

The change of the pump angle is determined by the deflection angle δ, which in turn depends on the pericentre passage velocity of the spacecraft, i.e. indirectly on its altitude (see Eq. (2-3)). Some representations of Tisserand graphs mark certain orbit altitudes by so called "tick marks", noting positions of e.g. 100 km altitude during the gravity assist and what kind of change is achievable by a gravity assist passage in this altitude (Kloster et al., 2011)

It has to be noted that the application of these graphs is merely an energy consideration and does not include any regards to phasing (Strange and Longuski, 2002). Miller and Weeks (Miller and Weeks, 2002) describe in detail how a trajectory can be built using gravity-assist maneuvers with a method of patched conics, also applying Tisserand's Criterion, based on the example of the *Cassini*-probe transfer from Earth to Saturn via Jupiter.

They follow a scheme (for impulsive missions) that can be summarized as follows (Miller and Weeks, 2002):

1. Determine encounter times at Jupiter and Saturn using an initial guess based on a Hohmann transfer: Via Lambert's theorem and under assumption of no mass for the planets a conic section with known flight time is determined for two selected position vectors (launch and target bodies).
2. Determine in- and outcoming velocity vectors at the gravity-assist partner (in this case Jupiter), in case the Jacobi constants for both segments do not match, the following method is applied:
    i. The encounter time of the gravity-assist partner is kept fixed, the times at the launch and target body are varied
    ii. For the pairs of encounter times regarding the launching to gravity-assist body and gravity-assist to target body a Lambert solution is obtained, which provides the orbit elements for determination of the Jacobi constant (e.g. using the Tisserand criterion)

iii. Once a match is found, the corresponding hyperbolic conic with regard to Jupiter is calculated (disregarding trajectories that intersect Jupiter or one of its satellites)
iv. Review of the launch-body and target-body encounters, regarding the energy conditions

3. If the encounter times produce usable results, a trajectory can be obtained using patched conics: Computing the velocities at the match points between the conics, which should be near the spheres of influence.
4. The trajectory is actually integrated using the times obtained from the patched trajectory.

2.4 Discussion on Low-Thrust Application

The previously described method is usually only applied to impulsive-thrust missions. Tisserand's Criterion itself is used for changes regarding an orbit based on a circular restricted three-body system and under the assumption that only gravitation acts on the spacecraft or small body. The following sections describe the process of low-thrust optimization and relate it to gravity-assist optimization.

2.4.1 Low-Thrust Trajectory Optimization

Trajectory optimization is the task of finding the best solution regarding a certain objective as part of a mission design process. The solution contains a steering strategy that is best suited to achieve the mission under consideration of the given objective, which is usually evaluated once the trajectory is calculated (Kemble, 2006). Depending on the methods used and the actual problem to be solved, it can be reduced to finding any solution at all for a given mission layout (Dachwald, 2003).

A trajectory maps a time interval $t \in [t_0, t_f] \subset \mathbb{R}$ onto a state space $X \subset \mathbb{R}^n$ (Kim, 2005), where n is usually 6 for spacecraft trajectories as the elements of X, $\{\vec{x}_{sc}\} \in X$ (sc denoting spacecraft vectors, in this case the state), need to be able to completely define an orbit, which requires six parameters. For trajectory calculations, it is common to use the orbital position $\{\vec{r}_{sc}\} \in \mathbb{R}^3$ and velocity $\{\dot{\vec{r}}_{sc}\} \in \mathbb{R}^3$ (Dachwald, 2003). Part of the trajectory calculation is also the control function U that maps from an interval, which is commonly identical to the one of the state space, onto a spacecraft control vector $\{\vec{u}\} \in \mathbb{R}^{\mathrm{m}}$. This control vector defines the thrust direction and throttle. (Kim, 2005)

The trajectory is usually calculated by integrating the equations of motion, which can be expressed by six differential equations of first order of the following form:

$$\dot{\vec{x}}_{sc}(t) = \bar{F}[\vec{x}_{sc}(t), \vec{u}(t), t], \bar{F}: \mathbb{R}^{6+\mathrm{m}} \rightarrow \mathbb{R}^6 \tag{2-25}$$

These differential equations are a dynamic constraint for the optimization problem. A control that is able to find an optimal trajectory is referred to as optimal control vector $\vec{u}^*$. (Dachwald, 2003) The task to find such an optimal control is referred to as optimal control problem (Kim, 2005).

Optimization Objectives

Each optimization is undertaken with regard to certain optimization objectives, resulting in a property that needs to be maximized (or minimized, which is exchangeable via the sign) at the end of the optimization process (Kemble, 2006).

Usually the objective function, which mathematically describes the objective(s), has the form of (Kemble, 2006):

$$J = \Phi\left(\vec{x}_{sc}(t_f), t_f\right) + \int_{t_0}^{t_f} L(\vec{x}_{sc}, \vec{u}, t)\, dt, \tag{2-26}$$

where Φ is a function of the final state and time, L is a general function of the instantaneous state, control and time, integrated over time along the trajectory.

For trajectory optimization, there are two categories of objectives (Dachwald, 2003):

1) minimization of transfer time
   i. for a given payload and propellant mass
   ii. for a given launch and propellant mass
2) minimization of propellant mass:
   i. for a given payload mass and transfer time
   ii. for a given launch mass and transfer time

It should be noted that the launch mass is not independent of the payload mass and therefore there is no full distinction between the two. On the other hand, the maximization of payload mass for a given launch mass (resp. minimization of launch mass for a given payload mass) is a typical system engineering optimization problem.

While the application of gravity-assist maneuvers does not necessarily benefit minimization of transfer time, it can reduce the required propellant mass and therefore even enable a mission in the first place.

With regard to Eq. (2-26) the objective of minimization of flight time is expressed as a function J_T (Kim, 2005):

$$J_T = \int_{t_0}^{t_f} dt = t_f - t_0 = T. \tag{2-27}$$

The minimization of propellant mass yields the following as objective function (Kim, 2005):

$$J_{m_p} = \int_{t_0}^{t_f} \dot{m}_p dt = m_p(t_f) - m_p(t_0), \tag{2-28}$$

where $\dot{m}_p$ is the propellant mass flow and m_p the propellant mass.

These objectives of optimization usually compete for trajectories – minimization of transfer time usually requires large thrust, which in turn requires large amounts of propellant. Therefore, trajectory optimization is as multi-objective optimization.

Optimization Constraints

Besides the optimization objective, the optimization is often subject to constraints that have to be fulfilled during optimization. There are various kinds of constraints and they can be distinguished in two types. One type of constraint, terminal, requires a condition to be fulfilled at a specific point in time, t_e, (e.g. proximity to a target body at the end of the trajectory) or it is evaluated during the whole trajectory. (Kemble, 2006)

The first of which can be expressed as (Kemble, 2006, p. 90):

$$C = f_1(\vec{x}_{sc}(t_e), \vec{u}, t_e) \tag{2-29}$$

where f_1 is the function that describes the constraint. The latter can be provided by (Kemble, 2006, p. 90):

$$C = f_1(\vec{x}_{sc}(t_f), t_f) + \int_{t_0}^{t_f} f_2(\vec{x}_{sc}, \vec{u}, t)\, dt, \tag{2-30}$$

where f_1 and f_2 are two functions that can be defined to describe the required constraint.

Path constraints affect the whole trajectory or a section of it (e.g. that the consumed propellant mass may not exceed the loaded propellant mass) and can be formulated as follows (Kemble, 2006):

$$C(\vec{x}_{sc}(t), \vec{u}, t) \leq f_1(\vec{x}_{sc}(t), \vec{u}, t) \tag{2-31}$$

Each type of constraint can be an equality or inequality constraint. The first of which is common for terminal constraints, the latter for path constraints. Generally, however equality and inequality are applicable for both. Equations (2-29) to (2-31) have to be adapted accordingly (Kemble, 2006).

Mission Requirements

There are several possible mission scenarios that can be subject to optimization. Their basic classifications are *orbit transfer problem*, *flyby problem* and the *rendezvous problem*. If required the scenarios can be extended to cover e.g. several flybys of different targets, etc. (Kim, 2005)

The mathematical formulation for each problem is (Dachwald, 2003):

- *Rendezvous problem:* The control function U needs to provide a control vector $\vec{u} \in \mathbb{R}^m$, which transforms the initial state vector of the space craft $\vec{x}_{sc}(t_0)$ (containing e.g. the velocity and position information) to the state vector of the target $\vec{x}_T(t)$ while subject to the constraints of the equations of motion and the terminal constraint that $\vec{x}_{sc}(t_f)$ needs to be equal to $\vec{x}_T(t_f)$.
- *Flyby problem:* The flyby problem is identical to the rendezvous problem, except that only the position components of the state vector need to be equal.
- *Orbit transfer problem:* In this case a vector $\vec{Z}$ containing the orbit elements has to be transformed to equal the elements of the target orbit at the end of the trajectory.

Solving of Low-Thrust Problems

High-thrust trajectories that use few impulsive thrust maneuvers usually have few dimensions regarding the search space of an optimization. For a single maneuver, e.g. for a flyby, the maneuver at the beginning of the mission is defined by two thrust angles and the thrust impulse's magnitude. This means the solution space has three dimensions (Dachwald, 2003). While a second maneuver might be necessary, it is already uniquely constrained by the first.

Low-thrust missions are characterized by long periods of continuous thrust. While the control variables in $\vec{u}$ remain the same, the continuity of the thrust in time makes the solution space infinite (Kim, 2005).

Only in rare cases such problems can be solved (Dachwald, 2003), therefore a usual approach is to numerically discretize the problem in order to reduce the dimensions of the solution space. Thus, the continuous time interval $t \in [t_0, t_f] \subset \mathbb{R}$ is divided into single points in time which then make up a discrete time interval $\bar{t}$ in the same limit points. Following this approach, the optimal solution

is no control vector $\vec{u}^*(t)$ but a control vector history $\vec{u}^*[\bar{t}]$, which has a finite number of dimensions, as the time interval $\bar{t}$ is also finite (Dachwald, 2003).

Optimization Methods

Optimization methods for trajectories can be divided into direct and indirect methods, which are usually local optimization methods (Dachwald, 2003).

Direct methods directly provide solutions for the control variables, whereas indirect methods solve a Two-Boundary-Problem to gain the solutions indirectly (Kim, 2005). The latter involves calculus of variations, i.e. calculation of the problem's Hamiltonian; the first of which is usually an approximation (Kemble, 2006).

Techniques for direct optimization are Non-linear Programming and Collocation, which involve the discretization of the time interval as well (Kim, 2005). No gradient calculations are used but interpolations, which only give approximate results (Kemble, 2006). Other methods are so-called shooting methods, where initial guesses are used to propagate trajectory solutions and evaluate these at the end of the propagation. In this case the initial solutions need to be guessed by the optimizer (Kemble, 2006), which makes profound knowledge of the problem and trajectory calculation in general a requirement for the user (Dachwald, 2003).

Equations of Motion and Propulsion Models

The equations of motion for the two-body-system are based on Newtonian mechanics and are expressed as (Kim, 2005):

$$\ddot{\vec{R}} + \frac{\mu}{R^2}\vec{e}_r = 0 \tag{2-32}$$

where $\vec{R}$ is the vector between the two masses of the system, $\vec{e}_r$ the unity-vector in its direction and μ is the gravitational parameter with $\mu = Y\ (m_1 + m_2)$, where Y is the universal gravitational constant and m designating the masses of the two bodies 1 and 2. In case of accurate problems, the right side of Eq. (2-32) has to be filled with terms for perturbations and thrust, see Chapter 4.

A thrust model for a solar electric engine can be expressed as follows (Dachwald, 2003):

$$\ddot{\vec{R}}_{thrust,SEP} = \frac{1}{m}\vec{F}(\kappa, R), \tag{2-33}$$

where κ as the throttle parameter and $\vec{F}$ describing the thrust (in direction and magnitude) (Dachwald, 2003). Combined, the complete equation to model a solar electric propulsion system is:

$$\ddot{\vec{R}} = \frac{1}{m}\vec{F}(\kappa, R) - \frac{\mu}{R^2}\vec{e}_r \tag{2-34}$$

Comparable equations can be derived for other thrust sources like nuclear electric propulsion or solar sails (Dachwald, 2003).

2.4.2 Gravity Assist Optimization and Low-Thrust Trajectories

Crain et al. (Crain et al., 2000) have used a combination of a genetic algorithm and a local optimizer based on recursive quadratic programming for gravity-assist maneuver optimization. The latter has been developed for optimization of gravity-assist maneuvers already, but only on a local scale. A sequential combination of both optimization schemes was applied to optimize previously *user-given* sequences for gravity-assist maneuvers (sequences: Earth-Mars-Earth and Earth-Venus-Earth).

A similar approach has been chosen by Zhao et al. (Zhao et al., 2012), who combined the differential evolution algorithm for global search with sequential quadratic programming (SQP) methods for local optimization. The gravity-assist maneuver has been modelled by a B-plane. A B-plane is a plane perpendicular to the incoming trajectory asymptote, where it "pierces" the planet's sphere of influence, i.e. where the planet's gravitational force is dominant, and containing the focal point of its hyperbolic trajectory. For Zhao et al.'s method *the sequence has not been part of the optimization* but was provided to the optimizer.

Debban et al. (Debban et al., 2002) also optimized *preselected* gravity-assist sequences yet with a different method (not using genetic algorithms, but also a broad search involving a cost function). In their case, the low-thrust maneuvers have been approximated as multiple high thrust maneuvers and furthermore the trajectories were optimized using a shape-based method for broad searches, which later on in the process are refined. The broad search involved only a two-body model with a patched-arc approach. This approach employs Keplerian conic sections for coasting and so called exponential sinusoids for thrust arcs, which can be described by the following equation (Debban et al., 2002):

$$r = k_0 e^{k_1 \sin(k_2 \cdot \theta + \Phi)}, \tag{2-35}$$

where k_0, k_1, k_2, θ and Φ are user-defined settings that affect the shape of and thus thrust on the respective trajectory leg. These arcs can be analytically approximated with Eq. (2-35). Evaluation is done via a cost function that regards propellant mass and the arrival v_∞ in approximation. Time of flight is not regarded and needs to be assessed by the user (Debban et al., 2002).

The authors (Debban et al., 2002) then further optimize the solutions obtained with the above described method with GALLOP (Gravity Assist Low-Thrust Local Optimization Program). During the optimization, the gravity-assist maneuver is modelled as instant rotation of the velocity vector and the trajectory is divided into legs, which range from one body to the next. These legs are further subdivided into equally long parts (length referring to time) and at the midpoint of the segment one Δv is applied (the sum of Δv modelling the continuous thrust). Between these positions conic arcs are used as trajectory approximation. The leg furthermore has a so-called match-point. The start of the leg is propagated forward to this point on the trajectory, the end is propagated backwards, until they match. A sequential quadratic algorithm is used for optimizing the spacecraft mass. The method proves to be an improvement regarding calculation effort when compared with other methods of finding and calculating gravity-assist maneuvers. Overall, the need for an astrodynamics specialist was reduced by Debban et al. yet not eliminated as still *a gravity-assist sequence has to be initially chosen*. (Debban et al., 2002)

The same method was successfully used by Yam et al. (Yam et al., 2004) for investigating a nuclear powered low-thrust mission to Jupiter (JIMO). McConaghy et al. (McConaghy et al., 2003) provide various examples of the application of this method, which has been verified by comparison with other methodologies, yet the actual gravity-assist sequence is not part of the optimization but is provided by the user of their tool.

Although not using genetic algorithms in their approach, Vasile and Campagnola also employed a combination of global search methods with local search to optimize gravity-assist maneuvers (Vasile and Campagnola, 2009). The global search was applied to evaluate a large number of trajectories for their quality and the best were then used as first estimate inputs for the local optimization. In their mission design, Vasile and Campagnola, optimized a transfer to Jupiter and following that a gravity-assist sequence of Jupiter's moons for a rendezvous with Europa. Different calculations were applied to obtain solutions for the first transfer and the subsequent gravity-assist sequence. The self-designed global search algorithms employed a simplified model, e.g. assuming

coplanar body orbits and modelling of the low-thrust by using a small number of impulsive maneuvers and no actual continuous thrust. The direct method for local optimization applied a q-Gauss quadrature sum scheme. Their method was successful in determining a trajectory design for evaluation of mission feasibility (Vasile and Campagnola, 2009).

Carnelli (Carnelli, 2005) and Carnelli et al. (Carnelli et al., 2009) have investigated the use of an evolutionary neurocontroller (ENC) for optimization of gravity-assist low-thrust trajectories. Carnelli assumed that artificial neural networks can exploit the possibilities of gravity-assist maneuvers for improvement of mission performance, because earlier calculations have shown that they are able to exploit so called solar-photonic-assist maneuvers in case of solar sail trajectories. These maneuvers bring a spacecraft close to the Sun to increase thrust before it moves further away (Carnelli, 2005).

Carnelli considered two different optimization schemes: First, the application of a single ENC for the whole mission or second, several ENCs, one for each trajectory leg. As the latter would have meant that the gravity-assist sequence had to be submitted by the user of the optimization tool, he chose the first option. The calculations however were not able to improve or even reach the results of existing trajectories, which was partly due to the fact that rewarding gravity-assist maneuvers via fitness did not prove sufficient to benefit the reproduction of gravity-assist trajectories. As only few trajectories are sufficiently close to the respective body's sphere of influence, their progress was not fast enough compared to trajectories not using a gravity assist. Furthermore, the effect of gravity-assist maneuvers on the trajectories is significant and very sensitive, which makes training of the neural network difficult and often even decreases the fitness of the trajectory, therefore causing such trajectories to be extinguished from the population. Additionally, in difference to the solar photonic assist, where the source of fitness improvement does not change its position, in case of gravity-assist maneuvers, the source of additional "free" Δv does change its position and therefore it is difficult for the artificial neural network to learn how to exploit it efficiently. As an alternative to using an ENC for gravity-assist optimization, Carnelli introduced a gradient based local optimizer into the process for single gravity-assist maneuvers, which proved successful due to the fact that the solution space is smooth and only has one optimum, when applying a B-plane. This gravity assist "consultant" for the network proved to operate successful in providing a suitable insertion into the body's sphere of influence (Carnelli et al., 2009 and Carnelli, 2005).

Yang et al. (Yang et al., 2013) have used Tisserand graphs for optimizing low-thrust gravity-assist maneuvers. However, they used simplified models that approximate a low-thrust maneuver as a series of impulsive maneuvers. Furthermore, they only used a two-body system and collected the complete trajectory via a patched conic approach. They also did not investigate the suitability of Tisserand's Criterion despite the violation of its preconditions (e.g. conservation of energy). The approach of using a model involving several impulsive-maneuvers as a series to obtain similar results as with a continuous thrust has already been applied by Sims and Flanagan (Sims and Flanagan, 1999). Tsuda et al. (Tsuda et al., 2012) used a comparable model of several impulsive maneuvers to simulate the effect of continuous thrust for optimizing the gravity-assist trajectory for *Hayabusa 2*. Also, in their method *the gravity-assist partner, Earth, was set and not part of the optimization*. The mission period before the gravity-assist maneuver involved only minor thrusting for course corrections (ca. 56 m/s over a period of ca. 6 months), i.e. it is more a coasting mission involving a gravity assist than an actual propelled mission (Tsuda et al., 2015).

Chen et al. (Chen et al., 2008) have used Tisserand graphs for designing – but not optimizing – low-thrust trajectories. They mention energy considerations, which prohibit the usage of Tisserand graphs as such for depiction of low-thrust transfers (but do not quantify these considerations), yet they use the graphs to visualize their low-thrust transfers and execute the gravity-assist effects. Their analysis reviews different thrust models (e.g. constant tangential thrust, constant thrust

perpendicular to the instantaneous semimajor axis) and different sequences; the sequence itself is however set by them and the evaluation of the trajectory occurs a posteriori. They use circular orbits for their calculations. Furthermore, no variation of relevant properties (e.g. the hyperbolic excess velocity) is applied, but fixed values (e.g. 3 km/s) are used.

For *BepiColombo* the mission design has taken about a decade (Garcìa Yanoz, 2006 and Jehn and Langevin, 2016). It required tools specialized for certain aspects of the optimization (Jehn and Langevin, 2016), involved also navigational issues (Garcìa Yanoz, 2006) and while not explicitly stated, apparently it has been assumed readily that a mission design has to involve an Earth and a Venus flyby (Jehn and Schoenmaekers, 2014). At the same time, a number of coasting arcs has been involved, making it not a "pure" low-thrust mission (Jehn and Schoenmaekers, 2014).

Generally, conducting a global search for sequences of gravity assists for low-thrust trajectories is still an open field. Especially the sequencing is usually not part of the optimization in an intrinsic way and usually left to the mission analyst. As the sequence can have a strong impact on the mission performance, it should still be part of the optimization.

2.4.3 Applicability of Tisserand Graphs

The previously described Tisserand's Criterion in application for gravity-assist maneuver layout assumes an exclusive effect of the gravitational force on the participating bodies, i.e. a thrust vector is not included. The Tisserand graphs have not been designed specifically for continuous thrust, which violates the preconditions of Tisserand's Criterion.

Nonetheless, if the error introduced by this deviation is small, Tisserand's graphs could help to evaluate the usability of a gravity-assist maneuver for a given mission scenario. The order of magnitude of this deviation is estimated in Chapter 4.

3 Scientific Goals and Proceeding

The described method of Tisserand graphs is a common and useful tool to design gravity-assist sequences for impulsive missions. They can be used for defining possible partner sequences beneficial for reducing a mission's required energy effort, but without considerations of phasing. Their applicability on low-thrust trajectories needs to be analyzed further. To allow this and to define a method capable of optimizing a gravity-assist sequence and low-thrust trajectory at the same time the following questions need to be evaluated and answered:

1. *Is it possible to optimize a gravity-assist sequence for low-thrust propulsion using Tisserand's Criterion in a similar way as for impulsive missions?*
2. *Which modifications of the method are required to regard energy changes due to the continuous thrusting of the low-thrust engine?*
3. *What would be the control variables of the optimization?*
4. *Which algorithms are best suited for the optimization and why?*
5. *What is a useful formulation of an objective function that allows the usage and determination of a suitable gravity-assist sequence?*
6. *How can gravity-assist maneuvers be "properly rewarded" without losing the perspective concerning non-gravity-assist trajectories (i.e. no non-gravity-assist trajectory has a better result in the optimization objective)?*

The final goal of such an undertaking is the creation of a tool that finds a globally (near) optimal trajectory under the application of gravity-assist maneuvers with sufficient accuracy to conduct feasibility analysis of a mission (but not high fidelity mission designs). This trajectory could serve as a first result for further processing but at least supply a gravity-assist sequence that brings improvement regarding the optimization objectives, i.e. regarding the maximum possible payload i.e. minimum required propellant mass, resp. Δv. To address these topics, the research in this thesis concentrates on the following two aspects, which are also depicted in Figure 3-1:

First, after the mathematical description of gravity assists as established in Chapter 2, mathematical and physical considerations are formulated to assess the possibility of utilizing Tisserand's Criterion for the purpose of gravity-assist sequencing on low-thrust trajectories. Furthermore, it is assessed if and how the idea behind the Tisserand graphs has to be modified to make them applicable by introduction of a correction term. Currently Tisserand's Criterion is used for impulsive-maneuvers, errors already occur by diverting from the theoretical concept of a three-body system, mid-course maneuvers or powered-flyby maneuvers. Tisserand's Criterion is based on energy conservation and therefore it is investigated how large the violation of this principle caused by low-thrust propulsion is in comparison to errors introduced by the mentioned effects.

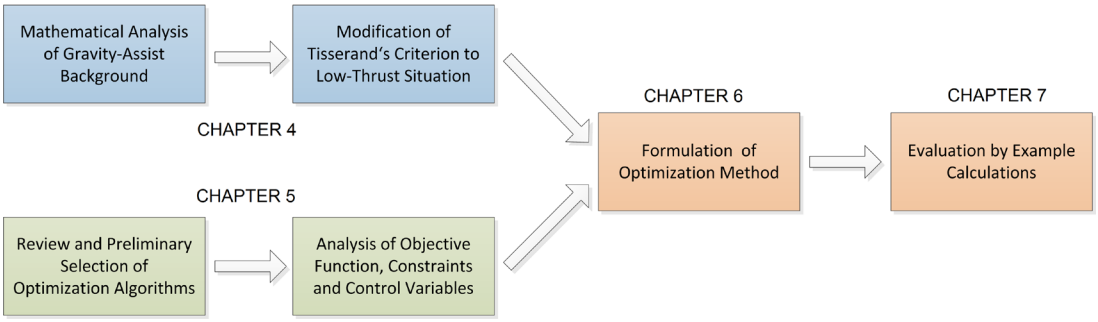

Figure 3-1: Path of the proceeding for the present thesis.

Once the mathematical aspects of the problem are sufficiently investigated and where necessary modified, the principle is embedded as far as possible into an optimization scheme, which allows the creation of optimized low-thrust gravity-assist trajectories. For this purpose, optimization algorithms are reviewed regarding their suitability and their ability to actually find such trajectories. As it is intended to find a global optimum, respective algorithms have to be used, e.g. evolutionary algorithms. Such algorithms are selected for further testing.

Furthermore, it has to be investigated which parameters are reasonable and effective for such a calculation. This concerns on one hand the properties of the gravity-assist sequence, e.g. the maximum number of encounters or minimum allowable distance to a given body and also demands regarding the timeframe, i.e. launch or arrival dates. This also affects the objective function that is used for the optimization.

The optimization must not concentrate on gravity-assist trajectories alone but also on regular non-gravity-assist trajectories to ensure that none exist which have a better performance regarding the objective function. The gravity-assist sequence has to provide improvement.

Therefore, two optimizations have to be conducted for given mission scenarios: One without gravity-assist maneuvers and one with. For the case of gravity-assist sequences, a scheme has to be derived; possible paths are shown in Figure 3-2: First (A), the optimization of a sequence of gravity-assist partners can be closely interlinked with trajectory optimization. Therefore, only the next partner is determined then the trajectory leg to this body is optimized. This is repeated until the trajectory is complete and the mission constraints have been fulfilled. Second (B), more analogous to Tisserand graphs, the sequence according to mission constraints is optimized and afterwards the trajectory for the whole mission, i.e. for all legs, is optimized as well. Furthermore, feasibility from a mission point of view (i.e. reckoning in e.g. thruster characteristics) cannot be incorporated into the sequencing. Therefore, it is possible that sequences cannot be fully followed in the implementation and thus a calculated sequence becomes useless or obsolete. The third option (C) is finding a sequence but only using it as a guideline for the trajectory calculation, where a new sequence can be calculated even during the trajectory optimization. So, once the next partner is reached it is investigated if the original sequence is still possible and advisable or if another option should be selected.

In general, due to technical constraints the gravity-assist sequence is not only subject to effects of natural background, e.g. phasing of the respective bodies, but also effects like target feasibility due to fuel mass or thrust magnitude.

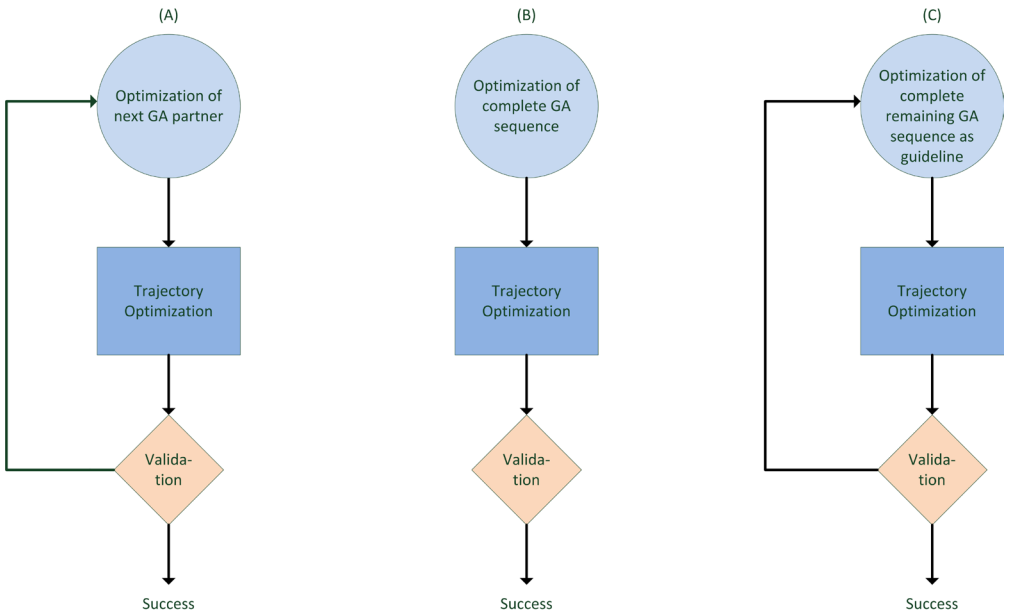

Figure 3-2: Overview over the alternatives of combining gravity-assist sequencing with actual trajectory optimization. A: Only the immediate next body is optimized regarding gravity-assist benefit, then the actual trajectory calculation is conducted and afterwards it is validated that the trajectory complies the mission constraints. B: The complete sequence is optimized a priori, and then the trajectory is calculated before validation. C: Knowledge of a sequence exists before trajectory calculation but it is not mandatory and serves only to guide the calculation.

Once the methodological scheme is established, the respective software tool is coded and tested to conduct the optimization. Furthermore, the effectiveness of the selected algorithms is reviewed and discussed, leading to an evaluation of the overall optimization method.

4 Applicability of Tisserand's Criterion for Continuous Low-Thrust Missions

This chapter is addressing the first two scientific questions of this thesis:

1. *Is it possible to optimize a gravity-assist sequence for low-thrust propulsion using Tisserand's Criterion in a similar way as for impulsive missions?*
2. *Which modifications of the method are required to regard energy changes due to the continuous thrusting of the low-thrust engine?*

These questions have to be answered because Tisserand's Criterion is based on the circular restricted three-body system where only gravitation is affecting the respective three bodies. This is violated by a scenario with low-thrust propulsion involved, as the spacecraft is typically continuously thrusting for a considerate amount of time during a given mission.

To analyze the thrust acceleration's impact on the overall situation and the applicability of Tisserand's Criterion, the background and equations used for deriving it are explained in this chapter. This allows an estimation of the error made by violating the basic assumptions of Tisserand's Criterion, i.e. those of the circular restricted three-body-system, occurring whenever Tisserand's Criterion is used in the actual solar system context, a deviation of the circular restricted three-body system. This error is then related to those caused by introducing the continouos low-thrust into the equations. Considering that Tisserand's Criterion as expressed by Eq. (2-8) has been successfully used in the real solar system situation despite deviations from the ideal circular restricted case, a similar or smaller order of magnitude of the error caused by the introduction of thrust would indicate applicability even in the low-thrust case. A larger order of magnitude would indicate that modifications are needed as a correction.

4.1 Multi-Body Problem Equations

The multi-body problem is formulated by applying Newton's law of universal gravitation, where the gravitational force of one body on the other is depending reciprocally on their distance to the power of two (Murray and Dermott, 2008). In this example the force on the mass *i* caused by mass *j* is given by (see Figure 4-1 for the respective vectors):

$$m_i \cdot \ddot{\vec{R}}_i = m_i \cdot m_j \cdot \Upsilon \cdot \frac{\vec{R}_{ij}}{R_{ij}^3}, \tag{4-1}$$

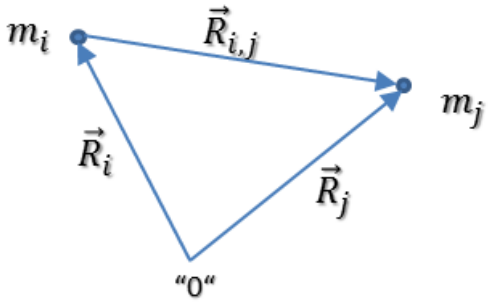

Figure 4-1: Position vector designation for a multi-body system.

where Y is the universal gravitational constant, m designates the masses i resp. j, R is the position vector for the mass with index i resp. between the two masses i and j. The latter is defined as:

$$\vec{R}_{ij} = \vec{R}_j - \vec{R}_i \tag{4-2}$$

A sketch of the involved masses and respective position vectors is given in Figure 4-1.

For more than two masses involved, the sum of all gravitational forces acting on mass i over n masses can be expressed by summing up their individual forces (Kemble, 2006):

$$m_i \cdot \ddot{\vec{R}}_i = Y \cdot m_i \cdot \sum_{j=1}^{n} m_j \cdot \frac{\vec{R}_{ij}}{R_{ij}^3}, for\ i \neq j \tag{4-3}$$

4.1.1 Conservation of Momentum

As the gravitational forces between the individual bodies of a system of masses are only internal forces, the momentum of the total system is constant. Since $\vec{R}_{ij} = -\vec{R}_{ji}$ the terms balance each other out, when summed up over all masses (Kemble, 2008):

$$\sum_{j=1}^{n} m_j \cdot \ddot{\vec{R}}_j = 0. \tag{4-4}$$

Integrating Eq. (4-4) over time yields (Kemble, 2006):

$$\sum_{j=1}^{n} m_j \cdot \dot{\vec{R}}_j = \vec{C}_1 \tag{4-5}$$

This equation describes the conservation of momentum, due the fact that there is no external force in the system of masses. Integrating Equation (4-5) again, results in:

$$\sum_{j=1}^{n} m_j \cdot \vec{R}_j = \vec{C}_1 \cdot t + \vec{C}_2, \tag{4-6}$$

which describes the position of the center of gravitation of the system of masses in time dependence (Kemble, 2006).

4.1.2 Conservation of Angular Momentum

The equations for the angular momentum of the system of masses are derived by creating the vector product of Equation (4-4) and the position vector $\vec{R}_j$ (Kemble, 2006):

$$\sum_{j=1}^{n} m_j \cdot \vec{R}_j \times \ddot{\vec{R}}_j = 0. \tag{4-7}$$

Integration over time provides the angular momentum:

$$\sum_{j=1}^{n} m_j \cdot \vec{R}_j \times \dot{\vec{R}}_j = \vec{C}_3. \tag{4-8}$$

Equation (4-8) shows that the angular momentum is constant and does not change over time.

4.1.3 Conservation of Energy

While the gravity-assist maneuver itself demands that the energy between the individual masses of a system cannot be constant, the total energy is conserved.

Multiplying Eq. (4-3) with $\dot{\vec{R}}_i$ yields (Alles, 2005):

$$m_i \cdot \dot{\vec{R}}_i \cdot \ddot{\vec{R}}_i = \Upsilon \cdot m_i \cdot \sum_{j=1}^{n} m_j \cdot \dot{\vec{R}}_i \cdot \frac{\vec{R}_{ij}}{R_{ij}^3}, for\ i \neq j \tag{4-9}$$

This can be summed up over all *i* masses:

$$\sum_{i=1}^{n} m_i \cdot \dot{\vec{R}}_i \cdot \ddot{\vec{R}}_i = \Upsilon \cdot \sum_{i=1}^{n} m_i \cdot \sum_{j=1}^{n} m_j \cdot \dot{\vec{R}}_i \cdot \frac{\vec{R}_{ij}}{R_{ij}^3}, for\ i \neq j \tag{4-10}$$

The time integral of the two vectors in Eq. (4-10)'s left side, is determined via integration by parts to:

$$\int \dot{\vec{R}}_i \cdot \ddot{\vec{R}}_i dt = \dot{\vec{R}}_i \cdot \dot{\vec{R}}_i - \int \dot{\vec{R}}_i \cdot \ddot{\vec{R}}_i dt \tag{4-11}$$

$$\Rightarrow 2 \int \dot{\vec{R}}_i \cdot \ddot{\vec{R}}_i dt = \dot{\vec{R}}_i^{\,2} \tag{4-12}$$

Using this for Eq. (4-10) the integrated left side becomes (Alles, 2005):

$$\int \sum_{i=1}^{n} m_i \cdot \dot{\vec{R}}_i \cdot \ddot{\vec{R}}_i \, dt = \sum_{i=1}^{n} \frac{m_i}{2} \cdot \dot{\vec{R}}_i^{\,2} = \sum_{i=1}^{n} \frac{m_i}{2} \cdot \vec{V}_i^2 \tag{4-13}$$

So, Eq. (4-13) is an expression for the kinetic energy summarized over all *i* masses of the problem. The right side of Eq. (4-10) can be rearranged to:

$$\Upsilon \cdot \sum_{i=1}^{n} \sum_{j=1}^{n} m_i \cdot m_j \cdot \dot{\vec{R}}_i \cdot \frac{\vec{R}_{ij}}{R_{ij}^3} = \Upsilon \cdot \sum_{i=1}^{n} m_i \sum_{j=1}^{n} m_j \cdot \dot{\vec{R}}_i \cdot \frac{\vec{R}_{ij}}{R_{ij}^3} = \Upsilon \cdot L \tag{4-14}$$

Using the definition of $\vec{R}_{ij}$ in Eq. (4-2) the term *L* can be further modified (Wakker, 2007):

$$L = \sum_{i=1}^{n} m_i \sum_{j=1}^{n} m_j \cdot (\dot{\vec{R}}_j - \dot{\vec{R}}_{ij}) \cdot \frac{\vec{R}_{ij}}{R_{ij}^3} \\ = \sum_{i=1}^{n} m_i \sum_{j=1}^{n} m_j \cdot \dot{\vec{R}}_j \cdot \frac{\vec{R}_{ij}}{R_{ij}^3} - \sum_{i=1}^{n} m_i \sum_{j=1}^{n} m_j \cdot \dot{\vec{R}}_{ij} \cdot \frac{\vec{R}_{ij}}{R_{ij}^3} \tag{4-15}$$

Because the summations are commutative in *i* and *j*, $\dot{\vec{R}}_j$ can be exchanged for $\dot{\vec{R}}_i$ in the first part of the term, but also for $\vec{R}_{ij}$ the indexes have to be exchanged, yielding $\vec{R}_{ji}$ as the order of the summations are switched. As $\vec{R}_{ij} = -\vec{R}_{ji}$, Eq. (4-15) can be rewritten as (it is noted that the magnitude of both vectors is equal and therefore needs no adaptation):

$$L = -\sum_{i=1}^{n} m_i \sum_{j=1}^{n} m_j \cdot \dot{\vec{R}}_i \cdot \frac{\vec{R}_{ij}}{R_{ij}^3} - \sum_{i=1}^{n} m_i \sum_{j=1}^{n} m_j \cdot \dot{\vec{R}}_{ij} \cdot \frac{\vec{R}_{ij}}{R_{ij}^3} \tag{4-16}$$

It can be seen that the first part of the right side is the same as the definition of *L* in Eq. (4-14) but with a negative sign. Therefore, the equation can be rearranged to:

$$2L = -\sum_{i=1}^{n} m_i \sum_{j=1}^{n} m_j \cdot \dot{\vec{R}}_{ij} \cdot \frac{\vec{R}_{ij}}{R_{ij}^3} \tag{4-17}$$

Analogously to Eq. (4-12) this can be further adapted to:

$$2L = -\sum_{i=1}^{n} m_i \sum_{j=1}^{n} \frac{m_j}{R_{ij}^3} \cdot \frac{1}{2} \frac{d}{dt} (\vec{R}_{ij}^2) = -\sum_{i=1}^{n} m_i \sum_{j=1}^{n} \frac{m_j}{R_{ij}^3} \cdot \frac{1}{2} \frac{d}{dt} (R_{ij}^2) \tag{4-18}$$

$$= -\sum_{i=1}^{n} m_i \sum_{j=1}^{n} \frac{m_j}{R_{ij}^3} \cdot R_{ij} \cdot \dot{R}_{ij} = -\sum_{i=1}^{n} m_i \sum_{j=1}^{n} \frac{m_j}{R_{ij}^2} \cdot \dot{R}_{ij} = \sum_{i=1}^{n} m_i \sum_{j=1}^{n} m_j \cdot \frac{d}{dt} \left(\frac{1}{R_{ij}} \right)$$

Therefore, the term for *L* finally becomes:

$$L = \frac{1}{2} \cdot \sum_{i=1}^{n} m_i \sum_{j=1}^{n} m_j \cdot \frac{d}{dt} \left(\frac{1}{R_{ij}} \right) \tag{4-19}$$

With this, the time integral of Eq. (4-10) can be written as (Wakker, 2007):

$$\sum_{i=1}^{n} \frac{m_i}{2} \cdot \vec{V}_i^2 - \frac{1}{2} \cdot \Upsilon \cdot \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{m_i \cdot m_j}{R_{ij}} = C_4, for\ i \neq j, \tag{4-20}$$

showing that the total energy of the system of masses is constant. For two masses, i.e. n = 2, this equation becomes the typical energy integral known for the two body system.

4.1.4 Equations of Relative Motion

In this section a formulation for the equations of relative motion of the masses is derived (Alles, 2005), which will be used to determine the magnitude of error in the three-body system caused by dropping other masses out of the consideration (thus diverting from the actual multi-body case in our solar system).

Reducing the mass *i* in Eq. (4-3) yields (Alles, 2005):

$$\ddot{\vec{R}}_i = \Upsilon \cdot \sum_{j=1}^{n} m_j \cdot \frac{\vec{R}_{ij}}{R_{ij}^3}, for\ i \neq j \tag{4-21}$$

as its equation of motion. Similarly, the equation for the mass *k* becomes:

$$\ddot{\vec{R}}_k = \Upsilon \cdot \sum_{j=1}^{n} m_j \cdot \frac{\vec{R}_{kj}}{R_{kj}^3}, for\ k \neq j \tag{4-22}$$

Referring to Figure 4-1 both these equations are related to the reference point "0". Furthermore, the motion of the mass *i* respective to *k* can be derived by subtracting Eq. (4-22) from (4-21) (Alles, 2005):

$$\ddot{\vec{R}}_{ki} = \ddot{\vec{R}}_i - \ddot{\vec{R}}_k = \Upsilon \cdot \left(\sum_{j=1}^{n} m_j \cdot \frac{\vec{R}_{ij}}{R_{ij}^3} - \sum_{j=1}^{n} m_j \cdot \frac{\vec{R}_{kj}}{R_{kj}^3} \right) \tag{4-23}$$

whereas for the first part of the bracketed term, $j \neq i$, and for the second $j \neq k$. To make this more consistent, the respective terms have to be excluded for each sum to make this a general condition for both sums:

$$\ddot{\vec{R}}_{ki} = \Upsilon \cdot \left(m_k \cdot \frac{\vec{R}_{ik}}{R_{ik}^3} - m_i \cdot \frac{\vec{R}_{ki}}{R_{ki}^3} + \sum_{j=1}^{n} m_j \cdot \frac{\vec{R}_{ij}}{R_{ij}^3} - \sum_{j=1}^{n} m_j \cdot \frac{\vec{R}_{kj}}{R_{kj}^3} \right), for\ j \neq k, i \tag{4-24}$$

It can be seen that the two single terms (outside the sums) are very similar and taking into account that $\vec{R}_{ik} = -\vec{R}_{ki}$ Eq. (4-24) can be reformulated to:

$$\ddot{\vec{R}}_{ki} = \Upsilon \cdot \left(-m_k \cdot \frac{\vec{R}_{ki}}{R_{ki}^3} - m_i \cdot \frac{\vec{R}_{ki}}{R_{ki}^3} + \sum_{j=1}^{n} m_j \cdot \frac{\vec{R}_{ij}}{R_{ij}^3} - \sum_{j=1}^{n} m_j \cdot \frac{\vec{R}_{kj}}{R_{kj}^3} \right) \tag{4-25}$$

$$\Leftrightarrow \ddot{\vec{R}}_{ki} + \Upsilon \cdot (m_k + m_i) \cdot \frac{\vec{R}_{ki}}{R_{ki}^3} = \Upsilon \cdot \left(\sum_{j=1}^{n} m_j \cdot \frac{\vec{R}_{ij}}{R_{ij}^3} - \sum_{j=1}^{n} m_j \cdot \frac{\vec{R}_{kj}}{R_{kj}^3} \right) = K\ , for\ j \neq k, i$$

The right side of the equation, *K*, is 0 in case of a two-body system and thus the equation becomes the usual two-body equation of relative motion (Murray and Dermott, 2008), also refer to Eq. (2-32). *K* can be further rearranged to remove the connection to the terms indexed with *i*. For this is has to be kept in mind that $\vec{R}_{ij} = \vec{R}_j - \vec{R}_i$ and $\vec{R}_{kj} = \vec{R}_j - \vec{R}_k$. Thus, $\vec{R}_j = \vec{R}_{kj} + \vec{R}_k$ and $\vec{R}_i = \vec{R}_{ki} + \vec{R}_k$. With this, it can be written that $\vec{R}_{ij} = \vec{R}_{kj} + \vec{R}_k - (\vec{R}_{ki} + \vec{R}_k) \Leftrightarrow \vec{R}_{ij} = \vec{R}_{kj} - \vec{R}_{ki}$.

This way *K* can be reformulated to:

$$K = \Upsilon \cdot \sum_{j=1}^{n} m_j \cdot \left(\frac{\vec{R}_{kj} - \vec{R}_{ki}}{R_{ij}^3} - \frac{\vec{R}_{kj}}{R_{kj}^3} \right), for\ j \neq k, i \tag{4-26}$$

And finally, the equation of relative motion between two masses in a multi-body system becomes:

$$\ddot{\vec{R}}_{ki} + \Upsilon \cdot (m_k + m_i) \cdot \frac{\vec{R}_{ki}}{R_{ki}^3} = \Upsilon \cdot \sum_{j=1}^{n} m_j \cdot \left(\frac{\vec{R}_{kj} - \vec{R}_{ki}}{R_{ij}^3} - \frac{\vec{R}_{kj}}{R_{kj}^3} \right), for\ j \neq k, i \tag{4-27}$$

For a multi-body system, the right side of the equation describes the influence that the remaining masses (besides *i* and *k*) have on those two masses (Alles, 2005).

4.2 Derivation of Tisserand's Criterion

In this section, Tisserand's Criterion is derived in detail to document where assumptions and approximations are used and later relate the error caused by violating these assumptions with the order of magnitude of deliberate errors during the derivation.

4.2.1 Hamiltonian of the Two-Body System

For the derivation of Tisserand's Criterion the usage of the Hamiltonian of the two-body system is required, which is derived in the following. It resembles an alternative formulation for the two-body system, but is based on the same physical law (Newton's law of universal gravitation) (Murray and Dermott, 2008).

Per definition a Hamiltonian is given by:

$$H = f(\vec{q}, \vec{p}, t), \text{ with} \tag{4-28}$$

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i} \quad \text{and} \quad \frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i} \tag{4-29}$$

Starting from Eq. (2-32), the equation of relative motion for the two-body system is expressed as:

$$\ddot{\vec{R}} + \mu \cdot \frac{\vec{R}}{R^3} = 0$$

For the three coordinates of $\vec{R}$ this means that the system contains three coupled, second order differential equations. The advantage of the Hamiltonian is that the motion can be described with six coupled, first order differential equations. (Murray and Dermott, 2008)

In the above equation, the relative position vector can be defined as (Murray and Dermott, 2008):

$$\vec{R} = r_x \cdot \vec{e}_x + r_y \cdot \vec{e}_y + r_z \cdot \vec{e}_z \tag{4-30}$$

where $\vec{e}_x$, $\vec{e}_y$ and $\vec{e}_z$ are unit vectors in each direction of the coordinate system. The linear momentum $\vec{p}$ is defined as (Murray and Dermott, 2008):

$$\vec{p} = \frac{m_1 \cdot m_2}{m_1 + m_2} \cdot \vec{v} = \mu^* \cdot \vec{v} = \mu^* \cdot \dot{\vec{R}} = p_x \cdot \vec{e}_x + p_y \cdot \vec{e}_y + p_z \cdot \vec{e}_z \tag{4-31}$$

Here $\mu^* = \frac{m_1 \cdot m_2}{m_1 + m_2}$ is the reduced mass of the two-body system (Murray and Dermott, 2008). Further, the Keplerian Hamiltonian is (Murray and Dermott, 2008):

$$H_k = \frac{p^2}{2\,\mu^*} - \frac{\mu \cdot \mu^*}{r}, \tag{4-32}$$

with:

$$\nabla_p = \frac{\partial}{\partial\, p_x} \cdot \vec{e}_x + \frac{\partial}{\partial\, p_y} \cdot \vec{e}_y + \frac{\partial}{\partial\, p_z} \cdot \vec{e}_z \tag{4-33}$$

$$\nabla_r = \frac{\partial}{\partial\, r_x} \cdot \vec{e}_x + \frac{\partial}{\partial\, r_y} \cdot \vec{e}_y + \frac{\partial}{\partial\, r_z} \cdot \vec{e}_z \tag{4-34}$$

Equation (4-29) must be true for the Keplerian Hamiltonian, which leads to:

$$\dot{\vec{p}} = \nabla_r \cdot H_k = -\frac{\mu \cdot \mu^*}{R^3}\vec{R} \tag{4-35}$$

$$\dot{\vec{R}} = \nabla_p \cdot H_k = \frac{\vec{p}}{\mu^*} \tag{4-36}$$

Equation (4-36) can be proven by Eq. (4-31) directly and Eq. (4-35) by differentiating it after time. So, Eq. (4-32) is in fact a Hamiltonian.

Inserting Eq. (4-31) into (4-32), including the definition of μ^* then reveals:

$$H_k = \frac{v^2}{2} \cdot \mu^* - \frac{\mu \cdot \mu^*}{r} = \varepsilon \cdot \mu^*, \tag{4-37}$$

here ε is the specific energy of the motion. It can be seen that the Hamiltonian is simply the energy equation of the two-body problem (i.e. the sum of kinetic and potential energy), scaled with the reduced mass μ^*.

4.2.2 Circular Restricted Three-Body System Equations of Motion and Pseudo-Potential

The restricted three-body system is a reduction of the multi-body system to three bodies, where one mass is infinitely smaller than the other two and one mass is dominant over the others in general. If the problem is also considered *circular* all orbits of the masses are assumed to be circles and therefore have an eccentricity of 0 and a constant angular velocity. This approximation is only a rough representation of conditions in the actual solar system, but allows estimates and the formulation of the Jacobi integral. This constant of the restricted three-body system (Murray and Dermott, 2008), which is required for the formulation of Tisserand's Criterion.

The concept of this is visually represented in Figure 4-2. Two coordinate systems are used, one inertial (ξ and η) and one rotating (x and y), whereas "0" is the center of mass of the system. The distance between m_1 and m_2 is constant (a consequence of the circular orbit). For the equations of motion it is further assumed that $\mu = \Upsilon \cdot (m_1 + m_2) = 1$ and that $\bar{\mu} = \frac{m_2}{m_1 + m_2}$. Therefore, for both major masses it can be written that (Murray and Dermott, 2008):

$$\mu_1 = \Upsilon \cdot m_1 = \mu - \Upsilon \cdot m_2 = \mu - \frac{\mu \cdot m_2}{m_1 + m_2} = 1 - \bar{\mu} \tag{4-38}$$

$$\mu_2 = \Upsilon \cdot m_2 = \bar{\mu} \tag{4-39}$$

It is also assumed that the distance between m_1 and m_2 is unity and following that, the normalized mean motion of the two masses, n, is unity, either.

With this, the equations of motion in the inertial coordinate system become (Murray and Dermott, 2008):

$$\ddot{\xi} = \mu_1 \frac{\xi_1 - \xi}{R_1^3} + \mu_2 \frac{\xi_2 - \xi}{R_2^3} \tag{4-40}$$

$$\ddot{\eta} = \mu_1 \frac{\eta_1 - \eta}{R_1^3} + \mu_2 \frac{\eta_2 - \eta}{R_2^3} \tag{4-41}$$

$$\ddot{\zeta} = \mu_1 \frac{\zeta_1 - \zeta}{R_1^3} + \mu_2 \frac{\zeta_2 - \zeta}{R_2^3} \tag{4-42}$$

where R_1 is defined as:

$$R_1 = \sqrt{(\xi_1 - \xi)^2 + (\eta_1 - \eta)^2 + (\zeta_1 - \zeta)^2} \tag{4-43}$$

and R_2 analogously.

The same equations can be formulated for the rotating coordinate system, which then include terms for the Coriolis acceleration and the centrifugal acceleration. They can be found in (Murray and Dermott, 2008).

With the definition of a pseudo-potential U (Murray and Dermott, 2008):

$$U = \frac{n^2}{2}(x^2 + y^2) + \frac{\mu_1}{R_1} + \frac{\mu_2}{R_2}, \tag{4-44}$$

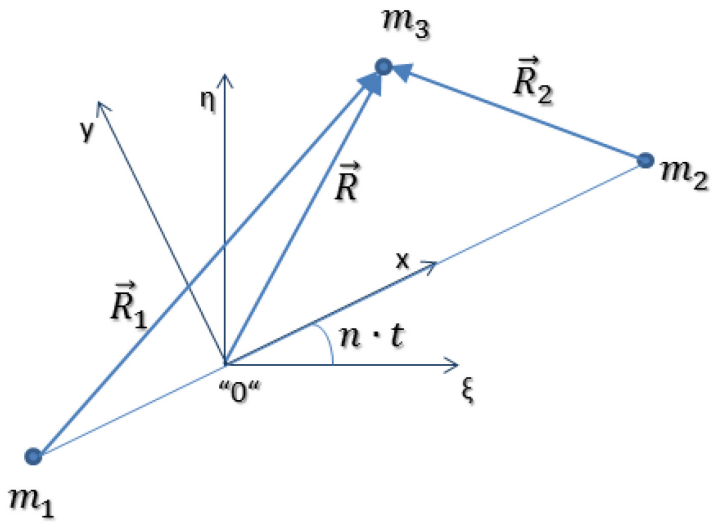

Figure 4-2: The situation in the restricted three-body system based on (Murray and Dermott, 2008).

where the first part of the term is the centrifugal potential and the two last fractions comprise the gravitational potential, this becomes (Murray and Dermott, 2008):

$$\ddot{x} - 2\,n\,\cdot\dot{y} = \frac{\partial U}{\partial x}, \tag{4-45}$$

$$\ddot{y} + 2\,n\,\cdot\dot{x} = \frac{\partial U}{\partial y}, \tag{4-46}$$

$$\ddot{z} = \frac{\partial U}{\partial z}. \tag{4-47}$$

Because not all acceleration terms for a small mass in the restricted three-body system can be derived with this potential, it is only a pseudo-potential. This pseudo-potential is used for the derivation of the Jacobi-Integral, a constant in the restricted three-body system.

4.2.3 The Jacobi Integral – Constant of the Restricted Three-Body System

The Jacobi Integral is an integral of the relative energy of the motion in a restricted three-body system and a constant for this motion, it is based on the pseudo-potential described earlier. It can be derived from Eq. (4-45) to (4-47), by multiplying each of these equations with the time derivate of their matching coordinate (e.g. Equation (4-45) with $\dot{x}$) and adding all three equations together (which is analogue to the steps in Section 4.1.3). (Murray and Dermott, 2008)

This yields:

$$\dot{x}\,\ddot{x} - 2\,n\,\dot{x}\,\dot{y} - \frac{\partial U}{\partial x}\,\dot{x} + \dot{y}\,\ddot{y} + 2\,n\,\dot{x}\,\dot{y} - \frac{\partial U}{\partial y}\,\dot{y} + \dot{z}\,\ddot{z} - \frac{\partial U}{\partial z}\,\dot{z} = 0 \tag{4-48}$$

Rearranged this becomes the total time differential of the pseudo-potential (Murray and Dermott, 2008):

$$\dot{x}\,\ddot{x} + \dot{y}\,\ddot{y} + \dot{z}\,\ddot{z} = \frac{\partial U}{\partial x}\,\frac{dx}{dt} + \frac{\partial U}{\partial y}\,\frac{dy}{dt} + \frac{\partial U}{\partial z}\,\frac{dz}{dt} = \frac{dU}{dt} \tag{4-49}$$

This can be integrated in time in a similar fashion as Equation (4-11), which leads to (Murray and Dermott, 2008):

$$\dot{x}^2 + \dot{y}^2 + \dot{z}^2 = V^2 = 2\,U - C_j, \tag{4-50}$$

where C_j is the Jacobi constant and V the velocity. Using the definition of U, this provides the definition of the Jacobi Integral for the rotating coordinate system (Murray and Dermott, 2008):

$$C_j = n^2(x^2 + y^2) + 2\left(\frac{\mu_1}{R_1} + \frac{\mu_2}{R_2}\right) - (\dot{x}^2 + \dot{y}^2 + \dot{z}^2) \tag{4-51}$$

With a coordinate transformation, the formulation can also be expressed in the non-rotating coordinates (Murray and Dermott, 2008):

$$C_j = 2\,n\left(\xi\,\dot{\eta} - \eta\,\dot{\xi}\right) + 2\left(\frac{\mu_1}{R_1} + \frac{\mu_2}{R_2}\right) - (\dot{\xi}^2 + \dot{\eta}^2 + \dot{\zeta}^2) \tag{4-52}$$

The Jacobi Integral is no real energy integral as momentum and energy are not conserved in the restricted three body system. This is because the assumed mass-less nature of the third mass

prevents it from causing changes in the planetary orbit (which is not true as in reality energy is transferred from the large to the small mass during an encounter or vice versa). (Murray and Dermott, 2008)

4.2.4 Tisserand's Criterion Following from the Jacobi Integral

The Jacobi Integral is also used for the derivation of Tisserand's Criterion. Originally, Tisserand used it to identify comets he observed. Throughout the encounter of a comet with (in Tisserand's case only) Jupiter the Jacobi Integral remains constant, which therefore is the basis for Tisserand's Criterion.

It should be noted that in this case thrust is not assumed (only gravitational forces) so any thrust is already a violation of Tisserand's Criterion.

To describe the small mass one position and one velocity vector are needed (Murray and Dermott, 2008):

$$\vec{R} = \begin{pmatrix} \xi \\ \eta \\ \zeta \end{pmatrix} \tag{4-53}$$

$$\dot{\vec{R}} = \begin{pmatrix} \dot{\xi} \\ \dot{\eta} \\ \dot{\zeta} \end{pmatrix}. \tag{4-54}$$

In this case using the inertial coordinates, the Jacobi Constant becomes:

$$C_j = 2\,n\,(\xi\,\dot{\eta} - \eta\,\dot{\xi}) + 2\left(\frac{\mu_1}{R_1} + \frac{\mu_2}{R_2}\right) - (\dot{\xi}^2 + \dot{\eta}^2 + \dot{\zeta}^2), \tag{4-55}$$

with R_1 referring to the distance between the small mass and the Sun and R_2 measuring the distance between the small body and the planetary body, e.g. Jupiter (Murray and Dermott, 2008), see Figure 4-2. Furthermore, it is assumed that the angular velocity of the planetary body is unity as is the semi-major axis (this scaling is used for easier calculations). In addition, the mass of the Sun is assumed to be dominant over the two other masses, i.e.: (Murray and Dermott, 2008)

$$\Upsilon(m_{sun} + m_{small}) \approx \Upsilon(m_{sun} + m_{planet}) = \mu_1 \approx 1, \tag{4-56}$$

here m_{small} is the mass of the small body and m_{planet} the planetary body's mass and μ_1 the gravitational parameter for the system of the Sun and the small mass. While originally formulated for the case of comets and Jupiter, the same is true for a spacecraft and a planet.

For the two-body system the energy integral for this case is:

$$V^2 = \dot{\xi}^2 + \dot{\eta}^2 + \dot{\zeta}^2 = \mu_1\left(\frac{2}{R_1} - \frac{1}{a}\right) \tag{4-57}$$

As the mass of the small mass is dominated by the Sun's mass, the distance of the small mass to the Sun is almost identical to the distance of the small mass and the common barycenter, therefore $R_1 \approx R$. In addition, as seen in Eq. (4-56), the gravitational parameter is unity, which yields (Murray and Dermott, 2008):

$$\dot{\xi}^2 + \dot{\eta}^2 + \dot{\zeta}^2 = \left(\frac{2}{R} - \frac{1}{a}\right) \tag{4-58}$$

Per definition, the specific angular momentum of this two-body system is (Murray and Dermott, 2008):

$$\vec{h} = \vec{R} \times \dot{\vec{R}} = \begin{pmatrix} \eta\,\dot{\zeta} - \zeta\,\dot{\eta} \\ \zeta\,\dot{\xi} - \xi\,\dot{\zeta} \\ \xi\,\dot{\eta} - \eta\,\dot{\xi} \end{pmatrix} = \begin{pmatrix} h_\xi \\ h_\eta \\ h_\zeta \end{pmatrix} \tag{4-59}$$

Derived from the fact that the vector of the angular momentum is always perpendicular to the orbit plane, it follows that (i being the inclination):

$$h_\zeta = h\cos(i) = \xi\,\dot{\eta} - \eta\,\dot{\xi}, \tag{4-60}$$

With this formulation of the two-body system (Murray and Dermott, 2008):

$$h^2 = a\,(1 - e^2), \tag{4-61}$$

and using the previous assumptions and Equations (4-58), (4-60) and (4-61), the Jacobi Integral for this case becomes:

$$C_j = 2\,n\left(\xi\,\dot{\eta} - \eta\,\dot{\xi}\right) + 2\left(\frac{1}{R} + \frac{\mu_2}{R_2}\right) - \left(\dot{\xi}^2 + \dot{\eta}^2 + \dot{\zeta}^2\right) = 2\left(\frac{1}{R} + \frac{\mu_2}{R_2}\right) + 2\,h_\zeta - V^2 \tag{4-62}$$

$$= 2\left(\frac{1}{R} + \frac{\mu_2}{R_2}\right) + 2\,h\cos(i) - \left(\frac{2}{R} - \frac{1}{a}\right)$$

$$= 2\left(\frac{1}{R} + \frac{\mu_2}{R_2}\right) + 2\sqrt{a\,(1 - e^2)}\cos(i) - \frac{2}{R} + \frac{1}{a}$$

$$= \frac{1}{a} + 2\frac{\mu_2}{R_2} + 2\sqrt{a\,(1 - e^2)}\cos(i)$$

V^2 has been replaced with Eq. (4-58), describing the situation of the Sun-satellite system before and after the encounter. Assuming further that the small mass' distance to the planetary body is large and the mass of the planet is small compared to the Sun's, i.e. $2\frac{\mu_2}{R_2} \ll \frac{1}{a}$, the terms containing R_2 resp. μ_2 are neglected, which leads to (Murray and Dermott, 2008):

$$\frac{C_j}{2} = const = \frac{1}{2\,a} + \sqrt{a\,(1 - e^2)}\cos(i) \tag{4-63}$$

$$\Leftrightarrow \frac{1}{2\,a} + \sqrt{a\,(1 - e^2)}\cos(i) = \frac{1}{2\,a'} + \sqrt{a'\,(1 - e'^2)}\cos(i')$$

This (dimensionless) formulation is important as it provides a functional connection between the values of a, i and e before and after (index ') a gravity assist. Tisserand used this to compare the constants and thus determine whether two objects he had observed where in fact the same before and after an encounter with Jupiter.

4.3 Effect of Violating the Assumptions of the Derivation and Correction Term

After reviewing the formulations necessary for the derivation of Tisserand's Criterion, this section describes the differences between the assumptions made during that derivation and the real situation in the solar system and especially in a low-thrust case. Estimates of the derivation's magnitudes are presented. A list of the underlying assumptions, their violations and the way of estimating their magnitudes is given in Table 4-1.

Table 4-1: Deviation between assumptions during the derivation of Tisserand's-Criterion and the physical conditions and situation.

Description	Occurrence	Estimation
Mass of all bodies in the multi-body system is assumed as constant, however the spacecraft changes its mass due to propellant usage	Eq. (4-1) and following, basic equations of the multi-body system	Effects due to mass change is attributed to thrust and the thrust acceleration in turn related to the gravity acceleration to identify areas of dominance
Distance between the large masses are assumed as constant (circular orbit), but in reality the orbits are eccentric (resp. the angular velocity is assumed to be constant)	General assumption for the restricted three-body system, see Chapter 4.2.2	Model based comparison of Tisserand Parameters in a circular system model and a non-circular system model
The mass of the comet/ spacecraft and the mass of the Sun are regarded as equal to the mass of the planet and the Sun, i.e. the solar mass is significantly dominant; an analogue simplification is made for the distances, where the distance between the comet/ spacecraft and the Sun is considered to be the same as the distance to the barycenter	Derivation of Tisserand's Criterion in Eq. (4-56).	Directly creating mass ratio
The distance between the spacecraft and the planet is assumed to be large, which is not always true, especially not during or near the encounter	Eq. (4-62) to (4-63)	Determine for which radius R_2 fulfills the expression $\frac{1}{a} = 2\frac{\mu_2}{R_2}$, see Eq. (4-62) for details

A more detailed analysis of these potential sources for errors is provided in the following sections and at the end a correction term for the introduction of thrust effects into Tisserand's Criterion is presented.

4.3.1 Non-constant Spacecraft Mass

For the formulation of the multi-body equations of motions, it is assumed that all masses in the system are constant over time, i.e. forces as time derivate of impulse are not caused by a time dependent mass. This is an accurate description for natural solar system bodies, but becomes inaccurate once spacecraft are involved that reduce their mass by using propellant for propulsion.

Dawn is a primary example of a low-thrust spacecraft on an interplanetary mission. To investigate the effect of neglecting the mass flow on the spacecraft, it is used as an example with the parameters as listed in Table 4-2.

With the given thrusting time and fuel mass, it can be calculated that the average mass flow for the fuel is 2.36 mg/s, which is the rate the spacecraft mass changes during the operation of the thruster. Assuming linear correlation of the mass flow and the thrust, the mass flow is at a minimum of 0.85 mg/s for a thrust of 19 mN and at a maximum of 4.07 mg/s at a thrust of 91 mN.

Table 4-2: *Dawn*'s parameters (NASA, 2007) used to compare its ratio of gravitation and thrust acceleration for error estimation.

Thrust Level	19 to 91 mN (NASA, 2007)
Launch Mass	1217.7 kg (NASA, 2007)
Fuel Mass (Xenon)	425 kg (NASA, 2007)
Thrusting Duration	2000 days (estimated) (NASA, 2007)
Average Fuel Mass Flow	2.36 mg/s*
MinimumFuel Mass Flow	0.85 mg/s* (at thrust of 19 mN)
Maximum Fuel Mass Flow	4.07 mg/s (at thrust of 91 mN)

*calculated with available numbers

In the multi-body equation, the spacecraft mass can be cancelled out of the equation:

$$m_{sc} \cdot \ddot{\vec{R}}_{sc} = \Upsilon \cdot m_{sc} \cdot \sum_{j=1}^{n} m_j \cdot \frac{\vec{R}_{scj}}{R_{scj}^3}, \tag{4-64}$$

$$\Leftrightarrow \ddot{\vec{R}}_{sc} = \Upsilon \cdot \sum_{j=1}^{n} m_j \cdot \frac{\vec{R}_{scj}}{R_{scj}^3}$$

which erases effects caused by a changing mass of the spacecraft. However, the mass flow as such has an impact on the forces acting on the spacecraft. Its effect is the thrust.

Thus, to accommodate these forces and the fact that the spacecraft mass changes, the equation has to be adapted, based on the momentum of the spacecraft and exhaust (with exhaust velocity *c*):

$$m_{sc} \cdot \ddot{\vec{R}}_{sc} = \vec{F}_{gravitation} + \vec{F}_{thrust} = \Upsilon \cdot m_{sc} \cdot \sum_{j=1}^{n} m_j \cdot \frac{\vec{R}_{scj}}{R_{scj}^3} + \dot{m}_{sc} \cdot c \tag{4-65}$$

This equation lists the gravitation forces acting on the spacecraft as well as the thrust force. The mass flow $\dot{m}_{sc}$ of the spacecraft is negative, representing the gas exhaust, which reduces the spacecraft mass.

To analyze the effect that the term containing the mass flow has on the overall force balance, the sample parameters from *Dawn* as given above are inserted into the equation. As the thrust force is the result of the mass flow exiting the spacecraft, it is investigated at which point the thrust becomes equal in magnitude to the gravitational forces. As the spacecraft mass is not considered to be constant, the acceleration is analyzed, i.e. the acting forces divided by the spacecraft mass.

Four cases are investigated:

1) Assuming minimum thrust (19 mN), constant during whole flight (case "min")
2) Assuming maximum thrust (91 mN), constant during the whole flight (case "max")
3) Assuming average thrust (55 mN), constant during the whole flight (case "av")
4) Thrust assumed to be proportional to the available spacecraft power, which also is reduced with $1/r^{1.8}$, the exponent of 1.8 is used to accommodate increase in solar cell efficiency due to the lower temperature at large solar distances (Löb, et al., 2007). The starting value is again 91 mN, the realistic case of a cut-off due to decrease of spacecraft power below minimum required power is ignored, i.e. the thrust can fall below *Dawn*'s actual minimum thrust (case "real")

Creating the ratio of the gravitational acceleration to thrust acceleration shows the effects caused by the mass flow of the spacecraft. Due to the prominence of the Sun and its role as central body, in

contrast to the undetermined position of the spacecraft relative to the planetary body, only the Sun's gravitation is regarded for this analysis. Figure 4-3 shows the result of this calculation.

It should be noted that *Dawn* is not designed to fly past a solar distance of Ceres (max. 3 AU), but picking up a realistic spacecraft design to estimate the acting forces is considered reasonable by the author. Because more accurate data about the mass flow of *Dawn* and its relation to the solar distance is not available to the author, a more accurate estimate was not possible.

In Figure 4-3, it can be seen that the ratio of gravitational acceleration to thrust acceleration is always significant (> 100). Even in the worst case scenario, i.e. for the assumption of a constant thrust of 91 mN, the gravitational acceleration is a factor of ca. 900 larger than the resulting thrust acceleration at the end of the investigated interval of solar distance (10 AU). The ratio is even larger for the other thrust approximations, showing that for a typical low-thrust mission, the gravitational acceleration usually dominates the force balance as presented in Eq. (4-65).

This means for a typical low-thrust mission the force resulting from the mass flow is indeed negligible for the instantaneous balance of forces. The energy change caused by the low thrust is not part of this analysis and is discussed in Chapter 4.3.5.

It should be noted that one major exception has not been regarded yet: Libration or Lagrange Points. At these positions the gravitational accelerations of planet(s) and Sun equal each other out so that only the thrust acceleration would remain. These areas are small in comparison to the whole trajectory, but it is still possible that the spacecraft passes through them. However, as Tisserand's Criterion is an energy quantity and not a forces quantity, these exceptions are not considered as critical, if at all, affecting only small parts of the trajectory.

4.3.2 Non-Circular Orbits for Major Masses

The basis for the derivation of Tisserand's Criterion is the restricted circular three-body system, where one of the major simplifications is that the planet's and Sun's orbits around the common

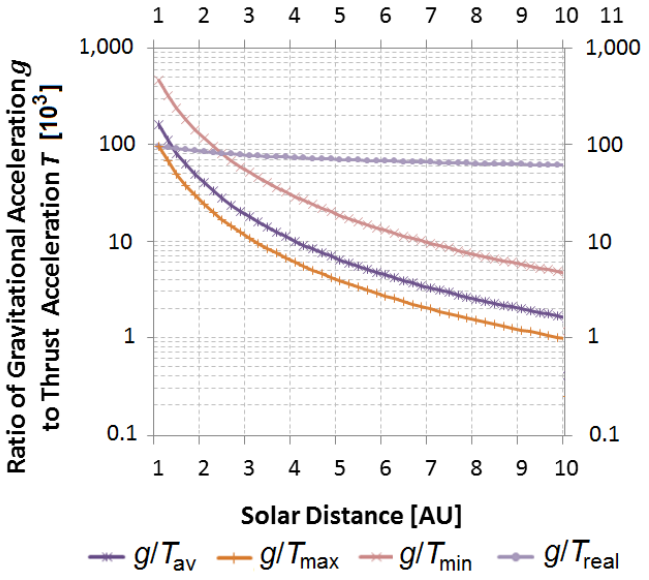

Figure 4-3: The ratio between the gravitational acceleration g caused by the Sun and thrust accelerations T of a *Dawn* analogue thruster as function of Solar Distance for four cases: average thrust (55 mN), maximum thrust (91 mN), minimum thrust (19 mN) and realistic thrust acceleration (91 mN/$r^{1.8}$).

barycenter are circular. In reality this is not the case. The planets' eccentricities range from 0.0068 (Venus) to 0.2056 (Mercury) (Bakich, 2000).

Reviewing Eq. (2-9), it is clear that a change of the distance between planet and Sun, also has an influence on the value of Tisserand's Criterion:

$$\frac{1}{a_1} + 2\sqrt{\frac{a_1(1-{e_1}^2)}{{R_{pl}}^3}}\cos i_1 \approx \frac{1}{a_2} + 2\sqrt{\frac{a_2(1-{e_2}^2)}{{R_{pl}}^3}}\cos i_2$$

R_{pl} is this distance. While the changes are small, the cubic exponent increases its dominance when compared with other parts of the term.

To analyze the impact of the effect of the planet's eccentricity on Tisserand Criterion, experiments have been conducted by creating models of a circular and non-circular three-body system respectively. Using both models calculations have been conducted to compare the effects of planetary flybys by a sample comet/ spacecraft first in the circular and then in the non-circular model. The variable of the spacecraft has been its original semi-major axis (before the flyby), and its hyperbolic excess velocity v_∞ with regards to the planet. An encounter was assumed once the spacecraft reached the sphere of influence, creating detectable changes in the spacecraft trajectory. Following these flyby calculations, the affected spacecraft's value of Tisserand's Criterion has been calculated and compared by determining the ratio of the two values for Tisserand's Criterion sets in the following manner:

$$T_\Delta = \frac{\left|T_{circular} - T_{elliptical}\right|}{T_{elliptical}} \cdot 100\%, \tag{4-66}$$

where T_Δ is the deviation of Tisserand's Criterion, and the indexes *circular* and *elliptical* denote Tisserand's Criterion values after the encounter in the circular resp. non-circular system model.

A number of 150 experiments were conducted with the results given in Figure 4-4. Each sample has been conducted with one set of parameters (i.e. a and v_∞) for the sample spacecraft, and the Sun as central body and one of the giant planets as the second body (due to their large mass) and Earth for comparison. For each sample it was assumed that the spacecraft was on a trajectory in the orbit plane of the planet, to eliminate effects due to inclination differences.

Investigating the depicted deviations, a correlation to the value of eccentricity is not visible. The error percentage appears to be independent of the eccentricity of the respective planet that has been encountered during the calculation.

However, it is apparent that the deviation can achieve very large ratios (up to 25%) even in the relatively small set of samples taken. One contribution for this is the effect of different values for the variable R_{pl} in Eq. (2-9). At the same time, the application of actual orbit data for the system model creates situations where planetary encounters do not occur with the same relative position between the spacecraft and the planet and therefore change how the spacecraft's orbit is affected and thus the corresponding value of Tisserand's Criterion (the experiments only evaluated cases where encounters occurred in both systems).

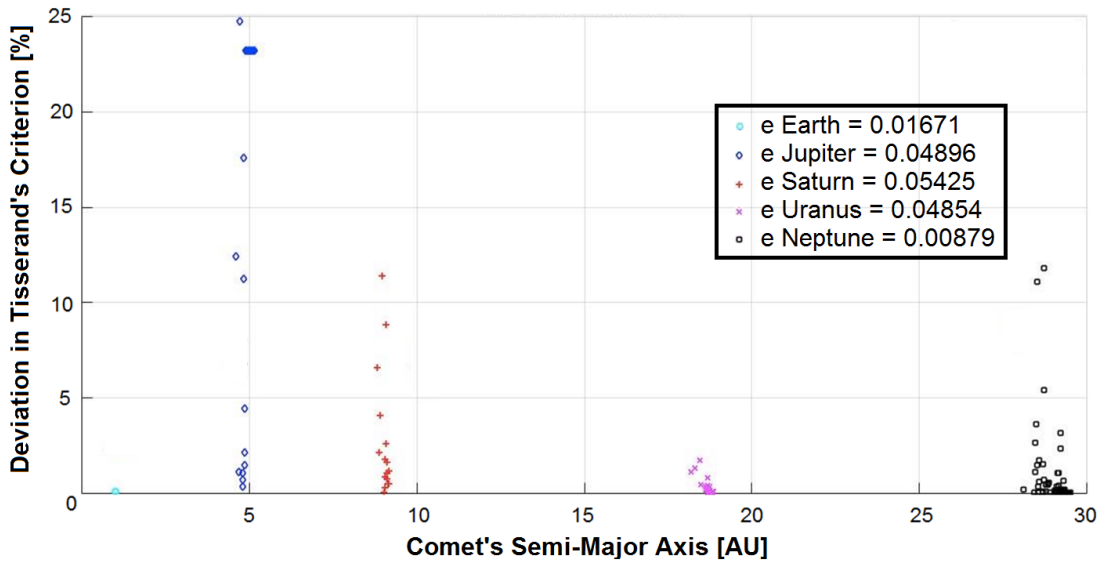

Figure 4-4: The ratio of deviation between values of Tisserand's Criterion after a comet's planetary encounter in a circular three-body system and a non-circular one. Each dot represents the comparison between the *circular* and *non-circular* encounter, calculated with identical a and v_∞.

The observed changes in Tisserand's Criterion reach significant scales (>10%) and due to the sampling method used it is conceivable that even larger values of deviations are possible, depending on the sample spacecraft to be used. In any case, the experiment described here shows that not in all cases the effect due to the non-circular nature of planetary orbits can be neglected when determining Tisserand's Parameter after a planetary encounter.

4.3.3 Negligible Spacecraft Mass

In Eq. (4-56) it is assumed that the spacecraft or comet's mass is negligible compared to the solar mass. Picking the equation up and calculating its actual value for the case of using *Dawn's* mass, the ratio of the sum of masses becomes

$$\frac{(m_{sun} + m_{sc})}{(m_{sun} + m_{jupiter})} = 0.999046 \tag{4-67}$$

The difference is only 0.095% of the assumed value of unity, therefore supporting the idea that the spacecraft mass is in fact negligible to the solar mass. As the mass distribution in the three-body system is also directly related to the barycenter of the system, this result also supports the assumption that the distance of the spacecraft to the Sun and the distance between spacecraft and barycenter are equal.

4.3.4 Small Distances between Planet and Spacecraft

During the derivation of Tisserand's Criterion, see Eq. (4-62), it has been assumed that the distance between the spacecraft (resp. comet) and the planetary body is very large and therefore the term containing R_2 is small. In this section it is investigated what conditions have to be met for this to be true and what happens, when this is not the case. How large is the resulting error magnitude? Beginning with Eq. (4-62), it is investigated what condition must be fulfilled so that the orders of magnitude of the two first parts, the fractions, of the equation become equal.

$$C_j = \frac{1}{a} + 2\frac{\mu_2}{R_2} + 2\sqrt{a\,(1-e^2)}\cos(i) \tag{4-68}$$

This occurs for:

$$\frac{1}{a} \overset{!}{=} 2\frac{\mu_2}{R_2}$$

$$\Rightarrow R_2 \overset{!}{=} 2\frac{\mu_2}{1}a$$

As μ_1 is equal to 1 in this formulation, this can be written as:

$$R_2 \overset{!}{=} 2\frac{\mu_2}{\mu_1}a = 2\frac{m_2}{m_1}a \tag{4-69}$$

So, for the term involving μ_2 to become larger or equal to the first part of the equation, the distance between the spacecraft and the planet needs to be smaller or equal to the mass ratio of the two involved bodies times the semi-major axis of the spacecraft:

$$R_2 \overset{!}{\leq} = 2\frac{m_2}{m_1}a \tag{4-70}$$

Using the example of the Sun (m_1 = 1.989 x 10^{30} kg (Murray and Dermott, 2008)) and Jupiter (m_2 = 1.899 x 10^{27} kg, a = 5.2 AU (Murray and Dermott, 2008)), the value for R_2 would become 1,485,440 km. Taking into account the sphere of influence, defined as (Prussing and Conway, 1993):

$$r_{SOI} \approx \left(\frac{m_p}{m_s}\right)^{2/5} \cdot r_{sp}, \tag{4-71}$$

where the index p denotes the planet and s the Sun and r_{sp} is the distance between the two bodies. With the same values as above, the sphere of influence of Jupiter has a radius of 48.2 million kilometers (factor 32.5 larger than the previously calculated value of R_2), which means in the above example R_2 lies well within Jupiter's sphere of influence. At this point Jupiter's gravitational influence on the spacecraft or comet dominates over the Sun's influence and its trajectory should not be described with a heliocentric formulation. It is apparent that the assumption of a large distance and therefore negligible term compared to the other parts of Eq. (4-62) is true for this example as the limit lies well within the sphere of influence (i.e. for a position, where the Tisserand Criterion is not applicable).

To investigate the situation in the solar system two sets of calculations for all planets have been conducted: One with varying the eccentricity of an example spacecraft's orbit (for a constant semi-major axis of 1.5 times the semi-major axis of the corresponding planet) in steps of 0.01 from 0.01 to 0.99. The other with varying the semi-major axis at a constant eccentricity (of 0.8 to ensure that the spacecraft and planet orbits intersect) from the respective planet's semi-major axis to a final semi-major axis of 5.1 times the initial semi-major axis. The distances have been scaled with the planet's solar distance, resp. the Sun's standard gravitational parameter μ_{Sun}.

These calculations were related to the nominal definition of Tisserand's Criterion, i.e. Eq. (4-63), and a comparison between Eq. (4-62) and (4-63), creating an estimate for the error caused by the omission of the planet in the derivation of Eq. (4-63). Planetary data has been used as given in (Murray and Dermott, 2008). These calculations show that the error is marginal.

The largest difference between the C_j containing the fraction $2\frac{\mu_2}{R_2}$ and omitting it has been determined for a very large eccentricity of 0.99 and for Jupiter. In this case the error is 3%. For most of the calculations, the error has been very small in the range of some hundredth percent.

It is therefore regarded as accurate that even a distance between spacecraft and planet of just the sphere of influence radius does not result in a considerable error.

4.3.5 Thrust Effect on Energy

Tisserand's Criterion is an energy quantity and therefore the impact of the thrust regarding the orbital energy of the spacecraft should be considered as well. Based on the assumption that for a given moment the spacecraft is on a trajectory that represents a Keplerian orbit (with altering properties for each time step, caused by the changes due to thrust), its mass specific orbital energy at that time t is expressed as:

$$\varepsilon(t) = -\frac{\mu}{2\,a(t)}. \tag{4-72}$$

This is the orbital energy of a Keplerian orbit (Prussing and Conway, 1993) with an added dependency on time, i.e. it is expressed as a series of osculating orbits.

Assuming, for instance, a mission going from Earth's solar distance to Jupiter's, i.e. from 1 AU to ca. 5 AU, this would mean a change of the specific energy of:

$$|\varepsilon_2 - \varepsilon_1| = |-\frac{\mu}{2 \cdot 5\ \mathrm{AU}} + \frac{\mu}{2 \cdot 1\ \mathrm{AU}}| \tag{4-73}$$

$$|\varepsilon_2 - \varepsilon_1| = \left|\tfrac{1}{5}\varepsilon_1 - \varepsilon_1\right| = |-\tfrac{4}{5}\varepsilon_1|$$

This change has to be achieved by the thrust and possible gravity assist effects and of course depends on the exact mission. In this example, 80% of the specific energy needs to be created by the thrusting and possible gravity assists to complete the mission. Even if only half of that energy change needs to be accomplished by thrust, this exceeds the magnitude of the previously explored errors and is not negligible.

While missions with less demanding energy changes are possible and conceivable, energy changes of only 10% to 20%, which would be a change within the error magnitude of the non-circular planetary orbits (see Chapter 4.3.2), are not large enough to require low-thrust propulsion in the first place. The reason to use low-thrust propulsion (and combine it with gravity assists) is to enable highly challenging missions. Therefore, the demanding missions have to be selected as benchmark.

4.3.6 Tisserand's Criterion under Consideration of Thrust

Tisserand's Criterion has been derived with the assumption that only gravitational forces act in the three-body system. For the intended application however, also thrust has to be incorporated in the system of equations. The effect of a thrust acceleration on Tisserand's terms is investigated in this section. The results have been introduced in (Maiwald, 2016) and presented in detail in (Maiwald, 2017 a).

Based on Equations (4-45) to (4-47), the summary of all accelerations – including thrust – is:

$$\ddot{x} - 2\,n\,\cdot\dot{y} = \frac{\partial U}{\partial x} + T_x, \tag{4-74}$$

$$\ddot{y} + 2\,n\,\cdot\dot{x} = \frac{\partial U}{\partial y} + T_y, \tag{4-75}$$

$$\ddot{z} = \frac{\partial U}{\partial z} + T_z, \tag{4-76}$$

where T designates the thrust acceleration in the respective coordinate (i.e. x, y, z) identified by its index. In an analogue manner as before, multiplying each of the Eq. (4-71) to (4-73) with the time derivate of the matching coordinate and adding them up, results in an equation that is (compare Eq. (4-48)):

$$\dot{x}\,\ddot{x} + \dot{y}\,\ddot{y} + \dot{z}\,\ddot{z} = \frac{dU}{dt} + \dot{x}\,T_x + \dot{y}T_y + \dot{z}T_z = \frac{dU}{dt} + \vec{V}\cdot\vec{T}, \tag{4-77}$$

where $\vec{T}$ is the spacecraft's mass specific thrust, i.e. thrust acceleration. Integrating it over time results in a term similar to the original Jacobi Integral as given by Eq. (4-50) and the added term involving the thrust acceleration:

$$C_j^* - 2\int_{t_1}^{t_2} \vec{V}\cdot\vec{T}\,dt = 2U - V^2. \tag{4-78}$$

Here C_j^* is the integration constant of the modified equations and t_1 is the point in time before the gravity assist and t_2 after. Furthermore, it is apparent that the right side is identical to the original Jacobi Constant. The thrust acceleration in Eq. (4-75) results in a time variance of the term, depending on the length of the time interval *dt*. Generally, this means that the modified Jacobi Integral can be expressed as:

$$C_j^* = C_j + 2\int_{t_1}^{t_2} \vec{V}\cdot\vec{T}\,dt. \tag{4-79}$$

Therefore, in difference to the original Tisserand's Criterion, the modified term is no longer a state variable, but depends on a time integral, here describing the mass specific work. This is to be expected as for a thrusting vehicle the orbit properties continuously change as well, changing the value for Eq. (4-63) for each point in time.

For a constant thrust acceleration vector parallel to the velocity vector and assuming both as constant in magnitude, the integral can be solved to:

$$C_j^* = C_j + 2\left(\Delta x\cdot T_x + \Delta y\cdot T_y + \Delta z\cdot T_z\right) = C_j + 2\,(\vec{R}_2 - \vec{R}_1)\cdot\vec{T} = C_j + 2\,\overrightarrow{\Delta R}\cdot\vec{T}, \tag{4-80}$$

where the Δ designates a difference in position for two points in time (labeled '1' and '2') observed in this equation, i.e. before and after the gravity assist. However, it is unlikely that the thrust vector will not change its direction over the course of a trajectory, making it adequate only for small time intervals. The term involving the thrust acceleration is an energy term – and therefore matches the integral – but its meaning is not directly derived. It is the energy bestowed on the spacecraft through the thrust. While the gravitational force, expressed by a pseudo potential, is a state variable and independent of the path of the spacecraft, the same is not true for the energy change of the spacecraft caused by thrust.

The situation of the respective position vectors is shown in Figure 4-5, where $\vec{R}_2$ is the vector at the end of the time interval and $\vec{R}_1$ at the beginning. The vector named $\overrightarrow{\Delta R}$ is the difference of the two, as used in Eq. (4-80). Also shown is the path $\Delta R'$, which is an example of how an actual partial trajectory might look like. While the difference between the two position vectors is the direct route, this needs not be true for the actual path the spacecraft takes. For small time intervals (depending

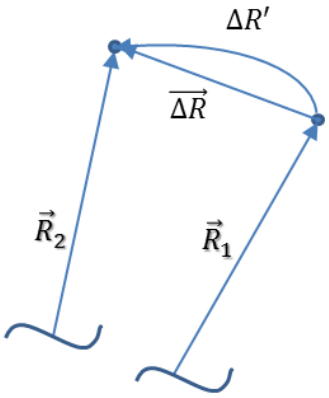

Figure 4-5: The position vector situation for the thrust acceleration addition of the Jacobi Integral.

on the velocity and thus distance the spacecraft covers in this interval) the difference between $\left|\overrightarrow{\Delta R}\right|$ and $\Delta R'$ can be small, but this cannot be generalized. Therefore, it should be clear that the term added to the Jacobi Integral with the simplification as given in Eq. (4-80) is not an accurate amount of work conducted by the thrust, but only an approximation, which becomes more accurate with smaller time frames, when simplified in the described manner.

With these considerations, the formulation for the gravity-assist maneuver can be modified. Analogously to Eq. (4-63), the same equation with regard of thrust can be expressed as:

$$\frac{1}{2\,a_2} + \sqrt{a_2\,(1-{e_2}^2)}\cos(i_2) = \frac{1}{2\,a_1} + \sqrt{a_1\,(1-{e_1}^2)}\cos(i_1) + 2\int_{t_1}^{t_2}\vec{V}\cdot\vec{T}\,dt \tag{4-81}$$

It is noted that for small time intervals the integral also becomes very small. As described in Chapter 2.4.2 the gravity-assist maneuver is often modelled as instantaneous rotation, i.e. for this model the time interval is zero, therefore the thrust effect is negligible as it becomes zero (dt = 0). This is equivalent to the difference of the position vectors $\overrightarrow{\Delta R}$ becoming zero, as due to the instantaneous rotation, the position vectors are identical for the conditions before and after the GA maneuver.

However, it has to be kept in mind that using this approach violates the assumption that the calculation is conducted in large distances of the planetary body (see Eq. (4-62)). Again assuming the instantaneous model the distance to the planetary body is equal for both conditions, before and after the encounter, and thus can be subtracted on both sides:

$$\frac{\mu_2}{R_{2,2}} + \frac{1}{2\,a_2} + \sqrt{a_2\,(1-{e_2}^2)}\cos(i_2) = \frac{\mu_2}{R_{2,1}} + \frac{1}{2\,a_1} + \sqrt{a_1\,(1-{e_1}^2)}\cos(i_1) + 2\int_{t_1}^{t_2}\vec{V}\cdot\vec{T}\,dt \tag{4-82}$$

where $R_{2,2}$ designates the distance between the spacecraft and the planetary body after the encounter and $R_{2,1}$ before, whereas μ_2 refers to the gravitational parameter of the planetary body (scaled with the Sun's, which is assumed to be unity throughout this derivation).

Further simplifications can be obtained in case the velocity is assumed to be near circular, i.e. in a spiral, and that the (constant) thrust acceleration is also acting tangentially on the spacecraft. As both vectors are parallel then, a scalar results from the integral. With θ as the angle covered by the line between origin of the respective coordinate system and the spacecraft and from Eq. (4-81) this yields:

$$\frac{1}{2\,a_2} + \sqrt{a_2\,(1-{e_2}^2)}\cos(i_2) = \frac{1}{2\,a_1} + \sqrt{a_1\,(1-{e_1}^2)}\cos(i_1) + 2\int_{t_1}^{t_2} r\cdot\frac{d\Theta}{dt}\cdot T\,dt$$

$$\frac{1}{2\,a_2} + \sqrt{a_2\,(1-{e_2}^2)}\cos(i_2) = \frac{1}{2\,a_1} + \sqrt{a_1\,(1-{e_1}^2)}\cos(i_1) + 2\int_{\Theta_1}^{\Theta_2} r\cdot T\,d\Theta. \tag{4-83}$$

As the radius r is an unknown (as the trajectory is not known) function of the angle Θ, this formulation cannot be further simplified without knowing a function for r.

Taking a closer look at the correction term in Eq.(4-79) and reformulating it with the help of the definition of the vector product, provides an estimate for the maximum possible value of that term:

$$\vec{V}\cdot\vec{T}\,dt = \left|\vec{V}\right|\cdot\left|\vec{T}\right|\cos\varphi\,dt, \tag{4-84}$$

where φ is the angle between the two vectors. The cosine of this angle can be between -1 and +1, so in general for each point in time the envelope of the correction term would be +/- $\left|\vec{V}\right|\cdot\left|\vec{T}\right|\cdot\Delta t$. For each point in time, represented by a given value for C_j, there would be an upper and lower limit for its modification term, depending on the respective time difference between the given points in time, i.e. thrusting time. This is illustrated in Figure 4-6.

The first aspect, which is apparent, is that the respective thrusting time $\Delta t'$ has to be assumed to allow defining the envelope. The thrusting time is already a part of the trajectory's control, defining when the spacecraft is thrusting and when it is coasting. This means, setting the thrusting time is a restriction of the search space to regions fitting this thrusting time. It is further related to the phasing of the mission's defining bodies (e.g. departure and arrival body), the actual thrust magnitude and direction, possible perturbation forces and propellant mass. An a priori selection of thrusting time already restricts the possible trajectories.

While assumptions about the thrust acceleration vector can be made, e.g. by defining thrust always in a tangential direction and with a certain maximum thrust; this is not equally true for the actual velocity. The velocity associated to a certain point in time of the mission's history is already a direct

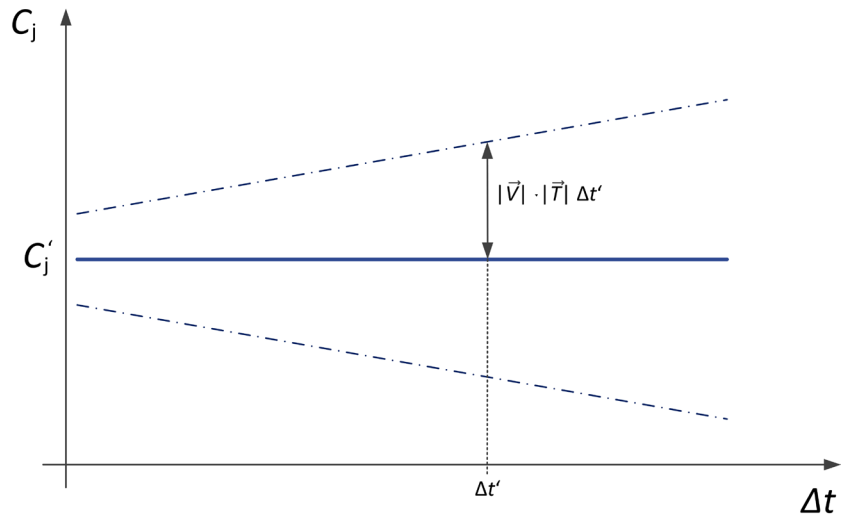

Figure 4-6: The envelope of maximum and minimum values for the correction term and a given thrusting time Δt. $\left|\vec{V}\right|$ is the magnitude of the velocity vector and $\left|\vec{T}\right|$ of the thrust acceleration vector.

representation of the trajectory flown up to that point. Any assumption about the velocity vector is already an assumption about the trajectory.

Even if the thrust acceleration is assumed to remain constant over time, the velocity cannot be constant. Otherwise, there would be no trajectory change, which is impossible when thrusting occurs. The velocity itself depends on the external forces, here expressed by the sum of all gravitational accelerations g (dominated by the central body) and the thrust acceleration and can be expressed as:

$$\vec{V}(t) = \int_{t_1}^{t} \left(\vec{g}(\tau) + \vec{T}(\tau)\right) d\tau, \tag{4-85}$$

as integral from a certain starting point t_1 until an unspecified point in time t. Combined this leads to the following formulation for the correction term:

$$\int_{t_1}^{t_2} \vec{V} \cdot \vec{T}\, dt = \int_{t_1}^{t_2} \left[\int_{t_1}^{t} \left(\vec{g}(\tau) + \vec{T}(\tau)\right) d\tau\right] \cdot \vec{T}\, dt, \tag{4-86}$$

Even when making assumptions about the thrust acceleration, any assumption about the velocity also results in an assumption about the gravitational acceleration behind it. This in turn depends on the relative positions of the system's bodies towards the spacecraft. Making assumptions about the velocity, i.e. the gravitational acceleration and thrust acceleration in sum, means the trajectory is already defined as the position and gravitational acceleration are directly connected to each other.

Therefore, assumptions about the velocity (history), e.g. to determine an envelope as pictured in Figure 4-6, actually already define the trajectory and therefore a trajectory optimization can no longer effectively occur. The definition of the velocity (history) restricts the solution space significantly. While previous experience can be used for making assumptions about the velocity, the goal of this thesis is to find a method that is able to explore the whole search space and not just parts, which are based on previous experience.

An estimation of the thrust effect can be obtained with equations for e.g. the osculating rate of the semi-major axis a, based upon the thrust acceleration T. It depends on tangential thrust acceleration T_t and the current velocity v. This can be written as: (Alles, 2005, p. 3-15):

$$\frac{da}{dt} = 2\frac{a^2}{\mu} \cdot v \cdot T_t. \tag{4-87}$$

A similar equation can be derived for the time derivate of the eccentricity. Combined with an assumed thrust duration, this can provide an analytical solution for the change of a. However, assumptions about the the thrust acceleration magnitude and history as well as v all are required. Therefore, the quality of this guess depends on the experience of the mission analyst. These assumptions are similar to those required to solve the correction term (Eq. (4-79) resp. (4-80)), and again are only estimates and provide no accurate solution. Therefore, such assumptions and planning can serve only as indications at best.

All of the simplifications mentioned before are constraints on the actual trajectory; too limiting assumptions could lead to very limited search spaces in the course of the optimization.

4.3.7 Recommendation on Usage of Tisserand's Criterion

The previous sections have shown that the error caused by assumptions for the derivation of Tisserand's Criterion in comparison to the situation of the actual solar system is most of the times negligible, however exceptions exist. Considering planetary eccentricity can yield a deviation of

Tisserand's Parameter in the order of 25%. Depending on the actual example used, even larger differences in Tisserand's Parameter could exist.

It is apparent that the involved – and intended – energy change caused by the propulsion is in violation of the unmodified Tisserand's Criterion and for typical missions is also not negligible. Therefore, a correction term has been derived, which can be included to consider the energy effect of the continuous thrusting. This term, however, is no state quantity.

Summarizing, Tisserand's Criterion can be applied to investigate gravity-assist sequences for low-thrust applications only with the inclusion of the correction term. This correction term also prevents an a priori evaluation of a trajectory as the trajectory information, i.e. position and velocity history, have to be known to solve the correction term.

5 Derivation of the Optimization Method

In the previous chapter the theoretical background for optimization of gravity-assist sequences was analyzed and the suitability of Tisserand's Criterion was discussed, after the derivation of a correction term. This provided the answers to the first two questions as given in Chapter 3. This chapter will now attempt to provide answers to the remaining questions:

3. *What would be the control variables of the optimization?*
4. *Which algorithms are best suited for the optimization and why?*
5. *What is a useful formulation of an objective function that allows the usage and determination of a suitable gravity-assist sequence?*
6. *How can gravity-assist maneuvers be "properly rewarded" without losing the perspective concerning non-gravity-assist trajectories (i.e. no non-gravity-assist trajectory has a better result in the optimization objective)?*

The validity of these answers, embedded into an optimization method, presented in Chapter 6, is checked and discussed in Chapter 7.

In comparison to the optimization methods described in Chapter 2, the main objective of the to be designed method is to include the gravity-assist partner(s) in the optimization process and thus allow a thorough search of the search space without a priori knowledge based on experience and assumptions. The optimization of gravity assists has to be forced into the mission model and therefore ensure that the gravity assist is actually optimized in difference to the work of Carnelli et al. (Carnelli et al., 2009), where the gravity assist had to be "found" by the optimizer. In this work, the mission is instead prescribed to have a user given number of gravity assists. However, it is not set which planets are the partners for the respective maneuvers so that the actual sequence is open for optimization. This way, only gravity-assist missions are evaluated by the optimizer, leading to a more focused search and optimization, and thus increasing success probability. The mission also needs to be benchmarked with a non-gravity-assist solution to ensure actual improvement of mission performance with the help of gravity assist(s).

Tisserand's Criterion is used for the determination of a gravity-assist maneuver's effect by basically moving along a Tisserand graph and thus changing the heliocentric orbit properties. The gravity-assist maneuver is modelled as instantaneous rotation of the velocity vector (Debban et al., 2002) as the time spent during the gravity assist is small compared to a total mission flight time of possibly several years.

5.1 Global Optimization Algorithms and Low-Thrust Application

The main task before the global optimization is to find a formulation of the problem that allows its optimization, i.e. an objective function, also containing the constraints (Storn and Price, 1997). The presence of several optima (i.e. several different GA maneuvers are possible) necessitates a global search (Venter, 2010). Various algorithm classes exist to optimize such problems. They are presented and reviewed in this chapter. It is further investigated which algorithm or algorithms are best suited for the presented problem based on the relevant variables describing and controlling the problem and a preliminary selection for testing is made at the end of this chapter.

The following sections describe known global optimization algorithms and their current usage. While these algorithms allow searching for global optima, a strict convergence cannot be proven for a general case (Venter, 2010).

5.1.1 Hill Climbing

The hill climbing algorithm is a simple optimization algorithm, where a new solution is always a modified version of a previous solution and accepted only if it is of better quality (depending on the optimization goal). Neighbouring candidates are evaluated and the next candidate is always the best performing neighbor. Therefore, hill climbing has a high risk of converging only to a local optimum and not a global one. (Russel and Norvig, 2003)

More sophisticated versions of the algorithm can incorporate multi-objective optimization as well as means to reduce the risk of convergence to a local optimum. These are for instance (Russel and Norvig, 2003):

- Previously visited solutions or closely related are excluded from further usage,
- not using only the best solution for progress, but also investigating other solutions (this is done in Simulated Annealing, see Section 5.1.3),
- random restart of the search to prevent visiting only one area of the search space.

5.1.2 Particle Swarm Optimization

Particle Swarm Optimization mimics the behavior of foraging animals and belongs to the class of evolutionary algorithms. Information regarding their movement and the swarm's movement is used to modify each swarm member's own behavior. Based on a previous solution $\vec{x}_i$ of the generation g and its velocity $\vec{v}_i$, a new solution is generated for the generation g+1 (usually for this the time interval Δt is equal to 1) (Venter, 2010):

$$\vec{x}_i^{\,g+1} = \vec{x}_i^{\,g} + \vec{v}_i^{\,g} \cdot \Delta t \tag{5-1}$$

The velocity vector is a combination of weighted (weights are w, c_1, c_2, and are determined in different ways, while r_1 and r_2 are random numbers between 0 and 1 created during each iteration of solutions, see Venter, 2010) difference vectors between the individual's best solution $\vec{p}_i$ and the previous solution as well as the swarm's best solution $\vec{p}^{\,s}$ and the individual's previous solution as well as its velocity (Venter, 2010):

$$\vec{v}_i^{\,g+1} = w\,\vec{v}_i^{\,g} + c_1 r_1 \frac{(\vec{p}_i - \vec{x}_i^{\,g})}{\Delta t} + c_2 r_2 \frac{(\vec{p}^{\,s} - \vec{x}_i^{\,g})}{\Delta t} \tag{5-2}$$

Adapting these five weighting factors to the optimization problem is challenging and requires trials and experience. Furthermore, constraint handling is also challenging and requires experience. (Venter, 2010)

5.1.3 Simulated Annealing

This optimization technique mimics the crystallization processes in metal during cooling. When cooled to the absolute minimum and with sufficiently slow cooling in theory the crystals would assume a minimum energy condition, i.e. a stable configuration. The slow cooling ensures that the crystals do not converge to meta-stable configurations. (Weise, 2008)

The premise is that a new solution candidate is created based on a previously known solution candidate and, picking up the idea of a lesser energy state, if the fitness of the new solution is better (better depends whether a maximum or minimum is sought for) than the previous one's, it is taken as new solution candidate. In case the fitness is worse compared to the previous solution candidate, the probability that it is taken as new candidate is a function depending on the current temperature of the annealing – the larger the temperature (i.e. the earlier in the process) the larger the probability. (Weise, 2008) This means, that with further progressing iterations of the search, the ability to accept less optimal solution candidates is reduced, which is tantamount to preventing convergence to the local optimum (Storn and Price, 1997).

Fitness is usually expressed as *energy* within Simulated Annealing optimization, which is to be minimized (as in annealing the minimum energy condition is approached). For each iteration of the optimization, the respective energy value is determined based on the fitness function. For a pairing of a current iteration's solution candidate and previous iteration's solution, the energy difference ΔE is determined (i represents the respective iteration) (Weise, 2008):

$$\Delta E = E(i+1) - E(i) \tag{5-3}$$

If the new solution candidate (i+1) has a better fitness (here assumed to be a represented by a smaller value for E), then the probability P that it is adopted as new solution is 1. If the previous solution (i) has a better fitness, then the probability that the solution candidate becomes the new solution is determined by (Weise, 2008):

$$P(\Delta E) = e^{-\frac{\Delta E}{k_B \cdot T}} \tag{5-4}$$

where k_B is the Bolzmann constant and T the current iteration's temperature. Due to the non-constant temperature, in the earlier parts of the optimization, Simulated Annealing allows moving out of the vicinity of a local optimum, while in later parts a once found area is optimized until the (global) optimum is reached, because based on Eq. (5-4), the probability to accept less performing solution candidates is reduced. (Weise, 2008)

The temperature drops with progressing iterations, simulating the cooling. The exact cooling schedule has to be selected before the optimization. Various schemes exist, e.g. (Nourani and Andresen, 1998):

- Linear temperature drop: $T(i) = T_0 - \eta \cdot i$ (5-5)
- Exponential temperature drop: $T(i) = T_0 \cdot \alpha^i$ (5-6)

The variable i is the number of iteration, η is a number scaling the linear drop and α a number < 1. Typically, the temperature remains constant for m iterations, whereas m is depending on the user (Weise, 2008) and would then replace i in Eq. (5-5) resp. (5-6). However, the method of lowering the temperature has a huge impact on the performance of the algorithm and its success rate (Dueck and Scheuer, 1990). It therefore needs experience and trials for proper usage. Besides selection of a scheme for reducing the temperature also a starting Temperature T_0 has to be selected to apply Simulated Annealing.

In addition, the method for creating new solution candidates for each iteration *i* has to be defined, based on a neighbourhood percentage, i.e. based on the difference which is accepted for each control variable per iteration in percent.

Methods exist to increase the speed of Simulated Annealing at the expense of convergence to the global optimum and to include multi-objective optimization (Weise, 2008). In a more general sense resp. definition, Simulated Annealing is also an evolutionary algorithm (Venter, 2010), as it is e.g. not gradient based, uses natural principles for finding a solution and new solution candidates are created out of previous ones.

5.1.4 Threshold Accepting

Threshold Accepting is a modified, simplified version of Simulated Annealing. It reduces the problems of finding the correct temperature decrease and replaces it with a simple threshold. This threshold is set at the beginning of the optimization. Just like for Simulated Annealing a new solution candidate is generated and then compared with the previous one. In difference to Simulated Annealing where a less performing solution candidate has only a certain probability of being accepted as new candidate, for Threshold Accepting, the new candidate is always accepted if the difference between its fitness and that of the previous solution candidate is exceeding a given threshold. According to test results of Dueck and Scheuer (Dueck and Scheuer, 1990), Threshold Accepting is on average superior to Simulated Annealing in the quality of its solutions. One major advantage is that the quality of the results is hardly sensitive regarding the method of reducing the threshold over time. (Dueck and Scheuer, 1990)

5.1.5 Evolutionary and Genetic Algorithms

The field of evolutionary algorithms is manifold and often certain terms are interchanged and ambiguous. Major categories of evolutionary algorithms, which are heuristic optimization algorithms (Weise, 2008), are:

- Genetic Algorithms (Ashlock, 2006): Uses data structure of fixed size, the solution is developed using mutation and recombination mechanisms of existing solution candidates, analogue to evolution of a species,
- Genetic Programming (Weise, 2008): Uses variably sized data structures and is mostly applied for programming optimization,
- Evolutionary Programming (Weise, 2008): Solution candidates represent whole species and not individuals; therefore, no crossover occurs, only mutation creates new solutions.

Genetic Algorithms allow optimization of problems where little or no information is available about the optimal solution, resp. only the desired result is known. Their optimization uses objective functions and probabilistic rules. The fact that they use populations of solution candidates allows recombination of good solution elements to form even better solution candidates (Pirlot, 1996). This mechanism is based on the schema theorem and is dependent on the probability of mutation and the nature of crossover between solution candidates (Mahfoud, 1995).

Solution candidates are mapped in their parameters in so-called chromosomes (Ashlock, 2006), while the different positions in this chromosome are called loci, the actual values are called allele (Mühlenbein, 1992) (e.g. in Figure 5-1 if $w_{1,3}$ can vary between 1 and 2, then 1 and 2 are the two possible alleles). All candidates of one generation form the population, while the population size grants the diversity of the candidates and therefore contributes to the global nature of the algorithm's search (Pirlot, 1996). Different means, i.e. mutation and crossover, modify the population. Each iteration of the population is called generation (Pirlot, 1996) and a solution candidate's fitness value is called phenotype (Hartmann, et al., 1998).

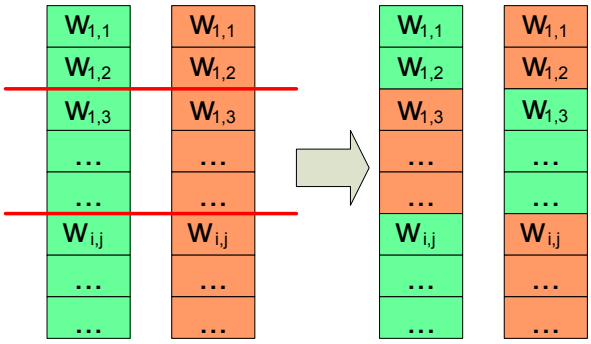

Figure 5-1: Two-point cross-over for two solutions with *n* dimensions and in the first generation.

Solution candidates are selected via various schemes for further processing. Roulette wheel selection uses the ratio of the individual's fitness value to the overall sum of fitness values, to determine the probability of selection. This ensures diversity at the cost of performance, as the best solution is not guaranteed (Ashlock, 2006) to be selected. (Weise, 2008)

Rank selection works similar to roulette wheel selection, but addresses each candidate with a rank. With a probability based on the rank, each candidate is selected. (Ashlock, 2006)

Tournament selection means that either in groups or in pairs every solution is compared to the others according to its fitness. The winners' children replace the less fit candidates. (Ashlock, 2006) This method is named elitism as the best candidate is always part of the next generation (Weise, 2008). Strong elitism increases the risk of optimizing to a local minimum (Weise, 2008).

Replacement of solutions usually follows similar schemes, after new candidates have been created following the selection of the parents (Ashlock, 2006).

After the selection, recombination occurs, where previous solution candidates are combined via crossover into new solution candidates. With processes similar to the selection after the crossover, these solutions are placed into the next generation for evaluation. Crossover is conducted in different schemes, e.g. point-crossover where one parent's chromosome is cut at one (or several points) and combined with the chromosome part of the other parent. (Ashlock, 2006) This is shown in Figure 5-1.

Mutation ensures also data diversity and introduces new information into a solution (otherwise, all solutions would be recombinations of the initial generation). Depending on a given probability, one or even several loci are randomly modified. (Ashlock, 2006)

Two methods exist to increase solution diversity: *niching* and *crowding*, intended to oppose premature convergence to a local optimum.

Niching

Niching applies the idea that ecological niches cannot be occupied by an infinite number of species (El-Betagy and Keane, 1999). Generally, the goal is to maintain good (i.e. high fitness) solutions to explore all optimum regions (Mahfoud, 1995). This is achieved by penalizing solutions' fitness values if they are close together, i.e. add no new information to the exploration of the solution space (El-Betagy and Keane, 1999). Niching comes at increased computational cost and time.

Crowding

Crowding also targets at improved solution diversity but at the step of solution replacement (Mahfoud, 1995). During crowding, solution candidates of a group (or whole population) are

compared to the newly produced children in terms of similarity and the most similar is replaced. This is called explicit crowding. Implicit crowding means only the parents are considered for being replaced. (Mengshoel and Goldberg, 2008)

In probabilistic crowding, the replacement occurs depending on a probability based on the fitness value of the compared solutions. In deterministic crowding it is enough if the fitness value is better. The latter is stronger in convergence. Mixing both methods in a portfolio and using each method depending on a certain probability, grants convergence and diversity. (Mengshoel and Goldberg, 2008)

5.1.6 Differential Evolution

Differential Evolution is an evolutionary algorithm that mixes mutation with recombination, like a mixture between Particle Swarm Optimization and Genetic Algorithms. Differential Evolution requires at least four members of a population and randomly recombines them with weighting factors and difference vectors between the individuals (Vasile et al., 2010). One major idea of Differential Evolution is that it uses information from the whole population of solution candidates to modify the current candidates and transform them to new ones (Storn and Price, 1996). Nevertheless, this is done less restrictive than in the case of simple recombination of two solutions into new solutions like it is part of genetic algorithms.

Generally, three categories of vectors exist for this method (Storn and Price, 1996):

- Target Vector $\vec{x}$: the given solution candidate vector, which serves for comparison, it has d dimensions, describing the problem
- Mutation Vector $\vec{v}$: vector modification created for each target vector, based on a difference of other current population members:

$$\vec{v}_i^{\,g+1} = \vec{x}_{r1}^{\,g} + F \cdot \left(\vec{x}_{r2}^{\,g} - \vec{x}_{r3}^{\,g}\right), \tag{5-7}$$

  here g represents the number of the generation, i the index of the vector (from 1 to n, with n being the number of solution candidates), $r1$ to $r3$ random indexes and F is a real, constant factor ϵ [0,2]. Variations of this definition include a larger number of vectors used for creating the difference vector.
- Trial Vector $\vec{u}$: The trial vector is basically the new solution candidate created out of the target vector and the mutation vector. For each of the d components of the vector a random number between 0 and 1 is determined and compared to a so-called crossover constant, also set by the user to a value between 0 and 1. If the random number of the current vector component is equal or smaller than the crossover constant, than the trial vector's component becomes the one of the mutation vector. Otherwise, it remains the one of the original target vector. However, in any case for a randomly selected component the mutation vector's value is used.

Once the trial vectors are determined, they are compared to the respective target vectors. If the fitness is better they replace the target vector, if not the target vector remains in the population of the next generation. (Storn and Price, 1996)

Vasile et al. (Vasile et al, 2010) use a modified version of Differential Evolution, where the mutation vector composition and creation differs in a way that it essentially combines it with the random part of the trial vector creation. The trial vector is then only the sum of the original target vector and the mutation vector:

$$\vec{v}_i^{\,g+1} = \boldsymbol{e}\left[\left(\vec{x}_{r1}^{\,g} - \vec{x}_i^{\,g}\right) + F \cdot \left(\vec{x}_{r2}^{\,g} - \vec{x}_{r3}^{\,g}\right)\right], \tag{5-8}$$

The meanings are the same as above, only that the function ***e(j)*** becomes 1 if the random number of *j*-component is larger or equal to the crossover constant and 0 if it is smaller, thus making the recombination already a part of the mutation vector. For their part, *r*1 can either be selected randomly, which they call exploration scheme, or it is the index of the known best solution (note the similarity to particle swarm optimization), which they label as convergence scheme. Replacement is also elitist as for Storn and Price. They also report that typical values for *F* are 0.75 and for the crossover constant it is 0.8. (Vasile et al., 2010)

5.1.7 Deterministic Methods

Besides the various types of evolutionary algorithms deterministic methods are one other branch of global optimizers. However, they are usually tailored to a specific problem (Venter, 2010). Methods like the DIRECT algorithm apply Lipschitzian optimization (Venter, 2010) (gradient based) for finding areas in the design space, which are worthwhile of further investigation and thus are subjected to local search algorithms. In test cases, however methods like Swarm Optimization and Differential Evolution have been beneficial over the DIRECT algorithm (Wyatt et al., 2004).

5.1.8 Evaluation and Selection

A general evaluation of (global) optimization algorithms is difficult, because no single algorithm is suited for all possible applications (Venter, 2010) and a formal, resp. analytical method determining convergence and optimization quality does not exist. While various methods of benchmarking exist, their results are specific to the given problems or optimized functions and cannot be generalized. Even the given mission that is to be optimized can change the topology of the search space.

In the field of trajectory optimization genetic algorithms have been used before, e.g. by Vasile et al. (Vasile et al., 2010) and benchmarked. Debban et al. (Debban et al., 2002) have used a similar broad search algorithm and a cost function for optimizing low-thrust trajectories and predefined gravity-assist sequences.

One major advantage of Differential Evolution is the little computation costs in comparison to e.g. Simulated Annealing and Genetic Algorithms. Furthermore, it has few control variables, which need to be set (Storn and Price, 1996). In trajectory optimization Differential Evolution outperformed most other algorithms, however low-thrust application has not been tested, only a multi-maneuver scenario (Vasile et al., 2010).

The performance of a search algorithm strongly depends on the topology of the search space, which is usually unknown for low-thrust trajectory optimization. While trajectory optimization has been conducted many times, providing a certain impression of possible search space topologies, the combination of sequence finding and low-thrust has not yet been tested and therefore currently has not been reviewed regarding its search space topology. Consequently, a selection of the most promising search algorithm based on their respective performance regarding a certain search space topology is not possible. Thus, three different kinds of algorithms are selected and are investigated regarding their performance on the given problem:

1) Differential Evolution: Broad search strategy, behaves favorable in previous tests due to low computation costs and is easy to apply
2) Simulated Annealing: Effective search strategy based on previous solutions, capable of searching the whole search space, but slow in comparison to Differential Evolution
3) Random Search: simple method use as benchmark to evaluate the performance of more sophisticated algorithms

In the course of the work, each of these algorithms is tested on example calculations to determine their suitability, see Chapter 7.3 for Random Search, Chapter 7.4 for Differential Evolution and Chapter 7.5 for Simulated Annealing.

5.2 Trajectory Model

As described in Section 2.4.2, methods exist to represent a possible trajectory with an approximation of its shape. Within the frame of this thesis, various shapes have been reviewed for their suitability. The polynomial shape has the following advantages over other methods (e.g. logarithmical shapes):

1. flexible regarding optimization goals,
2. broad solution variety (i.e. good exploitation of parameter variation)
3. accuracy of the results (near optimality)
4. applicability on three dimensional cases (if extended in its formulation)
5. applicability for multi-rendezvous missions

The effort in calculations is comparably large with regards to other shape-based approaches. Especially when comparing the quality of the solutions regarding optimality, a slightly increased computation effort when compared with other shape-based methods is regarded as acceptable by the author. Consequently, the shape-based method as described by Wall and Conway (Wall and Conway, 2009) is applied for this work.

5.2.1 Model Equations

The model is described by the following equation (Wall and Conway, 2009):

$$r = \frac{1}{a + b \cdot \theta + c \cdot \theta^2 + d \cdot \theta^3 + e \cdot \theta^4 + f \cdot \theta^5 + g \cdot \theta^6}, \tag{5-9}$$

where a to g are coefficients defining the trajectory's shape and are used so that it and derivates, such as the flight angle, satisfy the mission constraints, i.e. arrival and departure conditions, defined by the respective angles $\theta = 0$ and $\theta = \theta_f$. The angle is derived via the relative position $\tilde{\theta}$ to each other from the arrival and departure body and the number of revolutions N_{rev} set by the optimizer as control variable:

$$\theta_f = \tilde{\theta} + 2\,\pi\,N_{rev}. \tag{5-10}$$

Besides such constraints, this formulation has to follow the equations of motion, which Wall and Conway use in polar coordinates (Wall and Conway, 2009):

$$\ddot{r} - r\dot{\theta}^2 + \frac{\mu}{r^2} = T_a \sin\alpha \tag{5-11}$$

$$\frac{1}{r}\frac{d}{dt}\left(r^2\dot{\theta}\right) = r\ddot{\theta} + 2r\,\dot{\theta}^2 \tan\gamma = T_a \cos\alpha \tag{5-12}$$

Here T_a is the variable thrust acceleration, γ denotes the trajectory's flight path angle and α is the angle describing the thrust direction in the trajectory's plane (Wall and Conway, 2009). It should be noted that this is a polar form of Eq. (2-34), although without explicit mentioning of a throttle factor (implicit it is included, due to the variable thrust magnitude).

The definition of the flight path angle γ is (Wall and Conway, 2009):

$$\tan\gamma = \frac{\dot{r}}{r\dot{\theta}} = -r \cdot (b + 2c\theta + 3d\theta^2 + 4e\theta^3 + 5f\theta^4 + 6g\theta^5). \tag{5-13}$$

Wall and Conway assume *tangential* thrust for their model, which simplifies the relation between α and γ to (Wall and Conway, 2009):

$$\alpha = \gamma + n\pi \tag{5-14}$$

Here, n is either 0 or 1, i.e. thrusting occurs in flight direction or against it. The justification for *tangential* thrust is the maximum rate of changing the kinetic energy (Wall and Conway, 2009). The tangential thrust assumption allows a formulation of a thrust term, which is unique (see below), otherwise the thrust direction α would be another (real valued) dimension. Reducing the problem to a tangential thrust situation therefore reduces the number of dimensions of the optimization attempt.

Via transformation of the respective equations Wall and Conway derive formulations for the coefficients a to c and e to g (the indeces of 1 and 2 represent starting and final conditions respectively, the index f denotes the total angle swept during the transfer) (Wall and Conway, 2009):

$$a = \frac{1}{r_1}, \qquad b = -\frac{\tan\gamma_1}{r_1}, \qquad c = \frac{1}{2\,r_1}\left(\frac{\mu}{r_1^3\dot{\theta}_1^2} - 1\right) \tag{5-15}$$

$$\begin{bmatrix} e \\ f \\ g \end{bmatrix} = \frac{1}{2\theta_f^6}\begin{bmatrix} 30\theta_f^2 & -10\theta_f^3 & \theta_f^4 \\ -48\theta_f & 18\theta_f^2 & -2\theta_f^3 \\ 20 & -8\theta_f & \theta_f^2 \end{bmatrix}\begin{bmatrix} \frac{1}{r_2} - (a + b\theta_f + c\theta_f^2 + d\theta_f^3) \\ -\frac{\tan\gamma_2}{r_2} - (b + 2c\theta_f + 3d\theta_f^2) \\ \frac{\mu}{r_2^4\dot{\theta}_f^2} - (\frac{1}{r_2} + 2c + 6d\theta_f) \end{bmatrix} \tag{5-16}$$

The time derivate of Eq. (5-9) is (Wall and Conway, 2009):

$$\dot{r} = -r^2 \cdot (b + 2c\theta + 3d\theta^2 + 4e\theta^3 + 5f\theta^4 + 6g\theta^5) \cdot \dot{\theta}. \tag{5-17}$$

In combination with the equations of motions Eq. (5-11) and (5-12), as well as Eq. (5-13), a term for the angular rate can be formulated (Wall and Conway, 2009):

$$\dot{\theta}^2 = \left(\frac{\mu}{r^4}\frac{1}{\left[\frac{1}{r} + 2c + 6d\theta + 12e\theta^2 + 20f\theta^3 + 30g\theta^4\right]}\right) \tag{5-18}$$

From this the integral over time is derived, which has to be identical to the flight time of the trajectory described with the respective coefficients (Wall and Conway, 2009):

$$\int_0^{t_f} dt = t_f = \int_0^{\theta_f} \sqrt{\frac{r^4}{\mu}\left(\frac{1}{r} + 2c + 6d\theta + 12e\theta^2 + 20f\theta^3 + 30g\theta^4\right)}\,d\theta \tag{5-19}$$

This equation is used to finalize the coefficient d. For determining the final position, resp. angle θ_f initially a flight time has to be assumed. The coefficient d is then varied iteratively until this initial flight time and the flight time determined via Eq. (5-19) are identical. (Wall and Conway, 2009)

With all coefficients set, the respective trajectory is completely described in its positon data, as function of θ.

An example for a trajectory based on this model is given in Figure 5-2 with the spiral form of the trajectory clearly visible. The trajectory depicts a path from Earth to Jupiter with a time of flight of 2000 days. At arrival the spacecraft fulfills rendezvous conditions (see Chapter 2.4.1, page 25) and for the departure the hyperbolic excess velocity is also 0. The right side of Figure 5-2 shows the respective thrust history, which is derived in the following way.

Eq. (5-18) can be further derivated and then yields an expression for $\ddot{\theta}$, which is omitted here due to its length. Inserted into Eq. (5-12) it yields an expression for the thrust acceleration T_a (Wall and Conway, 2009):

$$T_a = -\frac{\mu}{2\,r^3\,\cos\gamma} \cdot \frac{6d + 24e\theta + 60f\theta^2 + 120g\theta^3 - \frac{\tan\gamma}{r}}{\left(\frac{1}{r} + 2c + 6d\theta + 12e\theta^2 + 20f\theta^3 + 30g\theta^4\right)^2} \tag{5-20}$$

Based on the definition of the thrust (Wall and Conway, 2009):

$$T_a = \frac{-\dot{m} \cdot c_e}{m}, \tag{5-21}$$

where c_e is the exhaust velocity and the rocket equation:

$$\Delta v = c_e \cdot \ln\left(\frac{m_0}{m_f}\right), \tag{5-22}$$

where m_0 is the initial mass and m_f the final mass, this yields in the end a formulation for the Δv required by the respective trajectory (Wall and Conway, 2009):

$$\Delta v = \int_0^{\theta_f} \frac{T_a}{\dot{\theta}}\, d\theta. \tag{5-23}$$

5.2.2 Model Application

The model as described before is implemented with the following stages:

1. Based on the mission constraints (e.g. launch date and flight time), the planets' ephimerides provide the initial and arrival conditions of the spacecraft, i.e. velocity and position r (the first of which modified by a hyperbolic excess velocity where applicable), and the angle difference $\tilde{\theta}$. Wall and Conway use circularized planet data, whereas in this thesis actual ephimerides are used (see Chapter 7.1).
2. From this and with Eq. (5-10) (once the optimizer has set the control variable N_{rev}) θ_f is determined.
3. Again with the help of emphemerides, the initial and arrival $\dot{\theta}$ are determined, as based on the total velocity and radial velocity (provided in the ephimerides), the angular velocity can be calculated via $\dot{\theta} = \sqrt{\frac{v^2 - \dot{r}^2}{r}}$.
4. Now the values for the flight path angles can be determined via the left side of Eq. (5-13), i.e. $\tan\gamma = \frac{\dot{r}}{r\dot{\theta}}$.
5. This sets all non-coefficient values for Eq. (5-15) and (5-16), i.e. they can be used to determine the coefficients *a* to *c* and *e* to *g* for the respective trajectory polynomial.

6. Based on the flight time from stage 1, coefficient *d* is iteratively determined by varying it until Eq. (5-19) equals that initially set flight time (in the frame of this work a secant method is used). At this point all coefficients are known.
7. A thrust history is established using Eq. (5-20) and a predetermined number of steps. Wall and Conway suggest 100 steps (Wall and Conway, 2009), which are thus applied in this thesis.
8. With the thrust history and Eq. (5-18) the integral of (5-23) can be solved, here with a trapezoid method (Wall and Conway, 2009).

Shape-based methods are useful for modeling low-thrust trajectories as they are fast methods and allow an immediate calculation of the thrust history over time (Wall and Conway, 2009). They represent a simple form of the before mentioned "shooting" methods of trajectory propagation (Chapter 2.4.1, Optimization Methods) and are an assumption of the shape of the respective trajectory. Generally, they are the equivalent of a solver for Lambert's problem with a low-thrust perspective (Wall and Conway, 2009).

As the shape of a trajectory is determined by a small number of parameters describing a polynomial or comparable function, instead of propagating a trajectory for each time step, the optimization is faster at the expense of accuracy and degrees of freedom. In the here described case the thrust direction is set as tangential.

Wall and Conway determine the difference at 9 h for a propagation method vs. 12.5 minutes for a shape-based calculation – with an accuracy loss of few percent. For their expression of the shape-based method three control variables are applied: Launch Date, Flight Time and Number of Revolutions around the barycenter. (Wall and Conway, 2009)

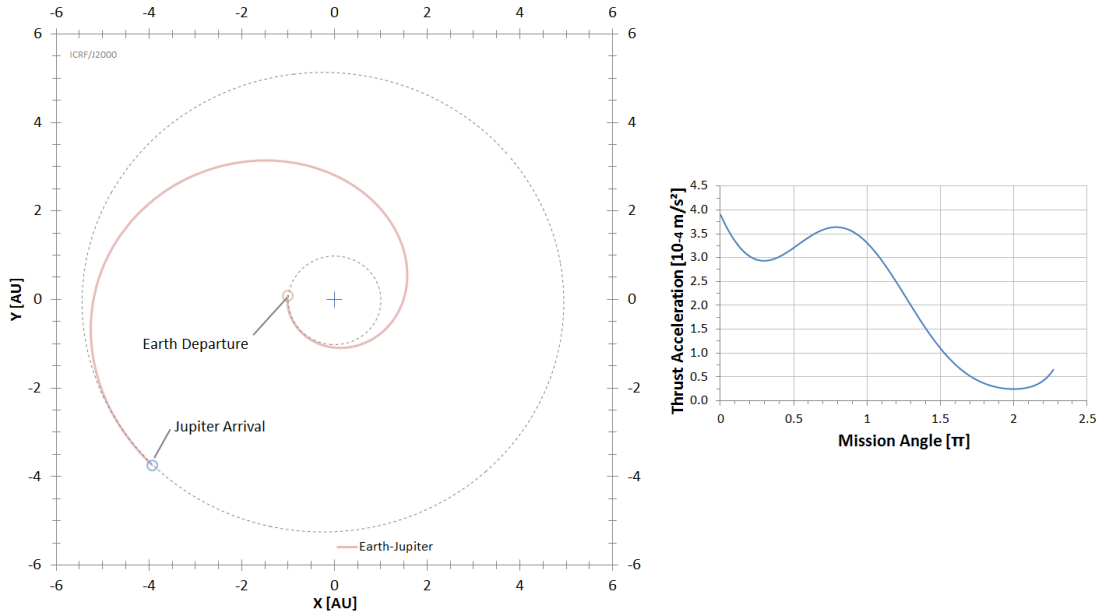

Figure 5-2: An example rendezvous trajectory without gravity assist calculated with the shape-based trajectory model (tangenital thrust) from Earth to Jupiter (v_∞ = 0 for departure and arrival). Coordinates are given in AU, coordinate system is the International Celestial Reference Frame (ICRF) J2000. On the right side is the corresponding thrust history over mission angle Θ in π, based on Eq. (5-20).

5.3 Phasing

As explained in Chapter 2.3.3, gravity-assist sequences found via Tisserand's Criterion, do not include considerations of phasing, i.e. it is open whether a spacecraft trajectory fitting that sequence exists. To prevent investigating or evaluating gravity-assist sequences unsuitable to the involved planet's phasing, the phasing has to be incorporated directly into the sequence's optimization process. This rules out methods (B) or (C) in Figure 3-2. For method (A) phasing is intrinsic, which consequently is selected.

The shape-based trajectory model allows a quick evaluation of the phasing. It can be immediately reviewed whether a potential gravity-assist partner can be reached, without a long propagation of a trajectory, which in the end might fail to reach the desired target. Using the shape-based model the gravity assist is forced to be evaluated and the target body is a variable describing the trajectory segment's end condition. At the same time, the model allows investigating in parallel whether or not trajectories without gravity assist are obtainable, which are better performing than the GA-sequence and thus evaluate the usefulness of the sequence.

5.4 Optimization Variables for Gravity Assist Low-Thrust Trajectories

As mentioned in Chapter 5.2, the selected trajectory model requires the usage of three control variables and a mission consists of several trajectory segments. These are defined by an initial body and a final body. These bodies also serve as the connection for the segments, as the initial body of one leg is the final body of the previous one, creating a match point. This is illustrated in Figure 5-3. The control variables for the shape-based model exist for each trajectory segment and are used to create the constraints against which the trajectory is shaped, e.g. a fly-by at a certain body at the end of one trajectory segment. The initial body of the first leg is also the overall departure body of the mission and the final body of the last leg is the arrival body of the mission (although arrival can also mean a flyby).

Observing the segment's launch date LD_{leg} and the flight time ToF_{leg}, the position and velocity vectors are created using ephemerides of the respective planet. To model the gravity assist the velocity of the spacecraft is modified by a fly-by velocity, resp. hyperbolic excess velocity, v_∞. This adds two more variables (for the planar case, three in case of three-dimensional application). Furthermore, the turning angle δ, see Figure 2-4, is also a control variable defining the outcome of the mission and a direct measurement for the effect of the gravity assist.

The major goal of the optimization is to include the gravity-assist partners as variables in the optimization process as well. This means for each gravity assist a total of four control variables is introduced into the problem besides those defining the trajectory, see Table 5-1.

The combination of all segments defines the mission, which is defined by mission variables, e.g. the mission flight time. These global variables have to match the sum of the respective local variables, e.g. the total mission flight time $ToF_{mission}$ has to be equal to the individual segments' flight times added together. The launch date of the following segment is equal to the launch date of the previous one plus its flight time and the launch date of the first segment is equal to the launch date of the mission $LD_{mission}$.

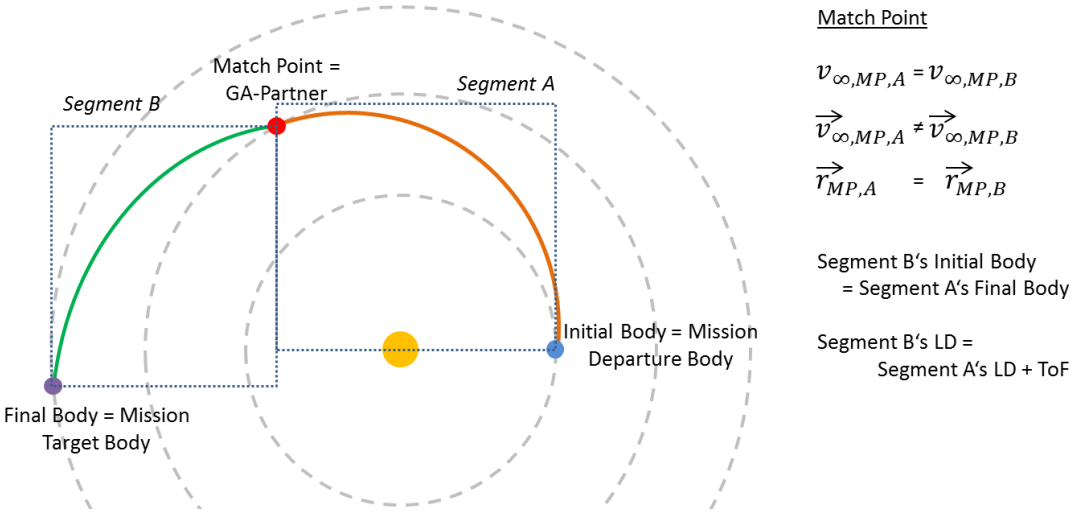

Figure 5-3: Structure of a two segmented mission, including the conditions defining the match point (MP) between the segments, always a gravity-assist partner. The hyperbolic excess velocities' magnitudes are identical for both segments at the match point, as are the position vectors. The velocity vectors are different (turned). The second segment's initial body is the final body of the previous segment.

Therefore, two kinds of variables, global and local, are defined. Global variables determine the overall mission. Local variables cannot be freely chosen, as they are constrained by the global variables and also depend on each other. The combination of discrete integer variables and non-discrete real valued variables makes the optimization a mixture between continuous and discrete optimization (Nocedal and Wright, 2006).

It has to be noted that while generally the thrust direction is part of low-thrust optimization, the respective variables are not used here as the trajectory model already sets the thrust as tangential. The thrust magnitude is a consequence of the remaining variables N_{rev} and ToF and therefore not explicitely expressed as variable in Table 5-1.

5.5 Repercussions between Algorithms, Control Variables and Search Strategy

The above made classification regarding local and global variables is mostly relevant for the selection of a search algorithm. Any algorithm that uses information of previous solutions for generating new solution candidates, i.e. evolutionary algorithms, can only use global variables for that, not local ones, because these are depending on each other and cannot be freely selected. Otherwise, e.g. the launch date of one segment could possibly not fit the criterion of being equal to the launch date of a previous leg plus its flight time. This restricts the capability of such search algorithms, because not all of the variables relevant for describing the problem can be evolved during the optimization.

The control variables in the described problem are not completely independent of each other. While there is no direct functional connection, the quality of a solution is depending on a combination of variable values. For example, the qualities of the variable of the gravity-assist partner and flight time of a given segment are depending on each other. A flight time which leads to a good trajectory segment from e.g. Earth to Mars is likely not suitable for a segment between Earth and Jupiter. The same is true for e.g. the turning angle, which e.g. is good for Jupiter, but might even violate constraints like minimum allowable distance for Mars. This means that for a recombination

Table 5-1: Summary of all relevant variables defining a gravity-assist mission and its segments using a shape-based trajectory model.

Variable	Description
global	
Mission Launch Date, $LD_{mission}$	The launch date of the overall mission, identical to the launch date of the first leg
Mission Flight Time, $ToF_{mission}$	Flight time of the overall mission, identical to sum of all segments' flight times
Mission Number of Revolutions $N_{rev,mission}$	Number of revolutions around the barycenter of overall mission, specific variable for shape-based trajectory model.
GA Partner, GA_{ID}	The actual partner for a gravity-assist, the essential variable to include the sequence in the optimization process
local	
Turning angle δ	Angle around which the planetcentric velocity vector is turned, to model gravity-assist effect (Figure 2-4), depends on pericentre distance of fly-by, i.e. it is constrained by the sphere of influence (maximum) and the closest allowed distance
Segment Launch Date, LD_{leg}	Sum of the previous segment's launch date plus its flight time; first segment's launch date is the mission launch date
Segment Flight Time, ToF_{leg}	Flight time for the segment, sum of all segments' flight time has to match mission flight time
Segment Number of Revolutions, $N_{rev,leg}$	Number of revolutions around the barycenter, specific variable for shape-based trajectory model, for one segment; the total of all segments' N_{rev} has to match mission N_{rev}
hyperbolic excess velocity, v_∞	Describes the relative approach/ departure velocity at gravity-assist partner; in x and y component

and mutation of the optimization variables it is likely that good combinations are actually separated and lead to less suitable results.

This already occurs for only the mere optimization of the trajectory as e.g. the number of revolutions and flight time is also relevant to each other. Nevertheless, the outset of the overall mission does not change so dramatically.

A change in the launch date will alter the launch conditions but not as dramatic as changing the flyby partner does change e.g. the suitability of a flight time that is kept from the original solution candidate to that new partner. For optimizing the whole sequence, the change of one variable, in this case the gravity-assist partner, has a dramatic effect on the whole mission and therefore the quality of all other variable values belonging to that exact mission design. This means that the fitness will react very sensitively to a change of that variable. The same is true for the actual times involved.

Due to the change of positions of the planets, which differ in their angular rates (due to different velocities), the time passing within one mission design actually strongly changes the suitability of previously good variable values. Radically changing flight time or launch dates will change the constellation of the planets in a way that changes the quality of a flyby. This is illustrated in Figure 5-4, where an example solution space is sketched in dependence of two random variables (note that the actual solution space has more dimensions than can be depicted). It is illustrated that small changes in one variable have the potential of significantly changing the solution candidate's quality.

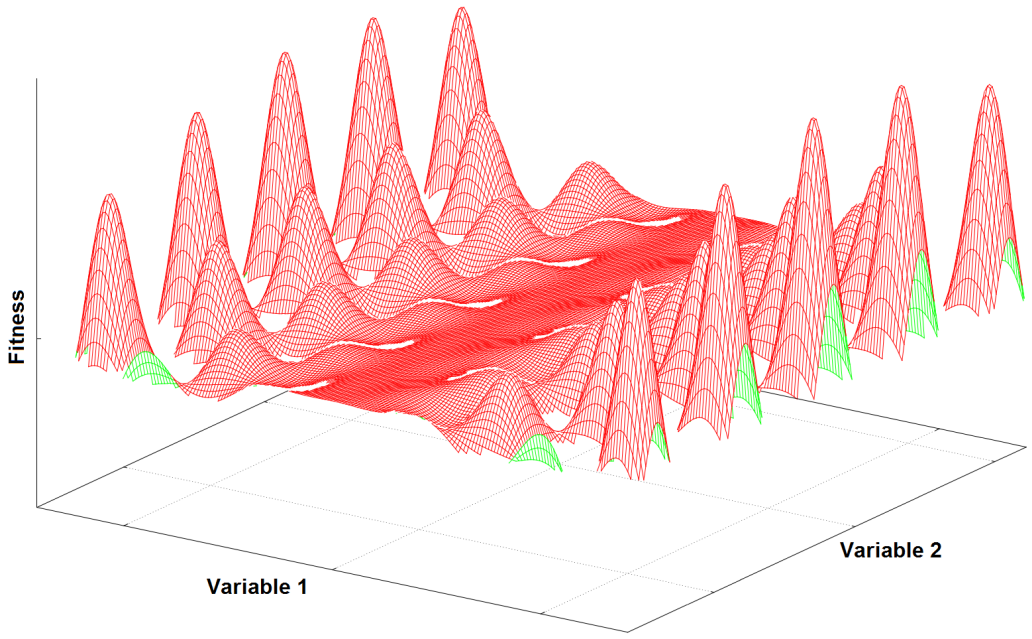

Figure 5-4: Sketch of a possible 2D solution space projection illustrating the sensitivity of a solution depending on changes in one or two variables. Some periodic structure might be possible due to repetition of GA opportunities based on synodic periods.

This has strong repercussions on the suitability of a search algorithm (affecting question no 4 of the scientific goals). As Evolutionary Algorithms depend on using information of other solution candidates for new ones and thus sorting out good properties from bad ones (schema theorem), the fact that a reasonable exchange, recombination or modification of variables is restricted renders the convergence properties of evolutionary algorithms no longer applicable or only in a restricted manner.

A possible change of search strategy could involve:

a) Use Random Search to find a number (e.g. 10) of good sequences (thus not using solution information to find new ones) and afterwards further optimize these, or
b) restrict the variables that are part of the recombination/ modification, thus basically combining the Random Search with the Evolutionary Algorithm; only make the global variables (e.g. mission flight time, GA-partner variables) part of the recombination and the local variables of the segments (e.g. flight time of one segment) random each time. This way overall good solution properties get not lost (e.g. if a flight time for the mission is known to be good)

Another issue is the number of gravity-assist partners changing the number of variables attached to each solution candidate. As described in Chapter 5.1.5 the solution candidate's variables are recombined with the help of vectors in case of evolutionary algorithms, i.e. they can only be recombined if the number of variables is identical. Otherwise, vector components are missing and therefore cannot be used to modify a solution. Consequently, this means only solution candidates with the same number of gravity-assist maneuvers can be used for recombination and mutation, as they have the same size of solutions, which is a necessary condition for using for Genetic Algorithms (Ashlock, 2006).

5.6 Constraining the Solution Space

For an increased efficiency of the search, the solution space can be constrained, thus eliminating non-useful regions of the solution space and improving the overall convergence of the search. This has been summarized in (Maiwald, 2017a) resp. (Maiwald, 2016) and is detailed below.

5.6.1 Δv Change

Eq. (2-1) describes the possible Δv change due to a gravity assist in dependence of v_∞, the pericentre distance to the planet r_{per} and the planets standard gravitational parameter μ_{pl}. For each gravity-assist partner the latter is a constant, i.e. there are only two variables describing the actual trajectory, i.e. v_∞ and r_{per}. For easier reference, Eq. (2-1) is repeated here:

$$\Delta v_{GA} = \frac{2\, v_\infty}{1 + \frac{{v_\infty}^2 \cdot r_{per}}{\mu_{pl}}} \tag{2-1}$$

It is clear from Eq. (2-1) that the possible Δv change is reciprocal to the pericentre distance for the hyperbolic passage of the gravity-assist partner – Eq. (2-1) is depicted in Figure 5-5 for the example of Earth, respectively its parts in Figure 5-6.

Looking at the limit values of this equation, it becomes clear that the maximum Δv for a given v_∞ is gained at a theoretical pericentre distance of 0 (the pericentre distance is always limited by the gravity-assist partner's body radius) and the minimum at infinity, see Eq. (5-24) and (5-25). This is to be expected: The larger the influence of the gravity-assist partner's gravitational force, the larger the effect of the gravity assist.

$$\lim_{r_{per}\to 0} \Delta v_{GA} = \frac{2\, v_\infty}{1+\frac{{v_\infty}^2 \cdot r_{per}}{\mu_{pl}}} = 2\, v_\infty \tag{5-24}$$

$$\lim_{r_{per}\to \infty} \Delta v_{GA} = \frac{2\, v_\infty}{1 + \frac{{v_\infty}^2 \cdot r_{per}}{\mu_{pl}}} = 0 \tag{5-25}$$

For a constant r_{per} however, the limit values for a changing v_∞ are both 0:

$$\lim_{v_\infty\to 0} \Delta v_{GA} = \frac{2\, v_\infty}{1 + \frac{{v_\infty}^2 \cdot r_{per}}{\mu_{pl}}} = 0 \tag{5-26}$$

$$\lim_{v_\infty\to \infty} \Delta v_{GA} = \frac{2\, v_\infty}{1 + \frac{{v_\infty}^2 \cdot r_{per}}{\mu_{pl}}} = 0 \tag{5-27}$$

Eq. (5-24) and (5-25) make it evident that there is a $\Delta v_{GA} \neq 0$ possible, however. Eq. (2-1) is a rational function and continuous (as it is a combination of two continuous functions, where the denominator cannot become 0 for $v_\infty \geq 0$) in [0,∞) and differentiable on the same interval. Therefore, it follows that according to Rolle's Theorem (Bronstein et al., 2000) an extreme exists.

Creating the derivative of Eq. (2-1) for v_∞ via the product rule, leads to:

$$\frac{\partial \Delta v_{GA}}{\partial v_\infty} = \frac{2}{1 + {v_\infty}^2 \cdot k} - \frac{4{v_\infty}^2 \cdot k}{(1 + {v_\infty}^2 \cdot k)^2}, \tag{5-28}$$

where k is a constant in this case, consisting of the term $\frac{r_{per}}{\mu_{pl}}$. To find the position of the extreme, the first derivate has to be equal to 0:

$$\frac{2}{1+{v_\infty}^2\cdot k}-\frac{4{v_\infty}^2\cdot k}{(1+{v_\infty}^2\cdot k)^2}=0 \tag{5-29}$$

$$\Leftrightarrow \frac{2}{1+{v_\infty}^2\cdot k}\left(1-\frac{2{v_\infty}^2\cdot k}{1+{v_\infty}^2\cdot k}\right)=0$$

$$\Leftrightarrow v_{\infty,ex}=\sqrt{\frac{1}{k}}=\sqrt{\frac{\mu_{pl}}{r_{per}}}.$$

To determine whether this really is an extreme value and of which kind, the second derivate of Δv has to be investigated, which is:

$$\frac{\partial^2\Delta v_{GA}}{\partial {v_\infty}^2}=\frac{4v_\infty\cdot k}{(1+{v_\infty}^2\cdot k)^3}({v_\infty}^2\cdot k-3), \tag{5-30}$$

Inserting $v_{\infty,ex}$, gives:

$$\frac{\partial^2\Delta v_{GA}}{\partial v_\infty^2}(v_{\infty,ex}=\sqrt{1/k})=-\sqrt{k}<0, \tag{5-31}$$

This shows that $v_{\infty,ex}$ is a maximum, which matches the graphs in Figure 5-5 resp. Figure 5-6 (left). Inserting the result of Eq. (5-28) into Eq. (2-1) reveals the maximum possible Δv for a given r_{per} to be:

$$\Delta v_{max}=\frac{1}{\sqrt{k}}=\sqrt{\frac{\mu_{pl}}{r_{per}}}. \tag{5-32}$$

This is identical to the circular orbit velocity for the pericentre distance of the hyperbola and also equals $v_{\infty,ex}$.

The influence on Δv due to a change of the pericentre distance r_{per} is expected. The closer the spacecraft is to the gravity-assist partner during the hyperbolic pericentre passage, the stronger its influence and therefore effect on the turning angle δ resp. the Δv-gain. However, the pericentre distance is restricted by physical properties (e.g. the gravity-assist partner's body radius) and also system properties of the spacecraft. Radiation or atmosphere effects need to be regarded when designing the fly-by trajectory. Therefore, the possible maximum is influenced by the system and physical constraints.

The influence on Δv due to a change of the velocity at infinity v_∞ is less straightforward. With the help of Figure 2-4, it becomes clear that the case of $\Delta v = v_\infty$ can only occur for an equilateral triangle, i.e. when the turn angle δ is 60°. The same result occurs, when Eq. (2-2) is transformed and for v_∞ the value $v_{\infty,ex} = \Delta v_{max}$ is inserted. In this case Eq. (2-2) becomes:

$$\Delta v_{GA}=2\,v_\infty\,\sin\frac{\delta}{2} \tag{2-2}$$

$$2\arcsin\left(\frac{1}{2}\right)=\delta, \text{for } \Delta v_{GA}=v_\infty$$

which is true for a deflection δ of 60°. Another expression for the turn-angle δ is (Bate et al., 1971):

$$\frac{1}{e}=\sin\left(\frac{\delta}{2}\right). \tag{5-33}$$

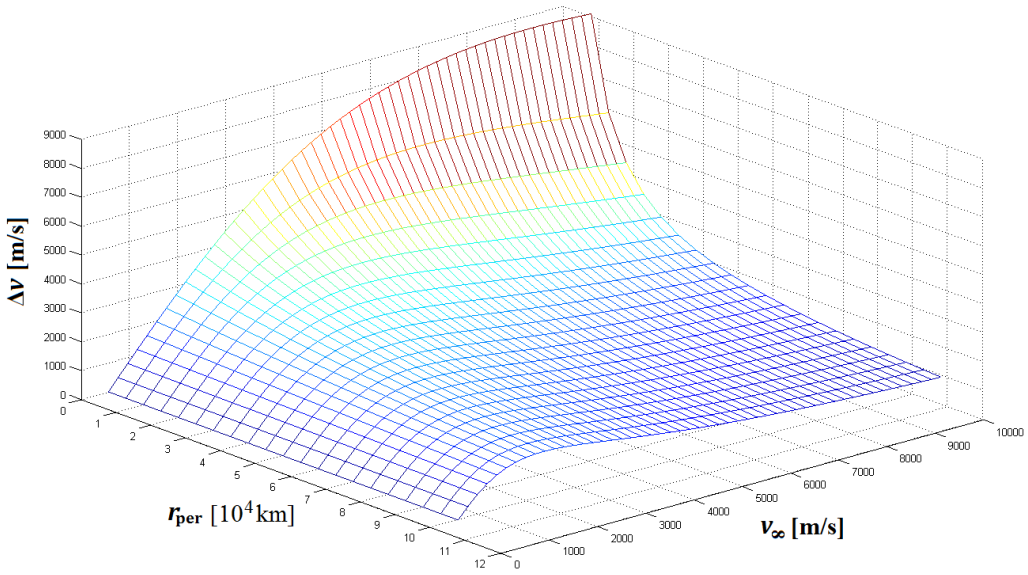

Figure 5-5: The Δ*v* change of a gravity-assist maneuver as function of hyperbolic pericenter distance r_{per} and v_∞ for the example of an Earth fly-by (μ_{pl} = 3.98 x 10^{14} m³/s²). (Maiwald, 2017a)

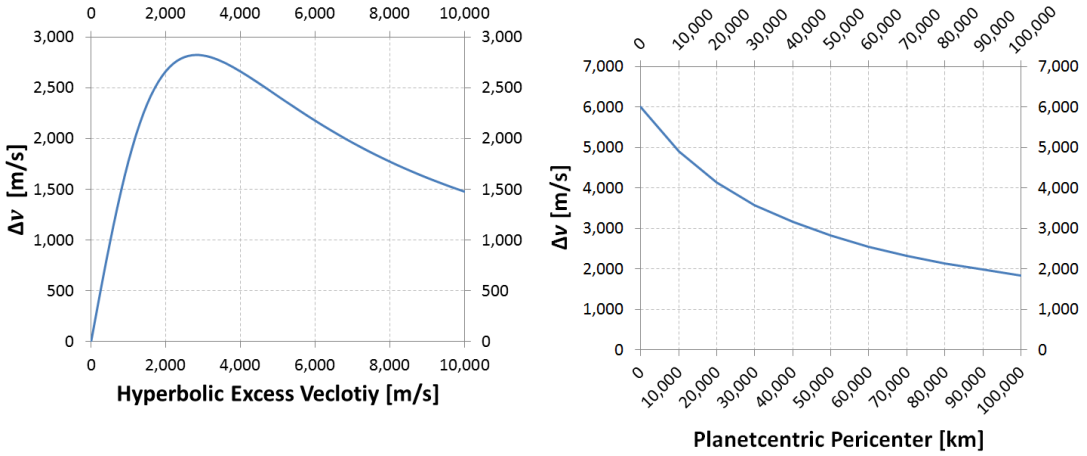

Figure 5-6: Δ*v* over pericenter radius r_{per} for the example of Earth as gravity-assist partner and a v_∞ of 3 km/s (right). Δ*v* over v_∞ for the example of Earth as gravity-assist partner and a pericenter distance of 50,000 km. (left)

This means, the maximum Δ*v* occurs for an eccentricity of 2.

Due to Eq. (2-2), it is apparent that for small values of δ, the *Δv* effect is also small, because of the sinus-part of the term. Subsequently, a larger δ, in the area of 180° would be preferable from this point of view, to gain the most from this part (for δ = 180°, the sinus becomes 1 as only δ/2 occurs in the term). From a theoretical point of view, this is to be expected, although realistically it is not possible as the radius of the gravity-assist partner is larger than 0.

Figure 5-7 illustrates that for δ to increase, v_∞ has to decrease for keeping the same *Δv*. The change of direction, due to the influence of the planet's gravitation, increases the more time is spent in its vicinity. Therefore, a small approach velocity is beneficial for the turn angle δ.

However, it is evident from Eq. (2-2) that regarding the Δv, a decreasing v_∞ directly affects the linear part of the term. Both effects balance each other at an angle of 60°, which is why there a maximum is found.

The movement of the spacecraft and the planet are not coupled until the spacecraft enters the sphere of influence of the gravity-assist partner – the v_∞'s direction and magnitude are basically random, depending on the location and time spacecraft and gravity-assist partner meet (and if the trajectory is not timed correctly, these two do not meet at all). However, the v_∞ is the boundary condition that hands over the properties of the spacecraft's trajectory between the heliocentric and planetcentric coordinate systems and vice versa in case of the outgoing velocity.

While Eq. (2-2) shows that a large v_∞ increases the gravity assist's Δv effect – in the linear part of the equation – it is also clear that a small turn angle δ reduces this effect. The explanation for this relation is analogue to the effect on the turn angle δ: a large v_∞ means only a relatively short amount of time (relative to a smaller value of v_∞) close to the gravity-assist partner and therefore subject to its gravitational influence. Consequently, the amount of turning (i.e. the value for δ) is limited and as, can be seen in Figure 2-4, only the turning of the v_∞ vector results in the occurrence of a Δv change.

Generally speaking, a large magnitude of v_∞ is not necessarily desirable because it:

- usually needs effort – in terms of energy – to be achieved, and
- the Δv change decreases after passing $v_{\infty,ex}$.

This relation can and should be exploited when searching for optimal gravity-assist sequences for a given mission. The selection of to be investigated trajectories could be reduced.

If the described effect is used for gravity-assist sequence optimization, the benefit is its independence of the spacecraft's system properties. The effect depends only on the trajectory itself and occurs for low-thrust maneuvers and impulsive maneuvers alike.

An example of a transfer for a spacecraft to a planet is pictured in Figure 5-8. The planetary orbit is assumed to be circular for clearer representation. At the meeting point of the spacecraft and planet the involved velocity vectors relevant for the gravity assist are shown, especially the relation between the heliocentric, incoming velocity and the planet's velocity should be focused on, as this results in v_∞, in this case the incoming velocity $v_{\infty,in}$. This shows that the meeting point and thus the exact trajectory strongly influence the v_∞.

For an encounter where the spacecraft's and the planet's heliocentric velocities are more aligned in the same direction (i.e. in this example at a position more forward counter clockwise) v_∞ would become smaller. Besides the angle, also the magnitude of the spacecraft's velocity has an influence.

The two vectors would be parallel if both partners would encounter each other at the spacecraft's apsides. The magnitudes would be equal if both orbits would be in fact circular. This would also mean that v_∞ is 0 and thus also the Δv as the gravity assist encounter would have degenerated into a rendezvous. Somewhere before this lies $v_{\infty,ex}$, where Δv has its largest value.

If the transfer trajectory and the point of encounter are known, this immediately provides a value for v_∞ and the planetcentric pericentre distance of the spacecraft. This in turn immediately provides the theoretically possible maximum Δv resp. $v_{\infty,ex}$, the velocity at infinity, where the maximum Δv occurs. Any trajectory which has a v_∞ that is too far distanced from $v_{\infty,ex}$ could be discarded, especially if it is too large, because in this case more effort than necessary has been made to achieve this trajectory, even though it does not provide an optimal Δv.

The diagram in Figure 5-6 (left) can be used to "advise" an optimization process on which trajectories are worthwhile to be investigated and which are not. Trajectories resulting in a v_∞ outside a certain interval around $v_{\infty,ex}$ could be excluded from further evaluation, with little effort, simply by determining the v_∞ related to this trajectory.

To use this guidance for optimization of propellant mass or transfer time, a reasonable scheme has to be established. If propagating the spacecraft trajectory forward in time, it is possible that by restricting the range of possible v_∞ at a previous encounter it becomes impossible to reach a near optimal v_∞ at a subsequent gravity-assist partner. This is especially true for outer solar system partners, where fewer gravity-assist options exist than in the inner system, where combinations of Venus, Earth and Mars are more available, because of the closer distances between the three planets.

Assuming however that a large Δv change by gravity assists reduces the propellant mass necessary for a mission and by trying to achieve this Δv change with as little gravity-assist partners as possible (by targeting the area around the maximum Δv change for each of them) a shorter flight time should be achieved. Therefore, using the Δv-v_∞ relation could improve the results, especially if propellant mass is to be optimized, which relates directly to the achievable Δv change.

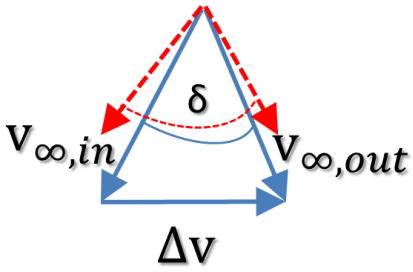

Figure 5-7: Change of the turn angle δ for decreasing values of v_∞.

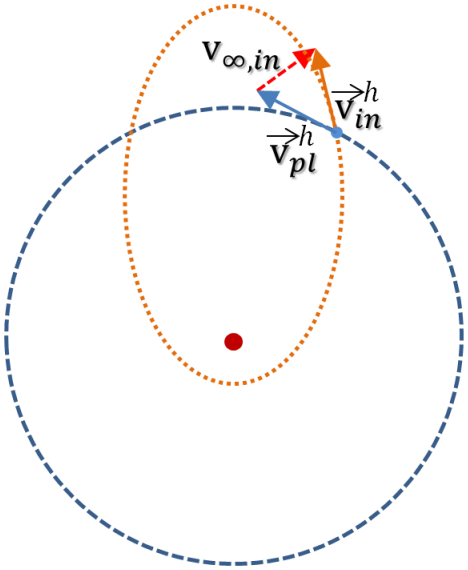

Figure 5-8: An example transfer trajectory (dotted line) that leads to a planetary encounter (the planet's orbit is dashed). Visible are also the heliocentric (upper index h) vectors of the spacecraft's incoming velocity (index in) and the planet's velocity (index pl). It is indicated how v_∞ can be derived from these vectors.

To apply this for optimization, a stepwise review of the encounter conditions is useful. For each gravity-assist partner candidate it is investigated whether or not the v_∞ is within a set interval of the respective $v_{\infty,ex}$. Afterwards other trajectory constraints are reviewed, e.g. minimum allowable distance to the planetary body. The order of this is of course not important and has been chosen here to have the comparison of v_∞ with $v_{\infty,ex}$ stand out.

It is further checked if the final state has been reached after the last leg of the trajectory. In this case a final state could be e.g. rendezvous with a planet as mission target, but it could also mean launch conditions, in case the optimization occurs backward.

There are a number of open issues regarding the previously described relations. It is for example interesting to note that the maximum Δv resp. $v_{\infty,ex}$ are equal to the circular velocity at the given r_{per} and it is currently unclear why this is the case. When calculating the pericentre velocity for the respective hyperbola, a value of $\sqrt{3}\ \Delta v_{max}$ (equaling the circular velocity at this distance) occurs, i.e. times $\sqrt{3/2}$ of the escape velocity. The significance of this has to be reviewed further.

If an evaluation is using the suggested constraint, the interval size around $v_{\infty,ex}$ where a trajectory is still regarded as valid or not, will have a strong influence on the optimization result. For the reasons mentioned above, i.e. a too narrow interval could prevent optimal solutions in later trajectory legs. Therefore, the interval size has to be tailored correctly. This issue is analyzed and discussed in Chapter 7.6.

5.6.2 Gravity Assist Partner Pool

Theoretically, for a sequence of gravity assists it is possible to randomly switch from one body to another during the course of the sequence to gain "free" Δv, until finally the target body is reached. However, from a global perspective, especially also considering flight time, it is not reasonable to include just any body as potential next partner. For instance, on the way to Saturn, flying from Jupiter back to Mercury for a potential gravity assist is likely harming the overall mission performance as the Δv to actually adapt the trajectory for flying past Mercury will already cost a large amount of Δv and in any case cause an increase in flight time. Therefore, it is advisable to restrict the pool of possible partners for gravity-assist partners.

It is proposed to use a parameter for the optimization that actually sets the limit of steps that are allowed to be taken beyond the current position. E.g. a limit of 1 would mean no planet further inwards than one or none further outwards can be selected as gravity-assist partner, i.e. with such a limit going from Jupiter back to Mercury would be impossible – only Mars and Saturn itself would be legitimate partners. This is further discussed in Chapter 7.6.

5.7 Objective Function

The objective function describes the optimization goals and its formulation influences the usefulness of the results. A badly formulated optimization function might actually hinder the process of optimization by under-rating valuable solutions.

Furthermore for the problem at hand, the function is influenced by two parts: the effect of the gravity assists (addressed by turning the velocity vector) and the Δv requirements for the trajectory legs, which together lead to an overall Δv requirement on the spacecraft and a resulting flight time. Violation of mission constraints, e.g. exceeding the arrival time at the target, needs to cause a penalty in the objective function to favor solutions, which are not violating these constraints.

From an optimization point of view, the Δv change of a gravity assist should make a gravity-assist trajectory perform better than a non-gravity-assist trajectory, i.e. no further incentive needs to be given for that. Consequently, the following objective function is used:

$$J = k\frac{1}{\sum_n \Delta v_n}, \tag{5-34}$$

whereas k is a factor that allows scaling of the fitness. This way it is possible to scale the fitness into digits > 1 for mission Δv values which are considered good and thus make well performing solutions easier recognizable and prevents numerical issues, caused by values close to 0. The index n runs through the number of segments for a given solution candidate.

Including the flight time in a similar manner is possible, but involves the problem that the flight time is usually constrained for the optimization by the user (to a maximum), unlike the Δv. Consequently, for missions with large Δv values, the fight time would be dominating the fitness term.

Eq. (5-34) is no simple linear addition of individual segments' fitness values, but adds them to a total, i.e. a global perspective is kept up during the evaluation, as:

$$\frac{k}{\sum_n \Delta v_n} \neq \frac{k}{\Delta v_1} + \frac{k}{\Delta v_2} + \frac{k}{\Delta v_3} + \cdots + \frac{k}{\Delta v_n}, \tag{5-35}$$

The right side of Eq. (5-35) is a simple linear addition of the individual segments' objective function. If the individual segments are optimized based on their individual Δv demands and such an objective function for the total mission is used, this would result in a series of local optima combined in a sequence, which not necessarily represents the overall optimum. A formulation as on the left side of Eq. (5-35) only considering the total Δv of all segments for evaluation automatically omits the segments' point of view and only regards the complete mission. This can include mission segments, which are not the best performing individually, but in sum would. Thus, the evaluation of the solution candidate occurs taking all its segments into account. Individual segments are *not* evaluated. Optimality of individual segments would be achieved if each segment is optimized individually however a series of optimal segments does not guarantee overall optimality. Conditions beneficial for one segment's fitness might have negative consequences for the overall mission. By keeping up the overall mission perspective and evaluating each solution candidate in completeness it is ensured that the best mission result is found.

6 Formulation of Optimization Method

The two previous chapters, dealt with the application of Tisserand's Criterion on gravity-assist optimization for low-thrust missions and established the building blocks for a method. This chapter now formulates the actual steps of the optimization method to be used in the coming section of the thesis by formulating all necessary steps to ensure that:

- *the gravity-assist partner(s) are part of the optimization,*
- *the analysis and evaluation of the search space is sufficiently complete to also explore regions beyond the "experience" of the mission analyst,*
- *mission feasibility can be evaluated.*

For impulsive missions Tisserand graphs (Chapter 2.3) allow the planning of energy feasible gravity-assist sequences for a given spacecraft and impulsive missions. Due to lack of phasing considerations, the planning with the help of Tisserand graphs does not constitute a sufficient condition, but it is a necessary one. Any sequence that is finally used is part of those that can be determined with the help of Tisserand graphs.

To use Tisserand graphs for the optimization of the sequence, the search has to involve the parameters for the initial body and the gravity-assist partner. Moving along a given contour in a Tisserand graph, defined by the respective hyperbolic excess velocity relative to the initial body, $v_{\infty,1}$, the spacecraft meets the contour of the gravity-assist partner with a different hyperbolic excess velocity, $v_{\infty,2}$, (note the magnitudes can be the same, but need not be) in reference to that planet. The heliocentric energy remains constant, however. At the crossing points of different contours, the transfer from one planet to the next is possible.

To automatically search reasonable paths and potential partners, i.e. such crossing points, the search space is discretized by selecting a given set of hyperbolic excess velocities (Strange and Longuski, 2002). Based on Tisserand's Criterion the number of these crossing points is finite for a given set of these excess velocities, thus allowing a structured search for them.

However, this differs for the application of low-thrust, as the energy is no state variable (see Chapter 4.3.6), but depends on the trajectory taken by the spacecraft to get to a certain position. As opposed to the energy conservation for a ballistic flight, in principle the final energy of a transfer leg for a low-thrust mission, is not restricted – the correction term can have any value (realistically restricted by the available propellant) - and a priori a selection of a path of the sequence is not possible. Consequently, this part of the optimization has to become an optimization variable, see Table 5-1.

One possibility is to search through the possible sets of variable values systematically: Run through a set of values for the arrival velocity (e.g. from 1 to 10 km/s in 1 km/s steps) at a given gravity-assist partner, run through a set of values for the turning angle δ, run through a set of next gravity-assist partners and so forth. This is illustrated in Figure 6-1. Chen et al. (Chen et al, 2008) implicitely demonstrate how a branch of one solution can be analysed, by selecting e.g. fixed v_∞ values for evaluation.

As the involved variables have an infinite number of possible values, no structured search can analyze the complete search space. Whatever step size is selected to run through the respective intervals, only a finite amount of solutions is evaluated. Therefore, a structured search is of no advantage, because theoretically it is possible that the optimum solution lies in areas, which are not evaluated. A heuristic search is consequentially preferable over a structured search, if an optimum is searched. For the heuristic search, the variables are randomly initiated and subject to the respective optimization algorithm. Therefore, there is at least a chance of finding the optimal solution.

The randomly initiated variables also include the gravity-assist partner, possibly restricted by a pool of partners (see Chapter 5.6), depending on the actual mission that is to be analyzed. As the final energy resulting from a possible trajectory's thrust history cannot be determined a priori, the arrival conditions at the partner can also not be derived from this. The optimization algorithm selects the heliocentric velocity at the second body, $v_{\infty,2,}$ the first potential gravity-assist partner, and from this also derives a corresponding vector (one component being an optimization variable, the other one following from the vector magnitude). Analogously to Tisserand graphs this means that the search concentrates on finding a connection, a crossing point between two contours, but

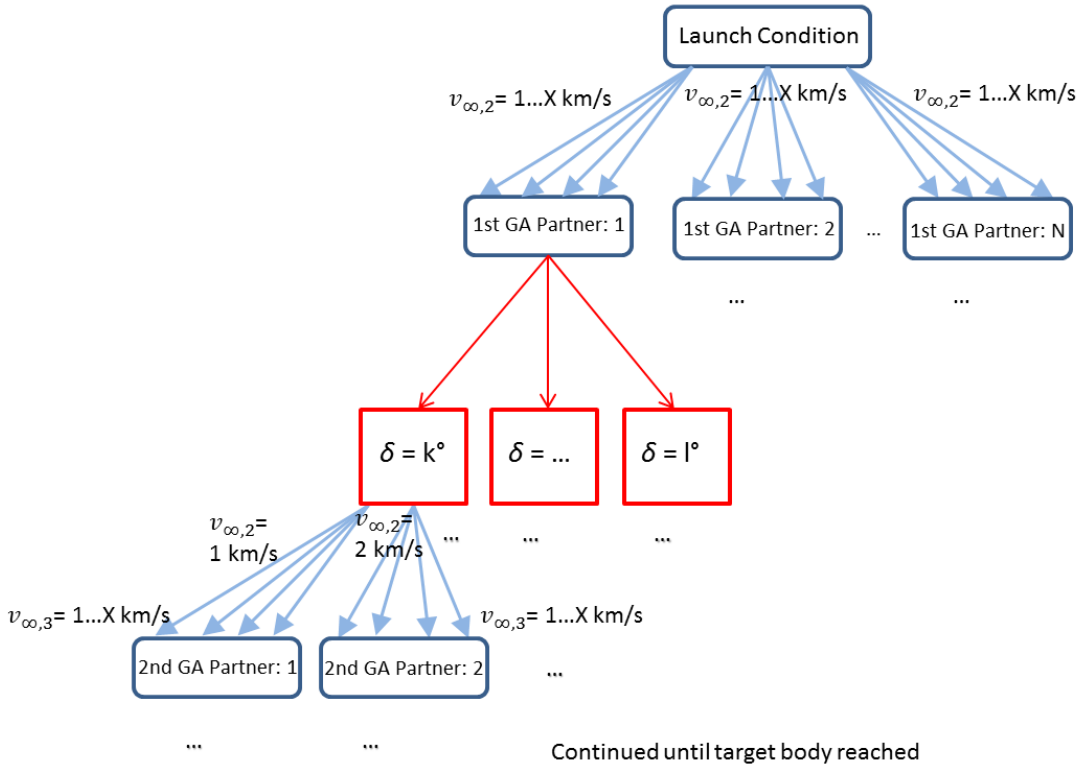

Figure 6-1: An example of a branch of a solution tree (starting at the top), illustrating the exploration of different values for the hyperbolic excess velocities v_∞, the gravity-assist partner and the gravity-assist effect, here given by the turning angle δ, ranking from *k* to *l*, depending on constraints for minimum and maximum values.

these contours are not known a priori.

Once the arriving condition is set for a given solution candidate, the effect of the gravity assist has to be determined. For this either the pericentre distance r_{per} or the turning angle δ is randomly set, constrained by minimum and maximum pericentre distances given by the gravity-assist partner (both are related to each other and the total Δv change via Eq. (2-2) and (2-3)).

The effect of the gravity assist can essentially be modeled with the help of Tisserand's Criterion, not including the correction term. First, for the part of the gravity assist it is assumed that no thrusting occurs as the thrust is only modeled in sections that are calculated using the trajectory model (see Section 5.2), which is also satisfying navigational requirements of no thrusting during planetary encounters (García Yarnoz et al., 2006). Furthermore, given the short amount of time needed for the gravity-assist maneuver in relation to the whole transfer between two planets, the correction term would be very small, even if thrust were included for that part of the mission simulation.

Figure 6-2 shows a possible transfer between two gravity-assist maneuvers, described by Tisserand graphs, as modelled by a shape-based trajectory, showing that for the parts of the thrusting, the orbital parameters are changed continuously until a condition beneficial for a gravity-assist maneuver is reached.

It is noted that if the mission's target body is selected as next planetary partner by the algorithm then the arrival condition is no longer a hyperbolic excess velocity but an arrival velocity matching the planet's own velocity. Further constraints for this method are the number of gravity assists attempted by the algorithm and e.g. the turning angle δ (thus setting a certain Δv), which could be constrained to be close to a possible maximum Δv, as described in Section 5.6.

Summarizing, the optimization of the sequence will happen along a route described in the following. As outlined in Chapter 3, three paths exist for optimizing trajectory and GA-sequence in

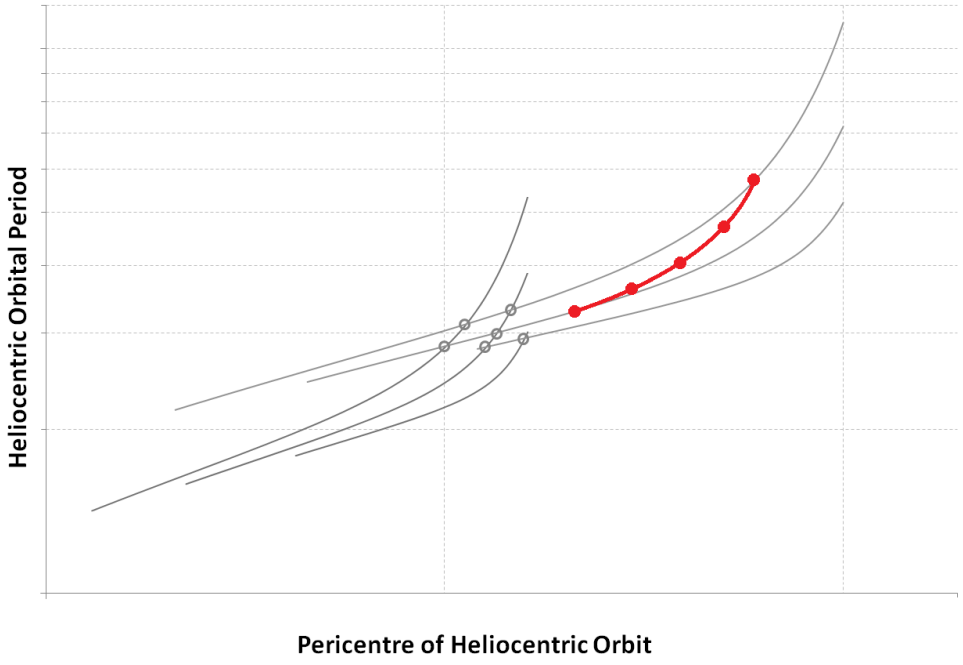

Figure 6-2: Example graphs (grey) for orbital period as function of heliocentric pericenter and a possible transfer between the respective energy conditions (as represented by the graphs) via a thrusting scheme (red, dotted line).

combination (see Figure 3-2). Finding a sequence of gravity-assist partners and then finding a trajectory that can follow this sequence has two major drawbacks:

a) As the validity of the sequence can only be determined a posteriori, a number of sequences that are invalid are optimized beforehand, without use, i.e. computation effort is wasted.
b) Trying to find an optimized trajectory for a given sequence does not mean the overall optimal goal is achieved. While the sequence might be optimal, in combination with the phasing optimality of the sequence and thus mission path may be reduced.

Consequently, this possible path of optimization is not advisable.

The mixed optimization path (option (C) in Figure 3-2) also has the disadvantage of requiring large efforts in computation for sequences, which in the end will not be used and thus reducing the efficiency of the tool and method. Therefore, the optimization should be sequential (option (A) in Figure 3-2), while at the same time include non-gravity-assist trajectories as comparison to ensure the better performance of the found sequential trajectory. Forcing gravity assists to occur might lead to an optimal gravity-assist mission design but needs not be overall optimal or even better than a non-gravity-assist mission. Furthermore, the optimization constrains differ between rendezvous and flyby. While the final mission goal might be either, for each gravity-assist partner only a flyby is a viable option, rendezvous are not possible.

Thus, the following stepwise optimization is conducted. A preliminary version has been published in (Maiwald, 2016) and updated in (Maiwald, 2017a):

(1) setting the constrains of the mission (e.g. starting body, target body, launch window),
(2) calculating an optimal non-gravity-assist trajectory, which serves as benchmark for the gravity-assist trajectory (i.e. any final gravity-assist sequence has to perform better than this benchmark without any gravity assists),
(3) create a population of solution candidates for each value of the number of gravity assists from 1 to a (user specified) maximum, with random initial values for all variables, where:
    a. constraints and variable interrelations are observed to create a valid mission sequence,
    b. the trajectory between the respective segment's bodies is modeled according to Chapter 5.2,
    c. gravity assist effect is considered by application of the turning angle on the planetcentric approach/ departure velocity (constrained as mentioned above),
(4) optimize the solution candidates within each population, recombination of mission global variables (see Table 5-1) occurs, mission local variables are set in dependence to global and local variables (e.g. the legs' flight times add up to the mission flight time, the launch date of the second trajectory segment depends on the arrival date of the first one),
(5) rank each population's final solution and the no-gravity assist benchmark to find final solution.

The creation of populations with a fixed number of gravity assists has two main advantages. First, the solution structure is identical within a population. Thus, within a population variables can be recombined among solution candidates. Second, no infinite loop is created, as theoretically possible if the number of gravity assist is open (although other measures can be used to prevent that). Creating separate populations means that information of solutions with different numbers of gravity assists cannot be transferred between these types of solutions. However, it is unlikely that variable values fitting e.g. a two-gravity assist solution are also (near) optimal for three-gravity-assist missions, e.g. regarding flight time, so this obstacle is not regarded as critical.

The described method has two major advantages over previous attempts of optimization gravity assist low-thrust missions (see Chapter 2.4.2). First, in difference to previous methods the gravity-assist partner is part of the optimization. Second, the optimizer is forced to conduct gravity assists and not find them accidentally while searching for mission trajectories. Therefore, it is ensured that a large number of gravity-assist mission candidates are evaluated, overcoming the obstacles described in (Carnelli et al., 2009).

It is noted that the trajectory between mission bodies is not subject to optimization in a segment point of view, only globally. A sequence of optimal trajectory segments is not necessarily an optimal solution for the overall mission. Therefore, the mission perspective is kept for the optimization.

For clarification, the list of control variables and their constraints is given in Table 6-1.

Table 6-1: List of control variables for the process of gravity- assist optimization and corresponding possible constraints.

Control Variable	Constraints
Gravity Assist Partner	Restricting the pool of possible partners by proximity, e.g. not including outer planets for missions in the inner solar system (e.g. Saturn being not a viable partner for missions targeting Mercury). A maximum number of gravity-assist maneuvers, to prevent infinite loops during the optimization
Gravity Assist Effect: turning angle δ or planetcentric pericentre distance r_{per}	Turning angle is dependent on planetcentric pericentre distance. This might be restricted to prevent impact on the planet or moving through regions of intense radiation. The sign of δ provides information about the position relative to the planet of the spacecraft during the gravity-assist. It thus has a similar role as a B-plane for a three dimensional case.
Arrival Velocity (identical to departure velocity at gravity-assist partner)	Possbily mission driven, e.g. if certain instruments or scientific reason prescribe a certain velocity, or maximum obtainable Δv, see Section 5.6

7 Evaluation of the Method

To evaluate the method, a software tool has been created. It is used to run experiments to analyze the:

- *applicability,*
- *performance, and*
- *usefulness*

of the sequence optimization method described in Chapter 6.

7.1 Tool Description

The tool applying the method described in Chapter 6 consists mainly of two parts: The trajectory model and the sequencer. Both are interlinked via the optimization and use a global search algorithm to find a common optimum for the trajectory. In parallel, a trajectory not using a gravity-assist sequence is optimized with the same trajectory model to evaluate the gain or possible loss in mission performance (in terms of mission Δv) by the introduction of gravity-assist maneuvers. The principle of the method is not depending on a three-dimensional solution and the originally investigated Tisserad graphs also only allow a planar view. Thus, the current tool incorporates only two dimensions, i.e. non-inclined orbits are assumedfor the calculations, by projecting them onto the ecliptic. The example missions mentioned in the introduction combining gravity assists with low-thrust propulsion, all used planets as gravity-assist partners, and therefore only planets are included as potential partners to evaluate the method. Other partners are conceivable, but are not implemented at this stage.

A major component of the tool are the ephemerides, based on NASA's Horizons database[1], which is able to generate ephemerides for all major solar system bodies. With the help of all participating bodies' orbital data, the optimizer creates position and velocity vectors required for solving the optimization problem. A detailed description of the tool and its interface can be found in Appendix 1.

The trajectory model is the polynomial model as described by Wall and Conway (Wall and Conway, 2009), see Section 5.2, with the respective control variables, i.e. number of revolutions around the barycenter, flight time and launch date. The thrust is assumed to be tangential as described before. The output is the radius history and thrust history as given by Eq. (5-9), resp. by Eq. (5-20), as well as the required Δv, along with the launch date, flight time and gravity-assist parameters, i.e. turning angle, encounter date and gravity-assist partner.

[1] JPL Horizons online solar system data ephemeris computation system, http://ssd.jpl.nasa.gov/?horizons

The code allows constraints based on user input in the following parameters:

- Maximum allowed numbers of revolutions ($N_{\mathrm{rev,max}}$),
- launch date and launch date window ($LD_{mission,max}$ and $LD_{mission,min}$),
- fight time maximum and minimum (ToF_{max} and ToF_{min}),
- maximum allowed thrust ($T_{\mathrm{max,allowed}}$), and
- minimum allowed solar distance ($R_{\mathrm{min,allowed}}$).

While the first constraint directly influences the search space by restricting the possible selection of the number of revolutions, violations of the last two constraints cause penalties to the fitness. Small violations should not lead to a complete removal of such solution candidates as this would mean risking to loose potentially good control information (see schema theorem, Chapter 5.1.5). Therefore, the optimizer checks whether the minimum allowable radius is violated and if it is the case, the fitness is modified by the following factor:

$$J^* = J \cdot \frac{R_{\mathrm{min,is}}}{2 \cdot R_{\mathrm{min,allowed}}}, \tag{7-1}$$

where $R_{\mathrm{min,is}}$ is the actually occurring minimum radius, J the ordinary fitness as determined by Eq. (5-34) and J^* the penalized fitness for violating the minimum allowable radius $R_{\mathrm{min,allowed}}$. The factor $1/2$ ensures that the drop in fitness is significant even if the violation is minor and the factor of $R_{\mathrm{min,is}}$ ensures that a very small minimum radius compared to the allowed radius results in a severe penalty to the fitness. However, especially small violations do not lead to an automatic drop of the solution candidate out of the population based on its small fitness, so that possible beneficial attributes of that trajectory remain in the population. An analogous penalty is created for violations of the maximum allowed thrust acceleration $T_{\mathrm{max,allowed}}$:

$$J^* = J \cdot \frac{T_{\mathrm{max,allowed}}}{2 \cdot T_{\mathrm{max,is}}}. \tag{7-2}$$

Since the limit is an upper limit, the allowed value is in the numerator.

Furthermore, the calculations are requiring maximum and minimum flyby radii. These can be derived from the sphere of influence in the first of which case and the planet radius added with a distance accounting for e.g. atmosphere for the latter. For some planets, e.g. Jupiter, the minimum distance is set by radiation limitations. These constraints are summarized in Table 7-1.

Table 7-1: List of the constraints regarding minimum and maximum (i.e. sphere of influence) allowed flyby radii and gravitational parameter for all bodies relevant for the calculations as used within the code.

Body	Gravitational Parameter μ	sphere of influence Radius	Minimum allowed Radius
Sun	2.958718364 x 10^{-4} AU³/d²	n/a	0.25 AU
Mercury	4.882888633 x 10^{-11} AU³/d²	7.553476 x 10^{-4} AU	2,739 km *
Venus	7.246295917 x 10^{-10} AU³/d²	4.200136 x 10^{-3} AU	6,351 km *
Earth	8.873925462 x 10^{-10} AU³/d²	6.183155 x 10^{-3} AU	6,678 km *
Mars	9.542814315 x 10^{-11} AU³/d²	3.836898 x 10^{-3} AU	3,693,5 km *
Jupiter	2.831629481 x 10^{-7} AU³/d²	3.228610 x 10^{-1} AU	356,990 km (5 Jupiter radii)**
Saturn	8.472592149 x 10^{-8} AU³/d²	3.649733 x 10^{-1} AU	120,000 (2 Saturn radii)**
Uranus	1.290955488 x 10^{-8} AU³/d²	3.462567 x 10^{-1} AU	51,300 (2 Uranus radius)**
Neptune	1.531755475 x 10^{-8} AU³/d²	5.795455 x 10^{-1} AU	49,560 (2 Neptune radii)**

*300 km above surface, based on (Jehn and Schoenmaekers, 2014), ** (Strange and Longuski, 2002), respectively coded in AU, here referred in km for easier recognition/ comprehension

Gravity assists can be conducted outside the sphere of influence, however this would change the trajectory even before the encounter with the planet and due to the small change is not regarded here, resulting in a patched-conic like approach based on the selected trajectory model.

7.2 Test Strategy

The most basic heuristic search is randomly selecting values for the control variables and evaluating the respective outcome based on Eq. (5-34) (i.e. the associated Δv) of the mission. All solution candidates are ranked and the best result is selected as the "optimum". This is obviously a very basic method of heuristic search and does not involve utilizing information of previous solutions for creating the next. As only part of the variables could be used for evolutionary optimization (see Chapter 5.5) it is relevant to establish the search performance in case no evolutionary approach is used at all. This way, it can be determined if the extra effort of applying evolutionary search strategies leads to an improved outcome or not. For this purpose Random Search results are compared with those of evolutionary search strategies. Differential Evolution and Simulated Annealing are tested according to the influence of their parameters, e.g. population size for Differential Evolution and temperature drop method for Simulated Annealing. Furthermore, the outcome of applying the constraints as described in Chapter 5.6 is tested. Finally it is attempted to reproduce existing trajectories.

7.3 Optimization with Random Search

To establish a benchmark of the search's performance for further optimization algorithms, a random search is applied to the given problem. This is done by randomly initiating all variables and calculating the respective mission outcome. For each number of gravity assists a separate population of solution candidates is created. The mission used as example has been a rendezvous transfer from Earth to Jupiter with initial and final hyperbolic excess velocities of 0.

7.3.1 Optimization Settings

The properties of the calculations are listed in Table 7-2. The algorithm's only property to influence the search is the population size, i.e. how many samples are randomly taken. Based upon the outcome of the Differential Evolution calculations, Chapter 7.4, the first set's population has been set to 4,450 population members. This is derived from the Differential Evolution population size of 50 times the average number of generations of approx. 90 for the gravity-assist solutions.

For the next batch of Random Search results, an increase of factor 4 has been selected and after that a factor of 2.5 both in similar steps as for Differential Evolution. This resulted in 17,800 resp. 63,000 samples. As the number of generations is not identical for the respective populations with different numbers of gravity assists, it is not possible to create completely identical numbers of evaluated solution candidates for Random Search and Differential Evolution. Nonetheless, the amount of evaluated solution candidates is similar, providing information whether or not the

Table 7-2: Listing of properties as used for the optimization with Random Search.

Property	Value
Stopping Criterion	Sample Size, i.e. 4,450, 17,800, 63,000
Sample Size:	4,450, 17,800, 63,000
Launch Date Window:	360
Time of Flight Interval:	1,000 days
Maximum Number of Gravity Assists	2 (i.e. 3 mission segments)

Table 7-3: Summary of the optimization outcome for a random search and 4,450 population members. Mission Δ*v* in m/s, Launch Date in MJD, Time of Flight in days.

Parameter/No. of legs	1 (0 GA)	2 (1 GA)	3 (2 GA)
$\Delta v_{mission}$ [average/standard deviation]	16,247.69/ 67.21	16,052.75/ 825.18	35,804.98/ 10405.42
$LD_{mission}$ [av./standard dev.]	56333/ 80.47	56203.8/ 250.4	56245.8/ 101.6
$ToF_{mission}$ [av./standard dev.]	2,497.6/ 175	3,078.6/ 250.43	3,193.4/ 115.7
$N_{rev,mission}$ [av./standard dev.]	1/ 0	1.9/ 0.6	1.75/ 0.9
1st GA Partner [# of times occurring]	-	Earth: 5, Mars: 4, Jupiter: 1	Venus: 3, Earth: 5, Mars: 2
GA Encounter Date [av./ standard dev.]	-	56856.9/ 464	59439.1/ 111
$ToF_{1st\ leg}$ [av./ standard dev.]	-	653.1/ 432.4	1,132.6/ 1059.1
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	1/ 0.5	0.9/ 0.6
2nd GA Partner [# of times occurring]	-	-	Mars: 2, Jupiter: 8
GA Encounter Date [av./ standard dev.]	-	-	61096.8/ 713.3
$ToF_{1st\ leg}$ [av./ standard dev.]	-	-	1,657.6/ 701.1
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	-	0.8/ 0.5

Table 7-4: Summary of the optimization outcome for a random search and 17,800 population members. Mission Δ*v* in m/s, Launch Date in MJD, Time of Flight in days.

Parameter/No. of legs	1 (0 GA)	2 (1 GA)	3 (2 GA)
$\Delta v_{mission}$ [average/standard deviation]	16,258.42/ 2.0	14,971.52/ 930.02	38,989.93/ 14449.26
$LD_{mission}$ [av./standard dev.]	56359.9/ 0.3	56187.6/ 113.9	56292.9/ 71
$ToF_{mission}$ [av./standard dev.]	2,518.4/ 15.8	3,156.4/ 232.4	3,349.1/ 127.4
$N_{rev,mission}$ [av./standard dev.]	1/ 0	1.6/ 0.7	1.9/ 1
1st GA Partner [# of times occurring]	-	Earth: 6, Mars: 3, Jupiter: 1	Earth: 4, Mars: 3, Jupiter: 3
GA Encounter Date [av./ standard dev.]	-	56773.3/ 580.9	57395.9/ 774.6
$ToF_{1st\ leg}$ [av./ standard dev.]	-	585.7/ 514	1,103/ 785.6
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	0.8/ 0.4	0.9/ 0.7
2nd GA Partner [# of times occurring]	-	-	Earth: 2, Mars: 2, Jupiter: 6
GA Encounter Date [av./ standard dev.]	-	-	58620.1/ 941.1
$ToF_{1st\ leg}$ [av./ standard dev.]	-	-	1,224.3/ 884.6
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	-	0.6/ 0.7

Table 7-5: Summary of the optimization outcome for a random search and 63,000 population members. Mission Δ*v* in m/s, Launch Date in MJD, Time of Flight in days.

Parameter/No. of legs	1 (0 GA)	2 (1 GA)	3 (2 GA)
$\Delta v_{mission}$ [average/standard deviation]	16,257.93/ 0.97	13,736.60/ 572.62	26,208.49/ 5755.32
$LD_{mission}$ [av./standard dev.]	56360/ 0	56190.8/ 131.6	56215/ 134
$ToF_{mission}$ [av./standard dev.]	2,518.9/ 14.13	2,979.1/ 310	3,297.3/ 117
$N_{rev,mission}$ [av./standard dev.]	1/ 0	1.5/ 0.5	2.13/ 1.03
1st GA Partner [# of times occurring]	-	Earth: 10	Earth: 8, Mars: 1, Jupiter: 1
GA Encounter Date [av./ standard dev.]	-	56499.2/ 194.1	57284.1/ 930.7
$ToF_{1st\ leg}$ [av./ standard dev.]	-	308.4/ 182.4	1,069.1/ 982.5
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	0.6/ 0.7	0.9/ 0.9
2nd GA Partner [# of times occurring]	-	-	Earth: 2, Mars: 4, Jupiter: 4
GA Encounter Date [av./ standard dev.]	-	-	58612.6/ 1369.5
$ToF_{1st\ leg}$ [av./ standard dev.]	-	-	1,328.5/ 750.9
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	-	0.9/ 0.6

evolutionary approach improves results compared to Random Search.

7.3.2 Results

The average outcome of the calculations regarding mission Δ*v* is listed in Table 7-3 to Table 7-5 and is visualized in Figure 7-1. The results vary little for the 1-leg calculations for the different sample sizes and are between ca. 16,247 and 16,258 m/s. The standard deviation for 4,450 is 67.2, for 17,800 it is 2 and for 63,000 it is about 1 m/s. For all cases the Δ*v* obtained with 17,800 samples and for 2 legs is below the 1-leg solutions, decreasing with increasing sample size. The lowest Δ*v* obtained with Random Search also occurs for 17,800 samples and is 12.76 km/s.

Figure 7-2 shows the best result of all random search calculations, which occurred for 17,800 samples and is a single-gravity-assist mission with a mission Δ*v* of 12.76 km/s. Next to it is the respective profile of the thrust acceleration. Clearly visible is the difference of thrust acceleration magnitude between the two segments.

In Table 7-3 to Table 7-5 it is shown that for single leg missions and for an increasing sample size, the standard deviation of other mission parameters besides mission Δ*v* also drops, although the difference between 17,800 and 63,000 is marginal. For 2 and 3 legs, this can only be observed for some of the parameters. For the average flight time and average encounter date of the gravity-assist partner there is a drop with increasing sample size. The standard deviation first increases from 4,450 to 17,800 but drops significantly for 63,000 samples.

Also the diversity of the gravity-assist partner for 2 leg missions decreases until in all cases Earth has been found as the gravity-assist partner. It should be noted that in two cases for 17,800 and in 8 cases for 63,000 samples, the encounter dates are very similar, between 55440 and 55501, which is illustrated in Figure 7-3, showing the clustering of solutions for these dates. Figure 7-4 shows the same for 2-gravity-assist missions, depicting the closest clustering for 63,000 samples.

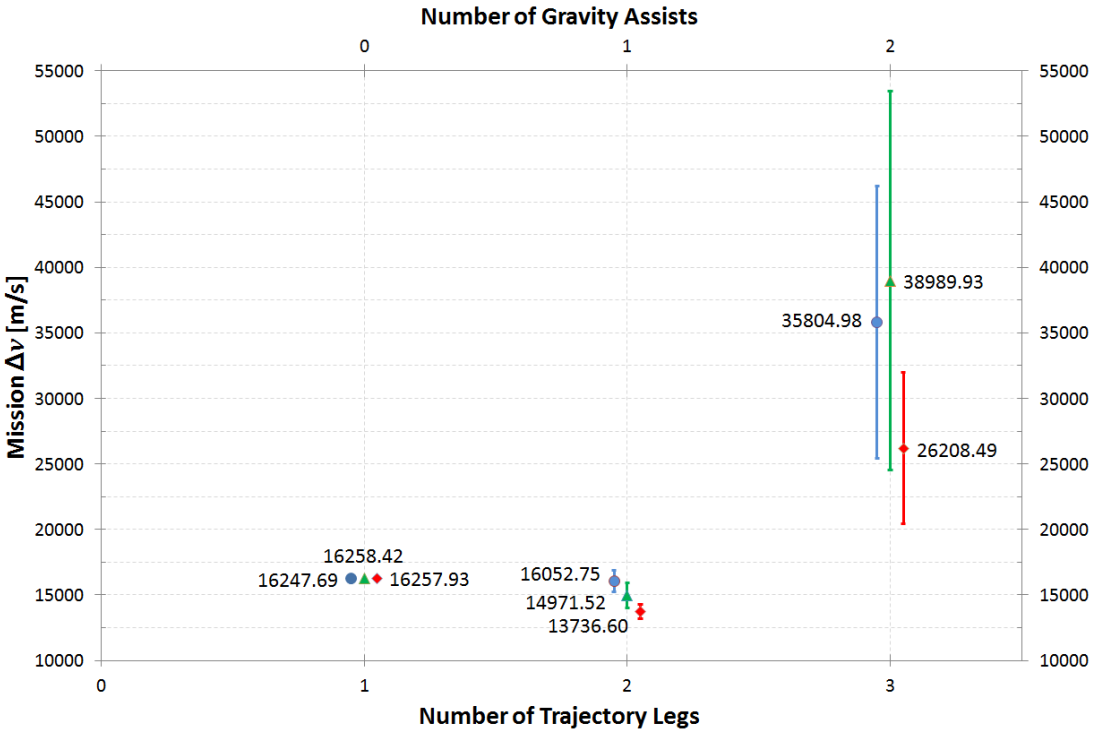

Figure 7-1: Average optimization results (mission Δ*v*) for random search in dependence on number of mission segments and population size. Blue (circle) depicts results for 4,450, green (triangle) for 17,800 and red (triangle) for 63,000 population members. The bars show the standard deviation of the solutions.

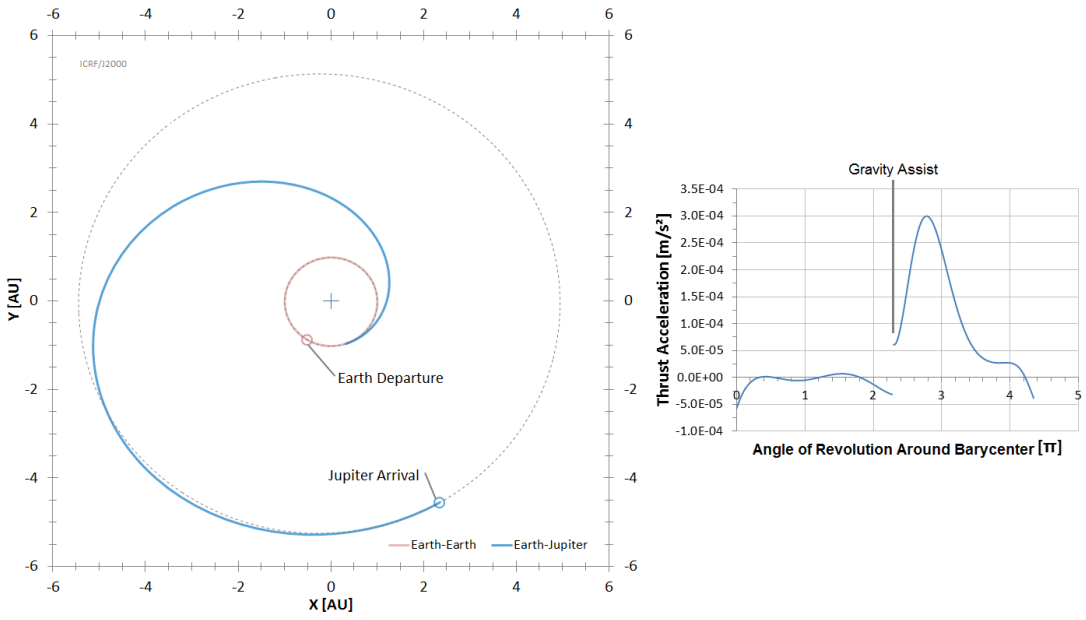

Figure 7-2: The best solution found for Random Search, which occurred for 17,800 samples. A mission from Earth to Jupiter with a single gravity-assist at Earth and a total mission Δ*v* of 12.76 km/s. The coordinates are in the International Celestial Reference Frame and scaled in Astronomical Units (AU). On the right side is the associated thrust history over mission angle in multiples of π.

For 3 legs, the standard deviation values are altering in their trend, sometimes increasing, sometimes decreasing between the different sample sizes, e.g. for the average encounter date of the first gravity-assist partner (110.95, 774.6 and 930.71 resp. for 4,450, 17,800 and 63,000 samples). In seven cases the standard deviation of the two larger sample sizes exceeds that of the smaller sample size. Similarly, the average values also do not show a clear pattern associated with an increase of sample size.

7.3.3 Discussion

It can be seen that the average mission Δv is improving with an increase of the number of samples. The standard deviation is small for single leg missions, but is significant for gravity-assist missions. It is clear that for the limited possible solutions the results improve for larger sample sizes. Therefore, the standard deviation between individual runs of calculations decreases with increasing population size for the single leg solutions. For 63,000 samples taken, the improvement is on average 15.5% comparing the single leg solutions with the 2 leg solutions. While the multi-gravity-assist mission outcome is in no case better than the non-gravity-assist trajectory, the order of magnitude is similar.

The fact that for an increase of samples the standard deviation also increases for multi-leg missions can possibly be attributed to the increased intrinsic variety. As the search is completely random, the solutions are not bound to end up in similar regions of the search space. Therefore, the results vary significantly and more so with larger sample size. The number of variables also increases with increasing leg numbers, which further decreases the chance of recreating a similar solution twice or more often.

For 63,000 samples of single gravity-assist missions, the search produced in 8 cases out of 10, similar gravity assists, i.e. based on an encounter date between 56440 and 56501 and with an Earth flyby. Even for 17,800 samples this encounter was provided twice as result. In difference to that, the gravity-assist properties for multi-gravity-assist missions are more random. The results obtained with 4,450 samples are distinguished clearly in their pattern from the other two search regimes due to later encounter dates, but for all three a pattern in phasing (shown in gradient) is visible in Figure 7-4. The encounter date's standard deviation increases with larger sample sizes and is largest for 63,000. This can be attributed to the fact that the amount of the variables increases for increasing numbers of gravity assists, and they are real valued, i.e. there is an infinite amount of new possible variations. The limited number of samples cannot produce reliable, repeating results in that case. Based on the law of large numbers, it can be said that the used sample size is not sufficient.

However, it should also be noted that the computation effort of randomly selecting 63,000 samples is smaller than for a more sophisticated search, which contains a prescription, resp. a pattern on how new solutions are created out of existing ones.

The thrust acceleration profile of the best solution obtained with Random Search, on the right side of Figure 7-2, shows that thrust is hardly used for the first leg. The spacecraft only slightly adapts the trajectory to obtain an actual flyby. After the gravity assist, it increases the thrust acceleration measurably to arrive at Jupiter, slightly exceeding Jupiter's orbital distance and thus breaking further to actually achieve a rendezvous at the desired point in time. Thrust profiles are further discussed in the following chapters.

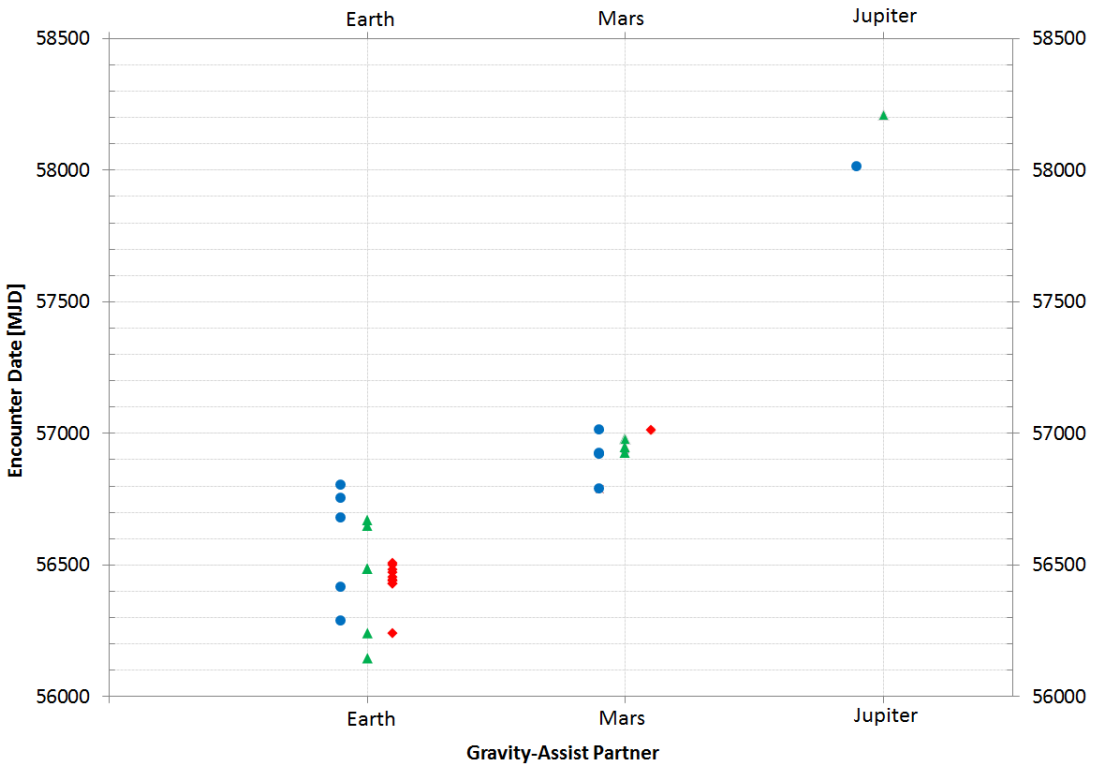

Figure 7-3: Single gravity-assist encounter dates for Random Search for all calculations over gravity-assist partner. Blue (circle) depicts results for 4,450, green (triangle) for 17,800 and red (triangle) for 63,000 population members.

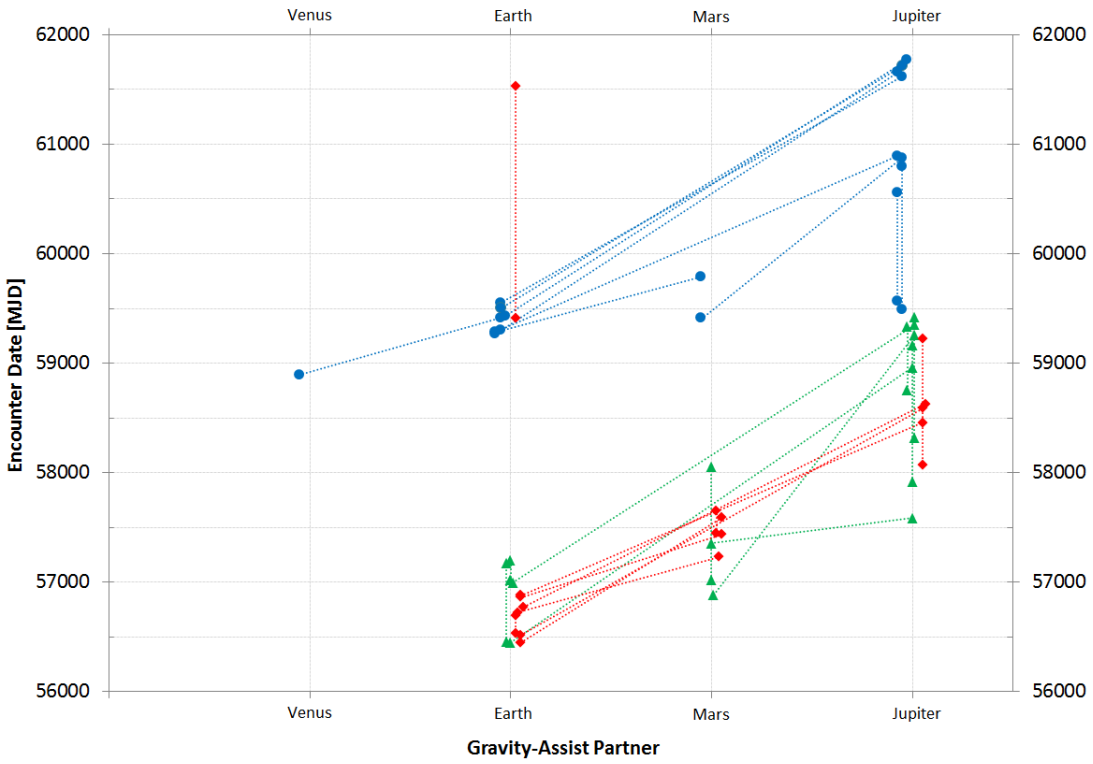

Figure 7-4: Gravity-assist encounter dates of 2-gravity-assist missions for Random Search for all calculations over gravity-assist partner. Blue (circle) depicts results for 4,450, green (triangle) for 17,800 and red (triangle) for 63,000 population members.

Concluding, it can be said that for a 2-leg mission Random Search can provide information about a valuable gravity assist if run repeatedly. This is not the case for multi-gravity-assist missions, where the outcome is less clear.

7.4 Optimization with Differential Evolution

Due to its robustness (see Chapter 5.1.8) Differential Evolution has been selected as a first example of an evolutionary algorithm. Its ability to use the global control variables for purposefully improving a solution to an optimum, although the used variables are only a selection of all control variables is investigated. During these calculations tournament selection is used, as designed by Storn and Price (Storn and Price, 1996), see section 5.1.5. Preliminary results and the principle of the search have been published in (Maiwald, 2017a) and a selection of final results in (Maiwald, 2017b). Similar as before, a rendezvous from Earth to Jupiter with initial and final $v_\infty = 0$ has been calculated.

7.4.1 Optimization Settings

To investigate the impact of the population size on the optimization quality, a fourty calculations have been conducted. The aim has been to review the search quality, expressed by Δv, with increasing numbers of mission segments (i.e. gravity assists). The algorithm properties used for the first batch of calculations are listed in Table 7-6.

The optimized mission has been a transfer from Earth to Jupiter, with a minimum launch date of 56000 Modified Julian Date (MJD). For each combination of number of gravity assists (from 0 to 2) and population size ten calculations have been conducted. After identification of useful solution candidates, the calculations were focused on a smaller window. This way it is investigated if using

Table 7-6: Listing of properties as used for the optimization with Differential Evolution in the first set of calculations.

Property	Value
Stopping Criterion	Max. number of generations (1,000) or 5 times in a row without replacement
Population Size:	50, 200 and 500
Crossover Constant	0.75
Differential Weight	1
Launch Date Window:	360
Time of Flight Interval:	1,000 days
Maximum Number of Gravity Assists	2 (i.e. 3 mission segments)

Table 7-7: Listing of properties as used for the optimization with Differential Evolution in the focused set of calculations.

Property	Value
Stopping Criterion	Max. number of generations (1,000) or 5 times in a row without replacement
Population Size:	500
Crossover Constant	0.75
Differential Weight	1
Launch Date Window:	210
Time of Flight Interval:	190 days (derived from the standard deviation of the previous results)
Maximum Number of Gravity Assists	1 (it was attempted to reproduce and improve the 2 leg solution)

more specific constraints the identification of the optimum can be improved. Therefore, the properties as listed in Table 7-7 have been used, based on the results of the previous calculation set, to reproduce and improve the one-gravity assist result.

7.4.2 Results

The results are listed in Table 7-8 to Table 7-10 for each respective population size. Figure 7-5 and Figure 7-6 show two examples of gravity-assist missions as optimized using the described method. Figure 7-5 is the best solution found in the first runs. Figure 7-6 is an example of a multi-gravity-assist mission performing worse than the non-gravity assist benchmark. For both results Earth is

Table 7-8: Summary of the optimization outcome for Differential Evolution and 50 population members. Mission Δ*v* in m/s, Launch Date in MJD, Time of Flight in days.

Parameter/No. of legs	1 (0 GA)	2 (1 GA)	3 (2 GA)
$\Delta v_{mission}$ [average/standard deviation]	16,256.56/ 0	14,717.19/ 1203.64	29,921.28/ 8652.57
$LD_{mission}$ [av./standard dev.]	56360/ 0	56100.5/ 80.75	56221.4/ 88.41
$ToF_{mission}$ [av./standard dev.]	2,500/ 0	3,032.2/ 219.26	3,161.2/ 310.16
$N_{rev,mission}$ [av./standard dev.]	1/ 0	1.6/ 0.7	2/ 0.82
1st GA Partner [# of times occurring]	-	Earth: 9, Mars: 1	Earth: 7, Jupiter: 3
GA Encounter Date [av./ standard dev.]	-	56534.7/ 97.91	57163.3/ 742.67
$ToF_{1st\ leg}$ [av./ standard dev.]	-	434.2/ 73.6	941.9/ 705.95
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	1.1/ 0.32	1.2/ 0.63
2nd GA Partner [# of times occurring]	-	-	Earth: 3, Mars: 4, Jupiter: 3
GA Encounter Date [av./ standard dev.]	-	-	57985.8/ 931.62
$ToF_{1st\ leg}$ [av./ standard dev.]	-	-	822.5/ 304.8
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	-	0.7/ 0.48

Table 7-9: Summary of the optimization outcome for Differential Evolution and 200 population members. Mission Δ*v* in m/s, Launch Date in MJD, Time of Flight in days.

Parameter/No. of legs	1 (0 GA)	2 (1 GA)	3 (2 GA)
$\Delta v_{mission}$ [average/standard deviation]	16,256.56/ 0	13,185.74/ 516.45	21,528.69/ 4032.75
$LD_{mission}$ [av./standard dev.]	56360/ 0	56059.5/ 43.51	56166.10/ 116.69
$ToF_{mission}$ [av./standard dev.]	2,500/ 0	3,066.3/ 235.15	3,277.10/ 156.88
$N_{rev,mission}$ [av./standard dev.]	1/ 0	2/ 0	2.9/ 0.57
1st GA Partner [# of times occurring]	-	Earth: 10	Earth: 7, Mars: 3
GA Encounter Date [av./ standard dev.]	-	56480.7/ 21.84	56750.2/ 250.08
$ToF_{1st\ leg}$ [av./ standard dev.]	-	421.2/ 38.79	584.10/ 244.89
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	1/ 0	1.0/ 0.47
2nd GA Partner [# of times occurring]	-	-	Earth: 2, Mars: 8
GA Encounter Date [av./ standard dev.]	-	-	57372.60/ 259.4
$ToF_{1st\ leg}$ [av./ standard dev.]	-	-	622.40/ 50.63
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	-	1/ 0

Table 7-10: Summary of the optimization outcome for Differential Evolution and 500 population members. Mission Δ*v* in m/s, Launch Date in MJD, Time of Flight in days.

Parameter/No. of legs	1 (0 GA)	2 (1 GA)	3 (2 GA)
$\Delta v_{mission}$ [average/standard deviation]	16,256.56/ 0	12,776.56/ 240.41	18,954.22/ 1621.94
$LD_{mission}$ [av./standard dev.]	56360/ 0	56112.1/ 76.47	56205.1/ 87.65
$ToF_{mission}$ [av./standard dev.]	2,500/ 0	2,968.4/ 178.07	3,272.0/ 132.68
$N_{rev,mission}$ [av./standard dev.]	1/ 0	1.9/ 0.32	2.5/ 0.71
1st GA Partner [# of times occurring]	-	Earth: 10	Earth: 7, Mars: 3
GA Encounter Date [av./ standard dev.]	-	56487.3/ 19.65	56703.8/ 200.22
$ToF_{1st\ leg}$ [av./ standard dev.]	-	375.2/ 76.36	498.7/ 149.4
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	1/ 0	0.9/ 0
2nd GA Partner [# of times occurring]	-	-	Earth: 4, Mars: 5, Jupiter: 1
GA Encounter Date [av./ standard dev.]	-	-	57388.3/ 513.39
$ToF_{1st\ leg}$ [av./ standard dev.]	-	-	684.5/ 430.22
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	-	0.9/ 0.32

the gravity-assist partner selected by the optimizer.

Depending on the population size, the average values for the mission Δv differ, as well as the number of generations used for the optimization. For a population size of 50, in 90% of the calculations for a single gravity assist (two segments) the results show a smaller Δv than the non-gravity-assist (one segment) benchmark mission, which has a Δv result of about 16.3 km/s. For population sizes of 200 and 500 the results for a single gravity-assist (two segments) mission are always better than the non-gravity assist solutions, with averages of about 13.2 resp. 12.8 km/s.

For missions containing two gravity assists (three segments) only for one calculation the result has been better than the non-gravity assist benchmark. This occurred for 500 population members and resulted in a Δ*v* of 16,212.3 m/s, an improvement of about 44 m/s. The results for multi-gravity assist calculations average on 29, 21.5 and 18.9 km/s for 50, 200 and 500 population members.

The non-gravity assist results are always identical with a Δv of 16,256 m/s as can also be seen in Figure 7-7. This has not been the case for multi-segment missions. The same figure and respective tables show the standard deviation for all other cases. Single-gravity-assist missions perform in average better (smaller mission Δv) than non-gravity-assist missions. This is not the case for multi-gravity-assist missions. However, in all cases an increase of population size also improves the solution quality and in most cases also reduces the solution standard deviation.

The number of generations also varies depending on population size and number of mission segments. It is almost constant for all single segment missions, between 129.8 (for 50 population members) and 134.7 (for 500 population members). For 50 population members, the number of generations decreases slightly in case of two and three segment missions (86, resp. 93.6 generations). The number of generations increases with increasing population sizes, as does the standard deviation, and decreases for multi-gravity-assist missions (three segments). Table 7-8 to Table 7-10 summarize the results of all calculations in average values and standard deviation for missions containing no gravity assist, a single gravity assist and two gravity assists in dependence of the population members (50, 200 and 500).

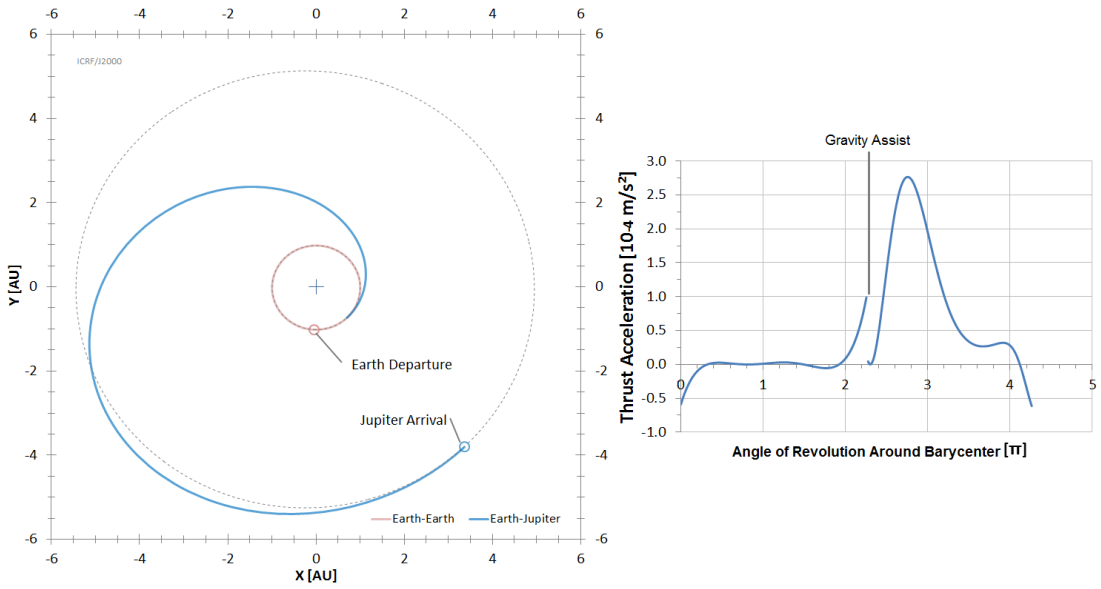

Figure 7-5: An example for a two-segment mission from Earth to Jupiter, involving a single gravity assist at Earth, with a total mission Δv of 12.47 km/s. The coordinates are in the International Celestial Reference Frame and scaled in Astronomical Units (AU). On the right side the associated thrust history over mission angle in multiples of π.

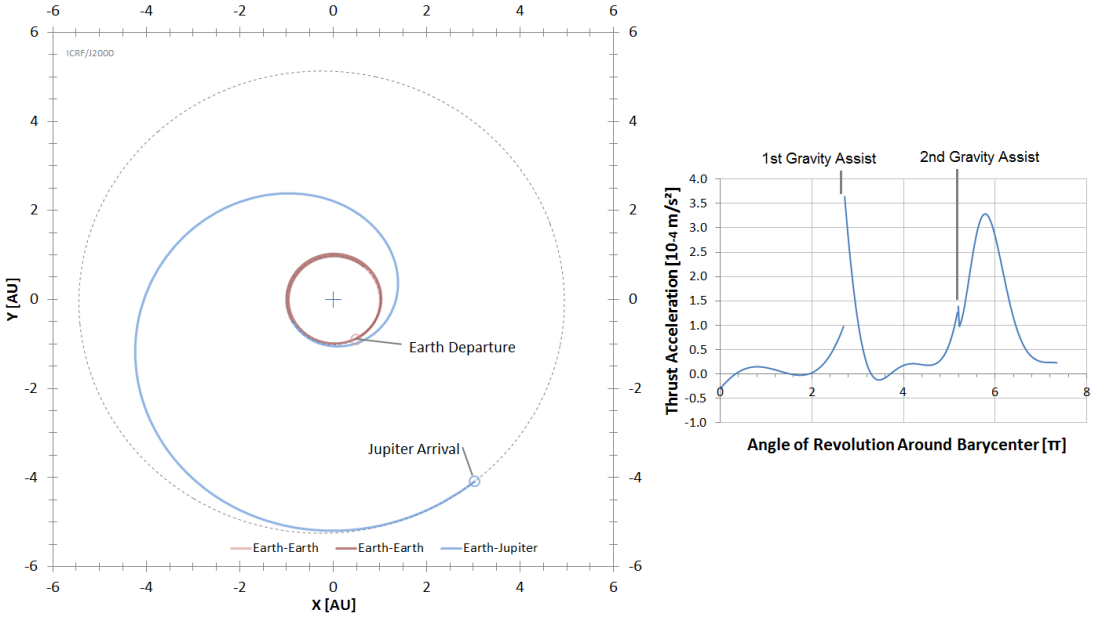

Figure 7-6: An example for a three-segment mission from Earth to Jupiter, involving two gravity assists at Earth, with a total mission Δv of 18.5 km/s. The coordinates are in the International Celestial Reference Frame and scaled in Astronomical Units (AU). On the right side the associated thrust history over mission angle in multiples of π.

It can be seen that the standard deviation decreases with increasing population members and in general the mission scenarios become less different. This is especially obvious for the encounter date of the single gravity-assist missions. The standard deviation decreases and a specific encounter is more specified. This is visible in Figure 7-9, where clear clusters of encounter dates can be seen for single-gravity-assist missions. Less clear and more distributed, similar sequences can be seen in Figure 7-10, where similar gradients are visible for several solutions.

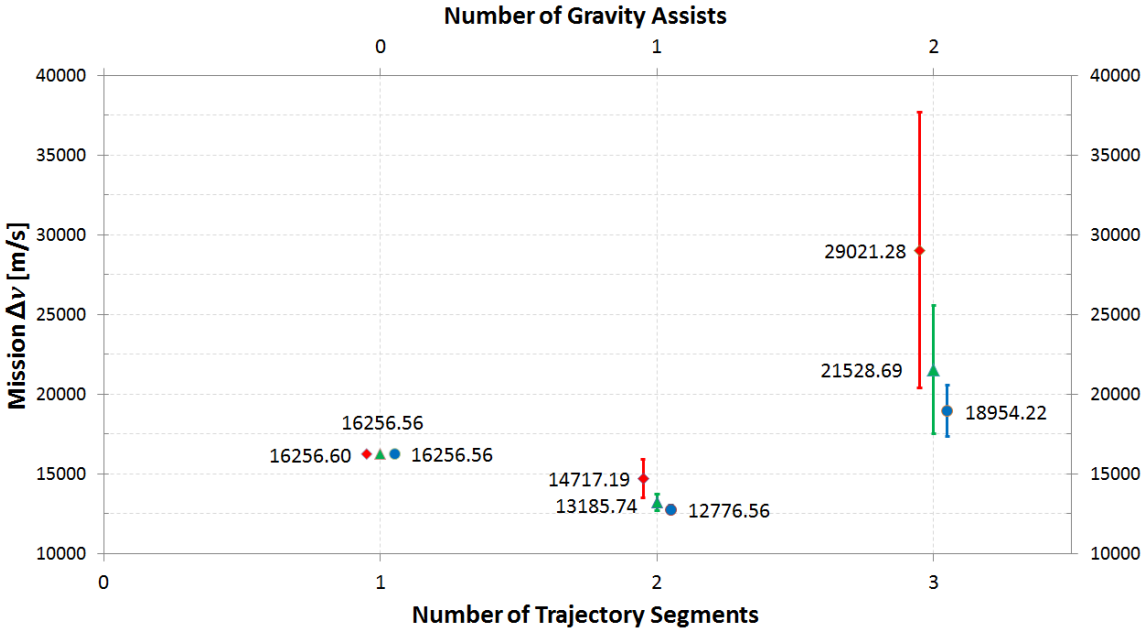

Figure 7-7: Average Differential Evolution results (mission Δ*v*) in dependence on number of mission segments and population size. Red (diamond) depicts results for 50, green (triangle) for 200 and blue (circle) for 500 population members. The bars show the standard deviation of the solutions.

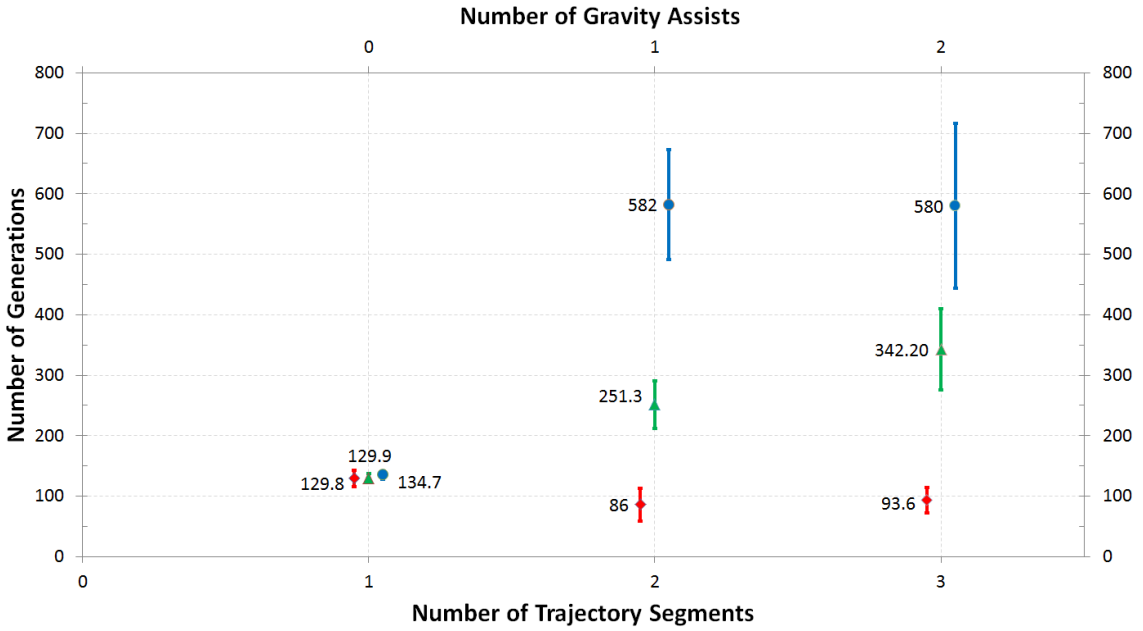

Figure 7-8: Average number of generations used for optimization with Differential Evolution in dependence on mission segments and population size. Red (diamond) depicts results for 50, green (triangle) for 200 and blue (circle) for 500 population members. The bars show the standard deviation of the average number.

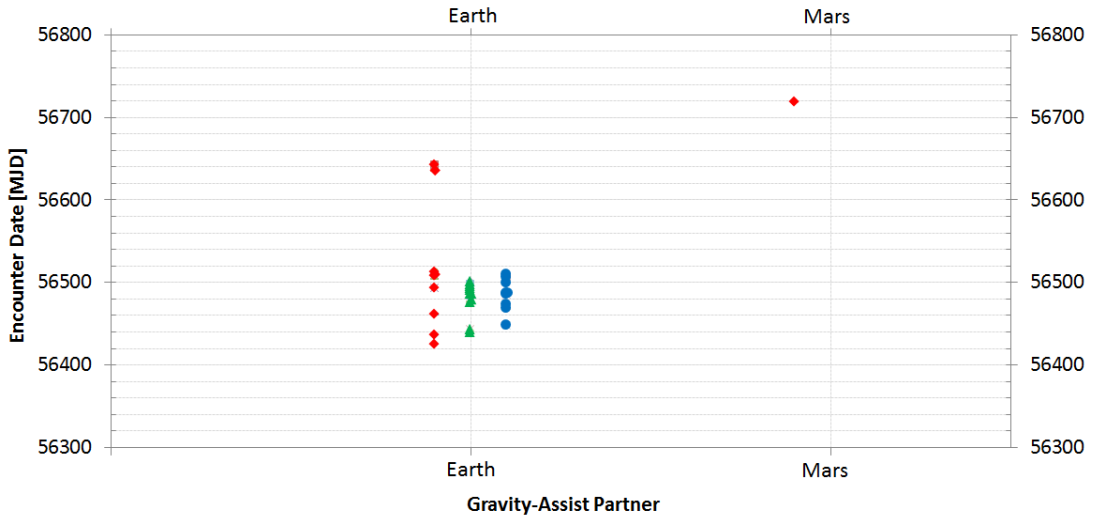

Figure 7-9: Single gravity-assist encounter dates for Differential Evolution for all calculations over gravity-assist partner. Red (diamond) depicts results for 50, green (triangle) for 200 and blue (circle) for 500 population members.

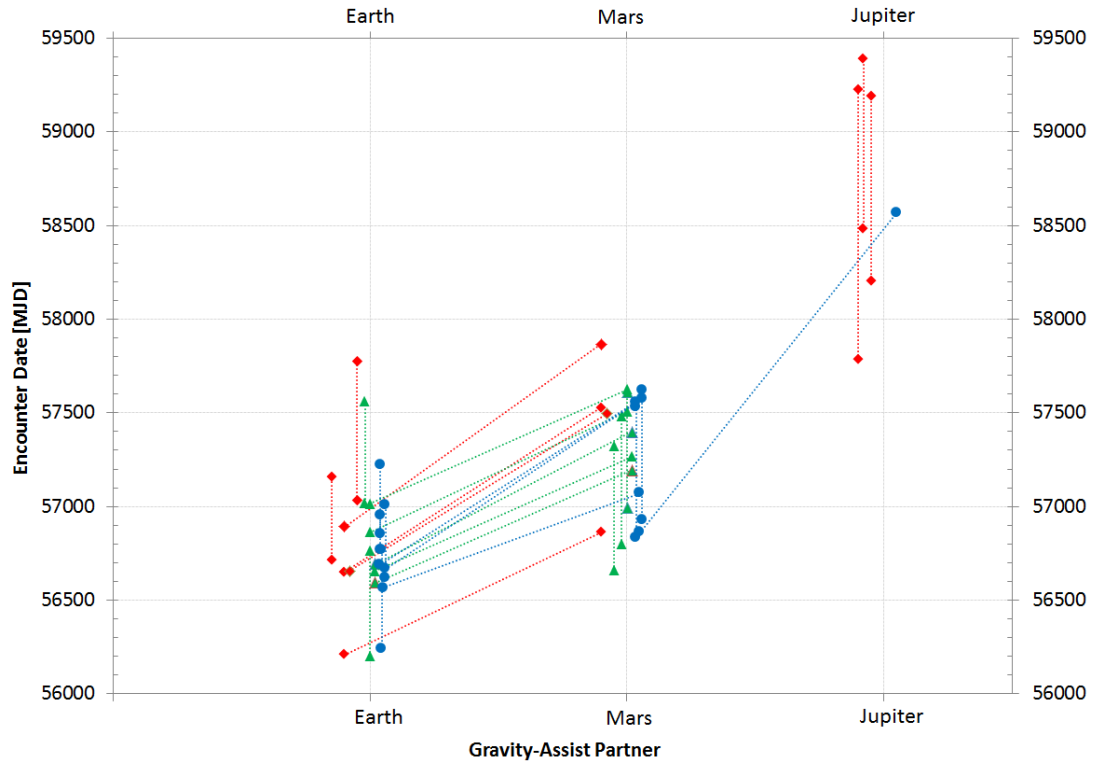

Figure 7-10: Gravity-assist encounter dates for 2 gravity-assist missions obtained by Differential Evolution over gravity-assist partner. Red (diamond) depicts results for 50, green (triangle) for 200 and blue (circle) for 500 population members.

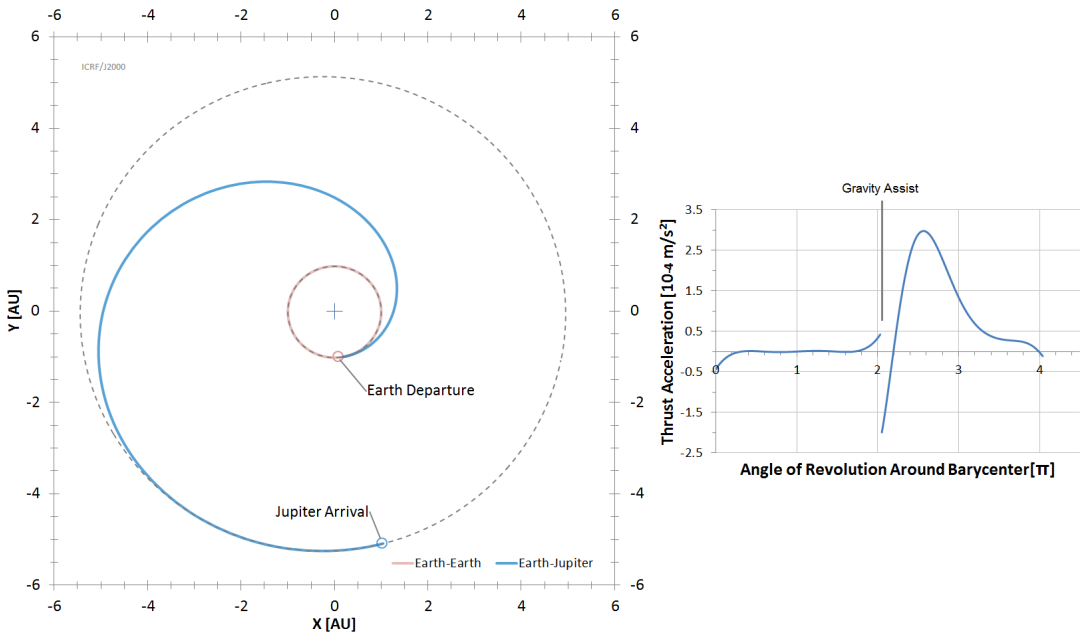

Figure 7-11: The best solution found in the focused calculations, a two segment mission from Earth to Jupiter, involving a single gravity-assist at Earth, with a total mission Δ*v* of 12.1 km/s. The coordinates are in the International Celestial Reference Frame and scaled in Astronomical Units (AU). On the right side the associated thrust history over mission angle in multiples of π.

Table 7-11 provides the results for the focused calculations attempting to reproduce and improve the previously found solutions for single gravity-assist missions (two segments). It can be seen that the average mission Δ*v* is about 100 m/s lower than the previous average result for 2-leg-missions (note: the average result for 1-leg-missions is about 60 m/s larger, this is caused by the fact that the previously found optimal launch date was outside the window of the focused calculations). In addition, the best result was further improved to 12.1 km/s. Its trajectory can be seen in Figure 7-11 along with its thrust acceleration profile.

The standard deviation of the average mission flight time has been reduced to almost one third of the original value, the first leg's flight time standard deviation has halved. The encounter date of the gravity assist is almost identical to the previously found and only differs by three days; the standard deviation has dropped to 75% of the previous value. The average number of generations has dropped to 519, whereas the standard deviation here increased. The launch date of the mission decreased, as did its standard deviation.

7.4.3 Discussion

Figure 7-7 shows that the method using Differential Evolution has been successful in improving missions with two segments in comparison with single segment missions without gravity assist. On average, even for 50 population members, the *Δv* is smaller than for the non-gravity-assist mission. In the best case, depicted in Figure 7-5, this improvement has reached 24%, i.e. 3.87 km/s better than the no-gravity assist solution. This can be considered a relevant improvement; even more so as respective propellant savings are exponential and not linear.

The solution quality in case of multi-gravity-assist missions (3 segments) decreases in comparison to single- and no-gravity assist solutions. The respective *Δv* is a factor 1.97 larger for 50 population members, 1.63 for 200 and 1.48 for 500 population members compared to the single-gravity assist.

The reduced quality for multi-gravity assist solutions can be attributed to the increased number of variables (with real values, i.e. infinite possibilities) in combination with the fact that these variables are not evolved, but reinitiated for every solution candidate. With an increased number of variables, it is more unlikely to find combinations for the variables, which result in solution improvement compared to no-gravity assist or single-gravity-assist missions. The areas of optimal solutions become smaller relatively to the overall search space for larger segment numbers.

This fits the change in numbers of generations used for the optimization, see Figure 7-8. With an increasing population size, the generation number increases only slightly for single segmented mission. In the given set of variables and constraints (see Chapter 7.4.1) for the single segment mission the number of variations is finite, as defined by $N_{\text{rev,mission}}$, the ToF_{mission} and the LD_{mission}.

All of these are integer values, e.g. only whole days have been considered (it should be noted that a mission constraint setting a flyby velocity for the beginning and/ or final planet would add a real value for single leg missions as well). The number of different possible values, k, is then:

$$k = N_{rev,max} \cdot LD_{window} \cdot ToF_{window}, \tag{7-3}$$

with $N_{rev,max}$ as maximum number for the number of revolutions (varied from 0 to that value during optimization), LD_{window} as interval for the Launch Date and ToF_{window} the interval of the flight time.

Adding one or more gravity assists adds more variables, but real valued variables, i.e. adding even a single gravity assist immediately enhances the number of possible solutions candidates to infinity. For optimizations with a population size of 50, the probability to find solution improvement is smaller compared to a population size of 200, or 500 and thus the respective results are of less quality (Figure 7-7) and the number of generations taken is also smaller with decreasing population sizes (Figure 7-8).

The maximum allowed number of generations has been 1,000 and the optimization stops once no improvement is achieved for 5 generations in a row. A number of 1,000 generations has not been used up once in the trial runs. Figure 7-8 shows that for a larger population size more variations can be evaluated, therefore the optimization runs for more generations.

For similar reasons, the standard deviation of the solutions for 2 and 3 segments is increasing from 50 to 200 population members, see Table 7-8 and Table 7-9. The larger number of population members allows more variation in the values of the control variables and therefore more changes

Table 7-11: Summary of the optimization outcome for Differential Evolution and 500 population members and the focused mission constraints. Mission Δ*v* in m/s, Launch Date in MJD, Time of Flight in days.

Parameter/No. of legs	1 (0 GA)	2 (1 GA)
$\Delta v_{\text{mission}}$ [average/standard deviation]	16,318.29/ 67.21	12,641.91/ 343.29
# of Generations [av./standard dev.]	80.4/ 2.17	519.7/ 116.11
LD_{mission} [av./standard dev.]	56068/ 0	56095/ 37.78
ToF_{mission} [av./standard dev.]	2,935/ 0	2,842.6/ 67.28
$N_{\text{rev,mission}}$ [av./standard dev.]	2/ 0	2/ 0
1st GA Partner [# of times occurring]	-	Earth: 10
GA Encounter Date [av./ standard dev.]	-	56484.2/ 14.37
$ToF_{\text{1st leg}}$ [av./ standard dev.]	-	389.2/ 31.83
$N_{\text{rev,1st leg}}$ [av./ standard dev.]	-	1/ 0

in the mission outcome.

Besides that, even for 50 population members, a majority of solutions at the end of the optimization is showing similar properties of the gravity assist, i.e. the actual gravity-assist partner and the encounter date. This is even more present for the single-gravity assist solutions calculated with 200 and 500 population members. The standard deviation does not differ significantly between these two sets of calculations. Neither is the outcome regarding Δv. Even though significantly more samples are taken, the improvement is marginal. For the example of single-gravity-assist missions and from 200 to 500 population members, the Δv is only improved by 2.5% compared to the no-gravity assist result. In addition, the difference between the encounter date deviations is marginal, showing that for identification of the optimal gravity assist encounter, the influence of the increase of population size is saturated. Even at 200 population members, the identification is reliable. For multi-gravity-assist missions the identification of valuable partners occurs as well, Earth is most often identified for the first gravity assist in both of the larger population regimes. Although not a single sequence is obtained, the sequences show similarities also in encounter date, as evident in Figure 7-10. The paths are similar for several cases. The encounter dates are spread over several hundred days, but the actual sequences are similar, as can be seen by the clustering for the gravity-assist partners Earth and Mars and similar gradients showing similar time intervals between the respective encounters.

These results are a significant difference to previous methods of optimizing low-thrust gravity assists. It is shown that for this given mission frame, and assuming a single gravity-assist, one certain gravity-assist possibility exists, with a benefit that outweighs the other options' that have been investigated and this gravity-assist possibility is identified by the search method in every optimization attempt, showing a good reliability of the result. To a lesser degree such behavior is also observable for 2-gravity-assist solutions.

Applying Differential Evolution does not yield better multi-gravity assist solutions than single- or no-gravity assist solutions, except for one instance, but the optimization reaches a similar order of magnitude of Δv values as for the other two regimes. This and the fact that evolving only the global variables of the mission leads to better 2-leg missions than 1-leg missions, shows that LD_{mission}, ToF_{mission}, $N_{\text{rev,mission}}$ and the gravity-assist partner(s) have a dominant role in determining the quality of a mission sequence. This dominance decreases with increasing numbers of variables, but still is measurable even for two-gravity assist solutions.

This is further underlined by the fact that for all possible regimes a comparison with the Random Search results shows improved solution quality by application of Differential Evolution. Even for the regime of 63,000 samples the direct comparison with the 200 population number calculations, accounting for about 50,000 samples taken, shows a difference of about 500 m/s in the solution's average Δv, i.e. 13.7 km/s vs 13.2 km/s. This means that the evolutionary character of the search is beneficial, although not all relevant variables are part of the evolutionary process.

In the second batch of calculations the search has been more focused to reproduce and improve the previously optimized solutions. However, even though the reduction of possible variable values has been significant (e.g. reducing the launch date from 1000 days to about 300), no significant solution improvement has been found. On average the mission Δv is very similar to the broader search. While the standard deviation for the number of generations has increased, the average value decreased by about 10%. This is expected as less variation is possible for smaller windows of the variables. On the other hand for all remaining variables the standard deviation decreased, i.e. the solutions were produced with more reliability. The outcome, which is considered as most important, the identification of the encounter date has however improved by only 25%.

Considering the goal of further refining the found solutions, reduction of possible variable variations did not produce a significant advantage. Neither did the mission Δv improve significantly, nor the reliability of the results. Indeed the Δv's standard deviation even increased.

This is attributed to the fact that the dominance of the mission's global variables is not absolute. Further improvement is likely only possible with specific variation of the local variables as well. Due to the goal of finding the gravity-assist sequence, this is not possible, however (see Chapter 5.5). Once the gravity-assist partner is identified, this could partly be helped by setting up a staged search in the following manner:

First, identifying areas, where good solutions are located (e.g. defining the gravity-assist partners, which represents the described method). Second, keeping the gravity-assist partner(s) constant and varying and evolving gravity assist related variables (hyperbolic excess velocity and turning angle) for further refinement of the solutions. This would still not completely exclude the problem concerning interrelated variables, e.g. the flight time of the mission is still the limit of the flight time for all legs, but it would include some more variables in the evolution process. The global properties of the search would be reduced by reducing the number of free variables, but at the same time improve the local search quality. Therefore, a combination might result in improved solution quality.

On the other hand, the standard deviation of the associated mission properties, e.g. Launch Date and Mission Flight Time, decreased. In case of the Launch Date the standard deviation was halved, although the window is about two thirds of the previous one. A reduction to 20% of the Flight Time window resulted also in a standard deviation of that value of about one half. This indicates it is beneficial to use smaller windows for mission parameters and do separate calculations into several windows to find a best solution instead of doing one calculation with a single but large set of parameter windows, which also follows from Eq. (7-3).

The thrust profiles for the two single-gravity-assist missions presented in Figure 7-5 and Figure 7-11 are similar, showing a common strategy. In both cases the spacecraft first thrusts opposite to the flight direction, breaking and thus reducing its orbital energy. Afterwards a near coasting strategy is applied, before the spacecraft increases its orbital energy with the help of a positive thrust and achieving a flyby at Earth. In both cases the spacecraft has a positive thrust gradient after the gravity-assist until its maximum thrust. This increases the spacecraft's solar distance allowing it to close in on the target body Jupiter. The latter parts of the thrust profile show a decrease in the thrust acceleration magnitude until the velocities of spacecraft and planet match. A similar strategy is attempted in the solution shown in Figure 7-11.

Summarizing, the current search method can find an encounter date for the given mission scenario, where using Earth as gravity-assist partner is useful, which is evidenced by the significant Δv improvement. Furthermore, it also shows that exploiting the inner body for the gravity assist leads to better results than using the outer body. This is likely attributed to the smaller orbital periods of the inner body, which make it is easier to find a trajectory allowing an encounter with the inner body once more for a gravity assist than for an outer body.

Not all solutions of the search share similar (if not identical) properties, e.g. regarding flight time. This shows that the search is not reaching the same optimum region for each search run. This could be caused by the large number of variations, but also by lack of keeping up diversity. Increasing solution diversity can be achieved by increasing the population size, which leads to better results, see Figure 7-7.

Although this is the case, the variation of the results (Figure 7-7) for 2-leg missions is not large compared to the overall average Δv value (ca. 4%). Thus it is possible that the optimal solutions do

not vary much in their Δv and therefore the optimizer has problems identifying the globally best solution. The areas of optimal results are large enough to be identified as evidenced by the good results.

In general, applying Differential Evolution allowed finding reliably a gravity-assist partner and improved the mission Δv significantly. While multi-gravity-assist missions were not optimized with the same improvement, it could be shown that the evolutionary search has provided better results than Random Search.

7.5 Optimization with Simulated Annealing

Simulated Annealing is also an algorithm that derives new solutions from previous ones, but it does not optimize in populations but always with a single sample. This is modified in iterations, see Chapter 5.1.3. To test this very different kind of algorithm also a number of calculations have been conducted. It has been investigated how the temperature drop influences the optimization outcome, i.e. for a linear drop Eq. (5-5) has been applied with $\eta = 5 \times 10^{-5}$ and for the exponential drop, Eq. (5-6) with analogously $\alpha = (1 - 5 \times 10^{-5}) = 0.99995$.

Normally, the control variables are varied by using similar factors for all control variables, however the control variables have very different ranges for the optimization problem of this thesis (flight time varies over some 1000 days, whereas the number of revolutions varies typically over single numbers, between 0 and about 5) and some are discrete, e.g. N_{rev}. For each parameter the neighbourhood percentage n_{p} describes now the variation of the value in the following way:

$$n_p = \frac{(\tau + 3\sigma - (\tau - 3\sigma))}{\Delta} \tag{7-4}$$

where τ is the expected mean value for the respective control variable, Δ the possible range and σ the variance. The value n_{p} is a user input and is varied in the course of the test calculations (see next section). With the help of Eq. (7-4) σ can be determined. It is then used via a Gaussian distribution to create the new value for the respective control variable in the following way:

$$X_{i+1} = X_i + \sqrt{-\ln(r_x) \cdot 2\sigma^2} \tag{7-5}$$

The value X is representing the value for the respective control variable for the iteration i resp. i+1 and r_{x} is a randomly created number. It is used in combination with X_{i} to find the value X_{i+1} via the Gaussian distribution curve.

An example for this procedure is given in Figure 7-12 for determining the next iteration's Launch Date based on a randomly created number (vertical axis) and the current iteration's Launch Date (horizontal axis, center). Beginning with e.g. a randomly created number of 0.1 and for a neighbourhood percentage of 30% a new Launch Date of 56054 is created. For integer values (as e.g. the Launch Date), the value is mathematically rounded.

7.5.1 Optimization Settings

The parameters for the search with Simulated Annealing have been similar to the search with Differential Evolution, but not identical and are listed in Table 7-12 to Table 7-15. Four different regimes have been run to test the influence of the numbers of iterations on the optimization outcome (Table 7-12), the influence of the neighborhood percentage (Table 7-13), i.e. in which proximity to the current solution a new solution is created (s. Eq. (7-4), resp. Figure 7-12), the initial temperature T_0 (Table 7-14) that is used as starting point of the iteration (see Eq. (5-5) resp. (5-6)), and the method for dropping the temperature, i.e. either linear or exponential (Table 7-15)

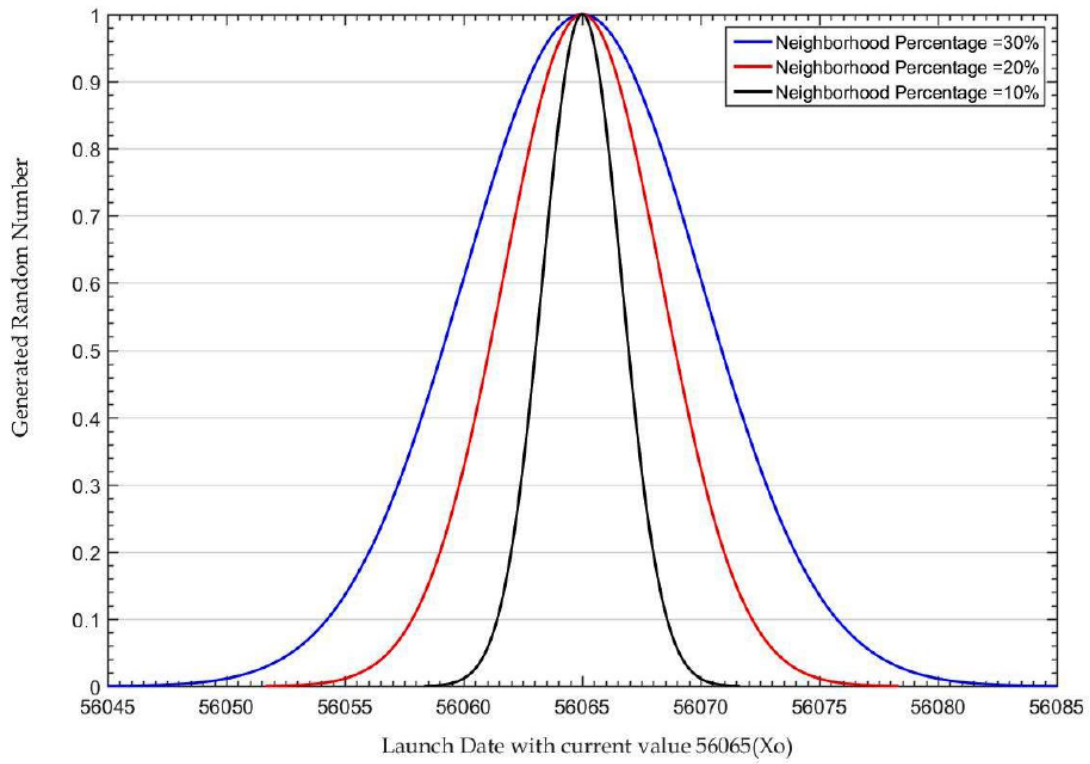

Figure 7-12: A set of Gaussian distribution curves for the current state X_o to determine the next iteration's value for the example of the Launch Date for neighourhood percentage values of 10%, 20% and 30%.

(refer to Eq. (5-5) and (5-6)). The stopping criterion has always been the maximum number of iterations.

The settings for investigating the influence of the number of iterations are listed in Table 7-12, those for investigating the effect of temperature drop method are found in Table 7-13. Reviewing the influence of the initial temperature value has been done with the parameter settings as provided in Table 7-14. The effect of changing the neighborhood percentage has been investigated with parameter values as given in Table 7-15.

7.5.2 Results

The results of the variations of iteration number are summarized in Figure 7-13. It can be seen that there is an increase in Δv for an increase in number of iterations from 100,000 to 200,000 and for 1-leg and 2-legs solutions. For 3-leg solutions the there is a drop. It can also be seen that there is a drop for all cases concerning an increase of the number of iterations from 200,000 to 500,000. With the exception of the 1-leg solutions, the best average solution occurs for the calculations with 500,000 iterations. The best value for 1-leg missions and overall best occurs for 100,000 iterations: 15,916.62 m/s. For 2- and 3-legs missions the standard deviation follows a similar trend, whereas it is also continuously dropping for increasing numbers of iterations in case of the 1-leg solutions.

In no case the algorithm is capable to find a solution using gravity assists, which is better than the no-gravity assist solution. In all cases of the 2-leg calculations Mars has been selected as gravity-assist partner, the selection in the multi-gravity assist calculations have been more varied, but Earth has been the majority of the first gravity-assist partner, i.e. 60%, 80% and 70% of the results for 100,000, 200,000 resp. 500,000 iterations.

Table 7-12: Listing of properties as used for the optimization with Simulated Annealing to investigate the influence of the number of iterations.

Property	Value
# of iterations	100,000, 200,000, 500,000
Temperature:	Exponential drop, starting at 5K
Neighborhood Percentage:	30%
Launch Date Window:	360
Time of Flight Interval:	1,000 days
Hyperbolic Excess Velocity at Start	1,000 m/s
Hyperbolic Excess Velocity at End	50 m/s
Maximum Number of Gravity Assists	2 (i.e. 3 mission segments)

Table 7-13: Listing of properties as used for the optimization with Simulated Annealing to investigate effects of the method of temperature drop (see Eq. (5-5) and Eq. (5-6)).

Property	Value
# of iterations	100,000
Temperature:	Linear and Exponential drop, each starting at 5K
Neighborhood Percentage:	30%
Launch Date Window:	100
Time of Flight Interval:	2,000 days
Hyperbolic Excess Velocity at Start	1,000 m/s
Hyperbolic Excess Velocity at End	50 m/s
Maximum Number of Gravity Assists	2 (i.e. 3 mission segments)

Table 7-14: Listing of properties as used for the optimization with Simulated Annealing to investigate the influence of the initial temperature value (T_0 in Eq. (5-6)).

Property	Value
# of iterations	100,000
Temperature:	Exponential drop, starting at 5K, 10K and 15K
Neighborhood Percentage:	30%
Launch Date Window:	100
Time of Flight Interval:	2,000 days
Hyperbolic Excess Velocity at Start	1,000 m/s
Hyperbolic Excess Velocity at End	50 m/s
Maximum Number of Gravity Assists	2 (i.e. 3 mission segments)

Table 7-15: Listing of properties as used for the optimization with Simulated Annealing to investigate the influence of the value for the neighbourhood percentage (see Eq. (7-4), (7-5) and Figure 7-12).

Property	Value
# of iterations	100,000
Temperature:	Exponential drop, starting at 5K
Neighborhood Percentage:	10%, 20%, 30%
Launch Date Window:	100
Time of Flight Interval:	2,000 days
Hyperbolic Excess Velocity at Start	1,000 m/s
Hyperbolic Excess Velocity at End	50 m/s
Maximum Number of Gravity Assists	2 (i.e. 3 mission segments)

It has to be noted that two solutions for a 3-leg mission and for 100,000 iterations exceeded the other results by two orders of magnitude and have therefore not been considered for the averaging of the results in order not to obscure any possible pattern. Nonetheless, in 20% of the cases this means a failure of the optimization.

The results of the investigation regarding method of temperature drop are depicted in Figure 7-14. It can be seen that for an exponential drop results are significantly better in the 2-leg-mission case, 16.7 km/ s vs. 19.5 km/s, whereas the difference between both drop methods are marginal in case of the 3-leg-missions. Here the results are 32.8 resp. 33.1 km/s. The linear drop is also associated with larger standard deviations, which is more prominent for the 2-leg-mission case. For exponential drop, the standard deviation is 505 m/s and for linear drop it is 3121 m/s.

For three legs again results occurred – for both drop methods – that exceeded the others by two orders of magnitude. This occurred once for the linear drop and twice for the exponential drop. They have not been regarded for the average calculations.

In Figure 7-15 the results for the different starting temperatures are depicted. For 3-leg missions a clear trend can be detected, where the solution quality improves with an increase in starting temperature. This trend is not visible for 2- or 1-leg-missions, where the results are ambiguous. At higher temperatures no results have been found, which exceeded the remaining average results by an order of magnitude or more.

Figure 7-16 illustrates the outcome of the calculations investigating the influence of the neighborhood percentage. Of the three investigated values, 10%, 20% and 30%, the best performing has been 20% for mission involving one or more gravity assist. Worst performing has been 10%, which resulted once even for the 1-leg-solution in a mission Δv almost 2 orders of magnitude larger than the remaining results and for 3-legs-missions only two results were in the

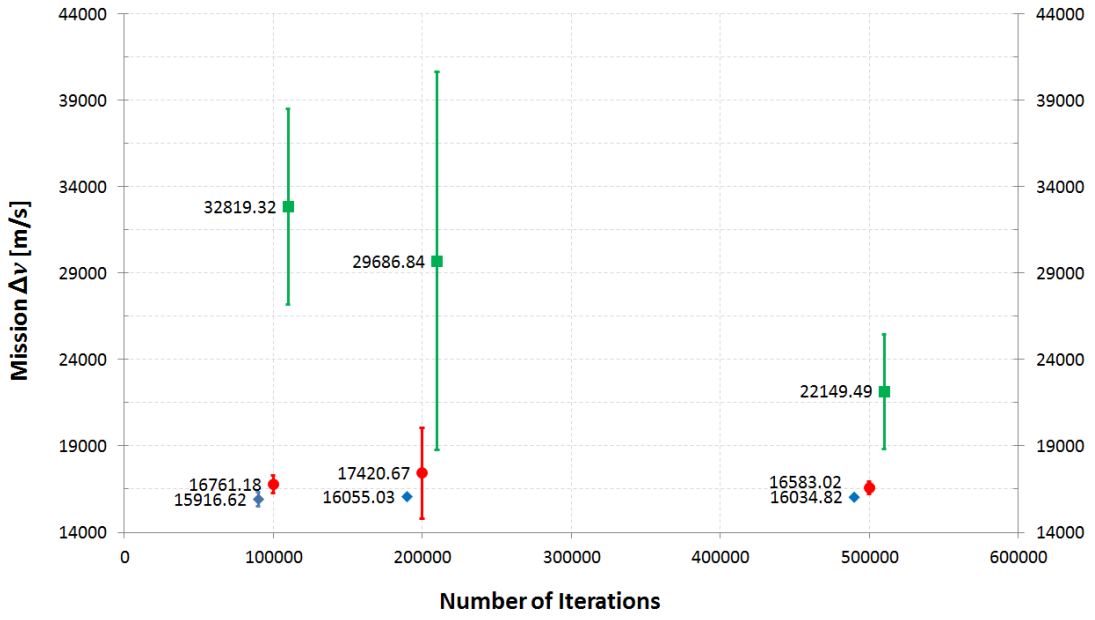

Figure 7-13: Average optimization results (mission Δv) of Simulated Annealing in dependence on number of iterations. Blue (diamond) depicts results for 1 leg, red (circle) for 2 and green (square) 3 legs. The bars show the standard deviation.

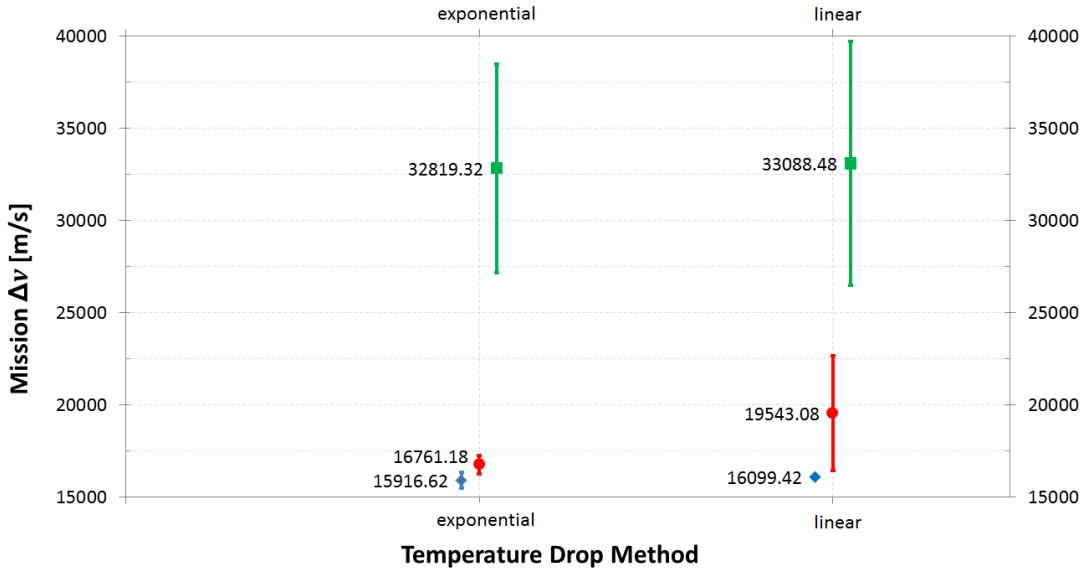

Figure 7-14: Average optimization results (mission Δ*v*) of Simulated Annealing in dependence on method of temperature drop (see Eq. (5-5) and (5-6) and number of segments. Blue (diamond) depicts results for 1 leg, red (circle) for 2 and green (square) 3 legs. The bars show the standard deviation.

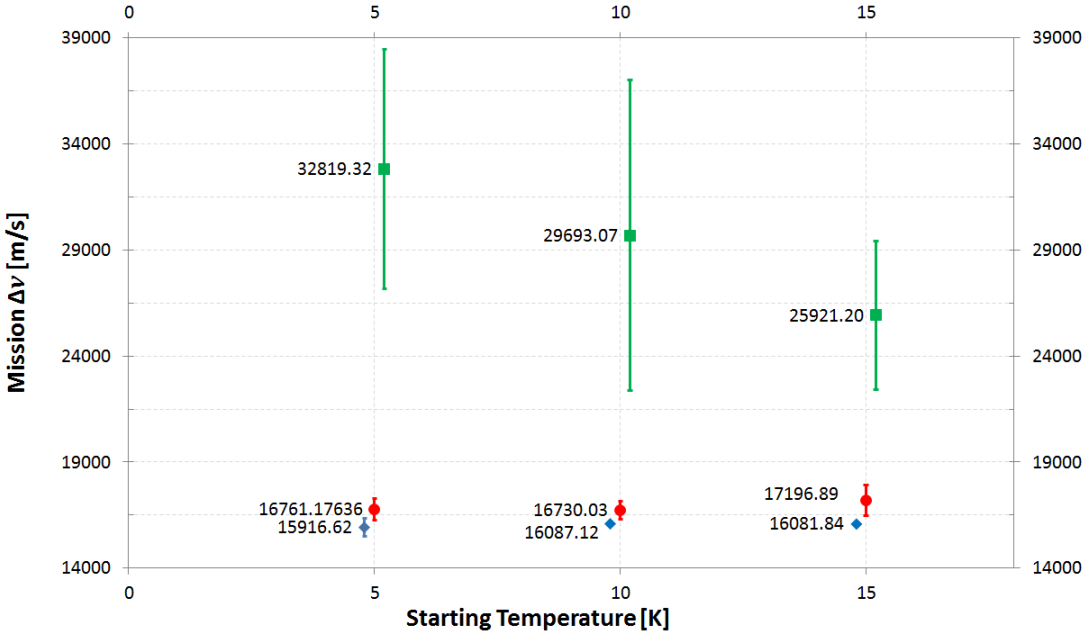

Figure 7-15: Average optimization results (mission Δ*v*) in dependence on starting temperature and number of segments. Blue (diamond) depicts results for 1 leg, red (circle) for 2 and green (square) 3 legs. The bars show the standard deviation of the solutions.

range of some 30 km/s, all others are in the range several thousand km/s (hence the standard deviation bar is exceeding the presented frame, selected to not obscure the remaining values).

Figure 7-17 depicts the best overall result found with Simulated Annealing with a Δ*v* demand of 13,772 m/s. The mission has two legs and conducts a gravity assist at Earth.

7.5.3 Discussion

The behavior of the search is somewhat unexpected in case of the variation of the number of iterations. A larger sample size should generally improve the solution quality as the probability of

finding the optimum region increases. However, for 1- and 2-leg missions the mission Δv increases when increasing the iteration number from 100,000 to 200,000 iterations. A similar pattern is visible for the standard deviation and also for the 3-leg solutions the drop in mission Δv is approximately only 10% for doubling the number of iterations. This can be attributed to an increase in diversity. The larger amount of iterations allows more divergence from the original good solution and in general increases the spread of the search in the search space. The chance of actually moving into beneficial regions with the search increases then again for even larger numbers, which is why the calculations with 500,000 show the best outcome for each respective amount of legs.

In general the performance of Simulated Annealing has been worse than for Differential Evolution. On average the 3-leg solutions took 290,000 samples to finish the calculations for the largest population size (500 members), but for Simulated Annealing using 500,000 iteration, almost twice as many (which also means roughly twice the amount of computation time), the mission Δv is more than 3 km/s worse. This means even when assuming that for a further increase in iteration numbers, the associated increase in computation effort would make this prohibitive.

The generally worse performance in comparison to Differential Evolution can also be attributed to the fact that for Simulated Annealing only a single solution is repeatedly modified. Due to the variable structure only a very limited amount of information can be reused resp. evolved and in case of Simulated Annealing only for one solution at a time. If a modification of a solution results in worse performance and is still accepted, as it is the case for early temperatures, this can lead the search away from an optimum. While this is intended to ensure diversity of the search, the fact that not all variables are evolved complicates things. Because even if initially good performing values of the global variables are refound and used, different settings for the local variables can create a bad overall performance of a particular solution candidate, preventing that this solution is accepted, especially at low temperature ranges. This way the algorithm has trouble returning to beneficial regions, even if it contains the optimal solution.

Differential Evolution has to overcome the same obstacle, but due to its usage of a population of solutions, the chance of retaining beneficial values of the global variables is larger and therefore even if the modification of one solution moves the search away from a beneficial area, other solutions can reintroduce it into the search if they have had similar values for the global variables. The chance of losing the beneficial information is smaller than for Simulated Annealing.

The behavior of the two different methods of temperature drop is as expected. As seen before, the optimization cannot be regarded as finished at 100,000 iterations. Therefore, it is understandable that the results for the linear drop – which retains larger temperatures for a longer time than the exponential drop and thus retains a larger probability for accepting worse solutions – are worse performing on average than for the exponential drop. Exponential drop converges more quickly and conducts a less diverse search, because worse results are not as easily accepted, making it difficult for the search to leave local optima and explore other regions of the search space. This apparent advantage of the exponential drop is lost for multi-gravity-assist missions, which can be attributed to the search space topology, see section 7.10.2.

Overall, at this small number of iterations a final evaluation of the temperature drop method cannot be made, as the optimization cannot be regarded as finished. However, larger numbers of iterations are prohibitive because of the computation effort involved with Simulated Annealing.

A clear decision regarding best temperature is also not possible, as for the 1- and 2-leg-missions the results are ambiguous and no trend is visible. The difference between the results is small, between 31 and 500 m/s. The larger temperature however is beneficial to the multi-gravity-assist missions, where the starting temperature of 15 K results in the best overall outcome and also

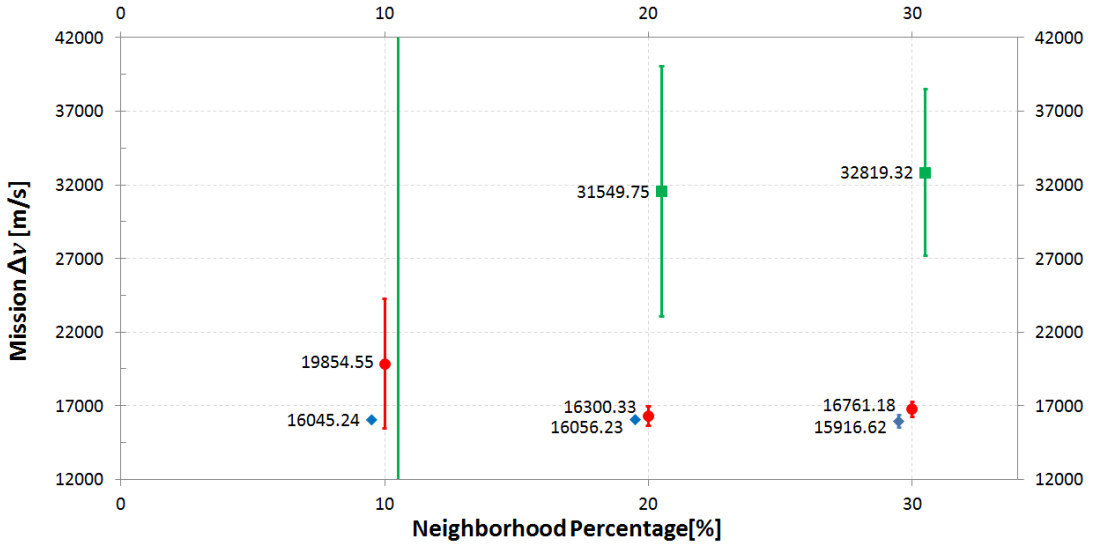

Figure 7-16: Average optimization results (mission Δ*v*) of Simulated Annealing in dependence on neighborhood percentage (see Eq. (7-4) and (7-5)) and number of segments. Blue (diamond) depicts results for 1 leg, red (circle) for 2 and green (square) 3 legs. The bars show the standard deviation of the solutions.

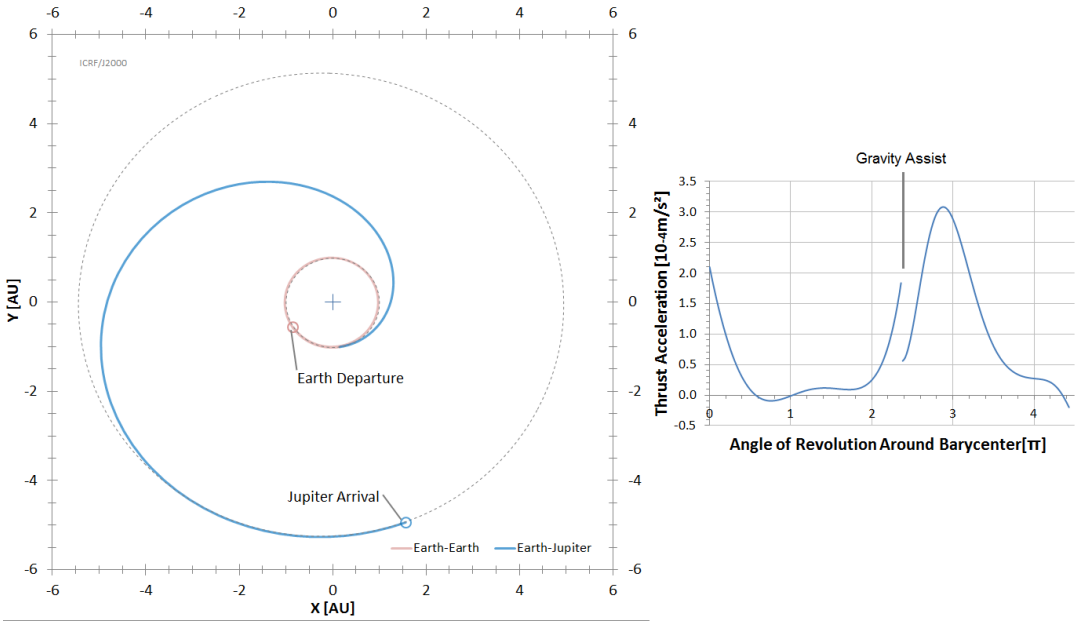

Figure 7-17: The best solution found Simulated Annealing, involving a single gravity-assist at Earth, with a total mission Δ*v* of 13.7 km/s. The coordinates are in the International Celestial Reference Frame and scaled in Astronomical Units (AU). On the right side, the resp. thrust profile over mission angle in π.

standard deviation (3,517 m/s). The average mission Δ*v* obtained with 100,000 iterations at 10 K is almost identical to that obtained for 5K with 200,000 iterations, the one obtained at 100,000 and 15 K is significantly better. This can also be attributed to the fact that the higher starting temperature allows more diversity, because the probability of accepting worse solutions is increased for the larger temperature in comparison to lower starting temperatures and identical iteration steps. This means that the exploration of the search space is better. Especially for multi-rendezvous-missions, a large temperature is desirable.

As to be expected, the neighborhood percentage does have a strong influence on the mission outcome for all numbers of legs. It can clearly be shown that for too small percentages the search through the solution space is very incomplete, leading to results that are several orders of magnitude worse than any other previously obtained calculation. For the two larger percentages, the results favor 20% in case of the gravity-assist missions. This fits that the 30% is limiting the convergence more than the 20%, but the fact that no significant change occurs, shows that the search quality is also not significantly improved. The change of neighborhood percentage does not influence the solution quality for these low numbers of iterations.

Figure 7-17 shows that good solutions are possible and the thrust profile is similar to those of the results gained by Differential Evolution. First an orbit change is achieved, followed by some time of relatively little thrusting and afterwards a maneuver allows the actual gravity assist. But the given example is the actual best solution found and is of less quality than the best solution obtained by Differential Evolution.

Overall, the numbers of iterations and the starting temperature have the strongest influence on the outcome of optimization with Simulated Annealing and should be carefully selected in case of applying that algorithm. However, even with more computation effort (i.e. numbers of iterations) and more solutions evaluated, Simulated Annealing cannot outperform Differential Evolution, as explained above. Therefore and because of the large computational effort involved, calculations combining all "best" setting found in these investigations, have not been attempted.

7.6 Optimization Constrained by Maximum Δv Change and Partner Pool

With reference to Chapter 5.6.1 calculations were conducted to investigate the usefulness of forcing the optimizer into a region of a gravity assist granting the maximum possible Δv. The basic concept is to exploit the energy benefits from a given gravity assist as extensively as possible, i.e. forcing gravity assists to be close to the theoretical maximum. Exploiting the energy benefits like this would mean a lower number of gravity assists and thus probably mission flight time and complexity.

The downside is that considering a gravity assist to be optimal based on its Δv outcome, does not necessarily mean an effective sequence can be found. Forcing a gravity assist to be close to the Δv maximum could make it more complicated to actually find subsequent gravity assists for the final approach due to orbital and phasing characteristics. For example conducting such a gravity assist a spacecraft might have too much energy for a rendezvous with the target body and therefore the spacecraft would need to reduce its energy by more thrusting, actually reducing the mission performance.

To analyze this, a number of calculations have been set up, investigating if sticking to the possible Δv maximum provides advantages for the search. Furthermore, it has been investigated whether or not limiting the possible pool of gravity-assist partners is beneficial to the optimization result. Sorting unlikely useful candidates out beforehand has been the goal, see section 5.6.2.

For this, two new parameters have been introduced into the calculations, d_{peri} and d_{velo}, each defined as value between 0 and 1. During the optimization as shown in Chapter 5.4, the turning angle δ is providing the value for the pericentre distance of the flyby trajectory. This distance is constrained by the minimum, r_{min}, and maximum, r_{SOI}, allowed distances, see Table 7-1. When using these constraints instead of randomly selecting the turning angle, the pericentre distance is selected based on the mathematical definition of d_{peri}:

$$r_{min} \leq r_{peri} \leq r_{min} + d_{peri} \cdot (r_{SOI} - r_{min}). \quad (7\text{-}6)$$

Once the pericentre distance is set, the hyperbolic excess velocity is also determined randomly, which directly yields the respective Δv change. The parameter d_{velo} defines how far from the hyperbolic excess velocity allowing the maximum Δv change as derived in section 5.6.1 the v_∞ is allowed and thus provides a constraint on the hyperbolic excess velocity in the form of:

$$v_{\infty,ex} \cdot (1 - d_{velo}) \leq v_\infty \leq v_{\infty,ex} \cdot (1 + d_{velo}). \quad (7\text{-}7)$$

7.6.1 Optimization Settings

For comparison, calculations were conducted with identical settings as given in Chapter 7.4.1, yet with a population size of 100. Gravity assists were controlled with additional variables providing constraints by using d_{peri} and d_{velo}. Likewise, the limiting of the partner pool was tested by first limiting it to 1 step out and inwards (e.g. going from Earth to Mars, or Jupiter back to Mars and not choosing Jupiter as a flyby partner after a flyby – or launch – at Earth). Then 2 steps forward and 1 backward, then 2 for both cases.

The parameter d_{peri} was varied in 10% steps, finding the best setting and afterwards d_{peri} was set to 20% and d_{velo} was varied in the same manner. Similar as before the partner pool was constrained, but also in combination with the constrained regarding Δv change. In a final phase both values have been set to 20% and the partner pool limits have been set to 1 for both directions.

7.6.2 Results

All runs resulted on average in a mission Δv below the no-gravity assist solution. Depending on the exact settings of the respective parameters, the results averaged between 14.2 and 15.9 km/s and standard deviations between 348 and 1,281 m/s. The best found trajectory in this scheme has a Δv requirement of 12.6 km/s.

The best improvement with regards to Δv change has been obtained for a pericentre distance of factor 1.2 of the minimum pericentre distance, i.e. a d_{peri} of 20%. This decreased the Δv mildly (190 m/s) in comparison to gravity-assist solutions without application of the constraint, but decreased its standard deviation by 65% from 1,000 m/s to 348 m/s, i.e. the reliability of the search has been improved.

Setting the allowed Δv change difference to 40% provided the best result in terms of standard deviation (448 m/s) for a d_{peri} of 20%. Other than that attempting to find better Δv by using this constrained provided improvement of about 360 m/s.

Varying the stepsize of the partner pool between 1 and 2, respectively differentiating between forwards (outwards for the particular mission) and backwards (inwards for the particular mission), shows the best results for only allowing partners within a stepsize of 1 with a Δv of 14.2 km/s, setting the parameters d_{peri} and d_{velo} to 20 resp. 40%. The standard deviation was however better for using a stepsize of 2 forward and 1 backward, 838 m/s vs. 719 m/s. Not using the Δv constraint provides the best results for a stepsize of 2 forwards and backwards with a mission Δv of 14.2 km/s and at the same time the highest standard deviation of 1,218 m/s.

Similarly to before, the gravity assist encounter was found mostly for Earth and at an encounter date around 56470 MJD, see section 7.4.2.

Table 7-16: Summary of the optimization outcome for Differential Evolution and usage of the constraints as 20% for both d_{peri} and d_{velo} and with a limit of a step size of 1 forward and backward regarding GA partner selection and without any constraints for a population size of 100.

Parameter/No. of legs	2	3
$\Delta v_{mission}$ w/ constraints (average/standard deviation)	14,071.0/562.7	26,163.6/ 5300.5
$\Delta v_{mission}$ w/o constraints (average/standard deviation)	14,099.6/ 931.4	34,323.2/ 8167.6

In the runs directly comparing the usage of the constraints' previously best settings with omitting the constraints altogether does not show relevant improvement of the average Δv (14.1 km/s vs. 14.07 km/s for usage of the constraints), but it reduces the standard deviation significantly (563 m/s vs. 931 m/s for calculations without usage of the constraints). This can be similarly derived for multi-rendezvous missions.

The results for the last phase of the tests are summarized in Table 7-16. The average Δv is slightly smaller in case of two segments, but significantly smaller, more than 8 km/s for three-leg-missions. In both cases also the standard deviation is has reduced.

7.6.3 Discussion

It is important to note that the inclusion of the constraints described in Chapter 5.6 does not always result in improvements of the Δv demand for the mission, but it reliably reduces the standard deviation. Therefore, usage of the constraints is not helping to find better solutions so much as to find similar solutions. Especially the latter is expected. Introducing constraints into the search means reducing diversity within the search. Before, increasing the population size has been the method with which to increase the chance of reliably finding similar results, this can be achieved with less population members already with these constraints. Increasing reliability with less population members means faster calculation times and easier repetition of solutions. Therefore, it is deducted that usage of these constraints is mainly a method of improving calculation time, without change of solution quality.

The parameter settings as used here are assumed to be sensitive to the respective mission and should be carefully set for each new mission scenario.

7.7 Comparison with *Dawn*

For comparison with actual mission results, first calculations for a *Dawn*-like mission have been conducted. The comparison has been conducted with ten calculations and utilizing Differential Evolution, as this algorithm performed best within the group the three selected algorithms.

Based on *Dawn*'s actual mission data (Launch Date: 54370, Flight Time to Vesta: 1388, First leg's flight time (to Mars): 509), search properties as documented in Table 7-17 have been selected. A sketch of the original trajectory can be seen in Figure 7-18.

The gravity assist occurred in *Dawn*'s first mission part, i.e. during the transfer from Earth to Vesta. Thus, only this mission portion has been optimized. The subsequent transfer from Vesta to Ceres hat not been regarded as it does not provide any insight into the quality of the search for gravity-assist partners.

Table 7-17: Listing of properties as used for the optimization with Differential Evolution in the set of calculations re-calculating the *Dawn* mission.

Property	Value
Stopping Criterion	Max. number of generations (1,000) or 5 times in a row without replacement
Population Size:	500
Crossover Constant	0.75
Differential Weight	1
Launch Date Window:	54300 plus 150 days
Time of Flight Interval:	1,200 to 1,500 days
Maximum Number of Gravity Assists	1
Maximum Number of Revolutions	2
Maximum Thrust Acceleration	1.15×10^{-4} m/s²

7.7.1 Results

The results are summarized in Table 7-18. The gravity-assist mission results have been on average a Δv of 11,164.14 m/s with a standard deviation of 1,890.41 m/s. The best obtained result has been 8,415.1 m/s. Six times the result has been below *Dawn*'s actual 11 km/s, but only three times below the non-gravity assist result obtained within this calculation (note: the non-gravity assist result is assumed to be lower than that of the actual mission flown, because of the disregard of inclination changes). The Launch Date has been on average 54334.4 with a standard deviation of 43.42 and the Flight Time has been 1,455.6 d on average, resp. with a standard deviation of 37.74 d. In all but one instances Earth has been the selected gravity-assist partner, only once Mars has been selected (at 9,763 m/s). In general, two regimes of results can be seen: one favoring a very short (70 to 90 days) flight time before meeting the gravity-assist partner and a long flight time of this first leg, (about 330 to 360 days). In both cases similar encounter dates with the gravity-assist partner have been obtained. For the first regime, it has been between 54391 and 54433 and for the second regime it has been between 54638 and 54668. The results with a larger Δv than the actual mission are all associated with the short flight time.

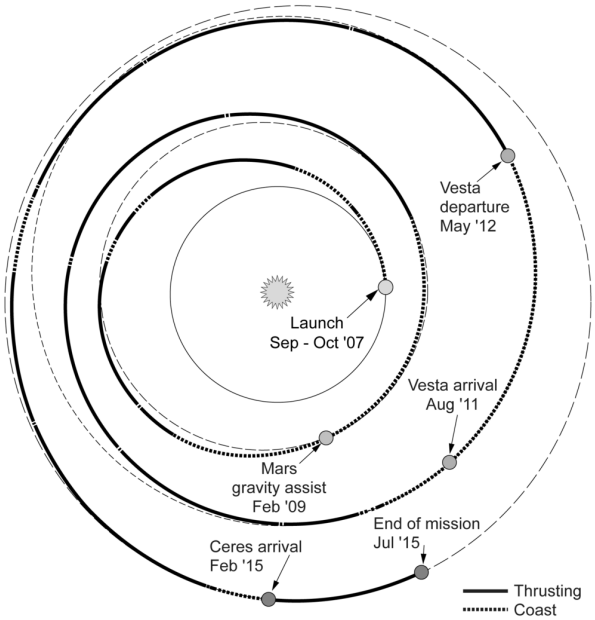

Figure 7-18: A sketch of *Dawn*'s original trajectory for the whole mission from Earth to Vesta and from Vesta to Ceres with a gravity assist at Mars in the first leg [image: NASA, public domain].

Table 7-18: Summary of the optimization outcome for *Dawn*'s trajectory. Mission Δ*v* in m/s, Launch Date in MJD, Time of Flight in days. Original Launch Date: 54370 and Flight Time: 1388.

Parameter/No. of legs	1 (0 GA)	2 (1 GA)
$\Delta v_{mission}$ [average/standard deviation]	10,120.3/ 0	11,164.1/1890.41
# of Generations [av./standard dev.]	61.1/ 1.29	657/ 163.2
$LD_{mission}$ [av./standard dev.]	54450/ 0	54334.4/ 43.4
$ToF_{mission}$ [av./standard dev.]	1,431/ 0	1,455.3/ 37.7
$N_{rev,mission}$ [av./standard dev.]	2/ 0	2. / 0
1st GA Partner [# of times occurring]	-	Earth: 9, Mars: 1
GA Encounter Date [av./ standard dev.]	-	54569.6/ 190.5
$ToF_{1st\ leg}$ [av./ standard dev.]	-	235.2/ 170
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	0.5/ 0.5

7.7.2 Discussion

It can clearly be seen that the results obtained for a *Dawn*-like mission deviate measurably from the actual mission, but are in the same order of magnitude. While the average Δ*v* is similar to the original mission's, there have been results with an improvement of more than 2.5 km/s. The launch date has a comparable value and only deviates by 36 days on average, i.e. 24% of the respective launch window as used for the optimization. The flight time differs by about 70 days, which is about 23% of the respective window used.

However, the gravity-assist partner is deviating in all but one cases. The reason for this could be that the model used in this tool is only planar, whereas *Dawn* is a three-dimensional mission. Vesta has an inclination of about 7°. Furthermore, the trajectory flown by *Dawn* has not been a mission enabler, but only increased the mission's margin (Rayman et al., 2006). This is likely the reason why the best result obtained by this method is indeed improving the actual mission's quality by more than 2 km/s, possibly attributed to Earth being the gravity-assist partner. At the same time, the average Δ*v* obtained by GOLT and its difference to the original value stress once more the complexity of this kind of mission. The fact that two regimes of possible gravity assist scenarios are part of the results, i.e. one with a significantly lower Δ*v* than the actual mission and one with a larger one , shows that the evolutionary algorithm has problems finding the global optimum. In this case the algorithm had to deal with the fact that likely only two gravity-assist partners are valuable, Earth and Mars, as they are located between Earth and Vesta in terms of radial distance. Due to its large distance using Jupiter as gravity-assist partner can hardly be considered beneficial. And as can be seen in that example, obtaining the actual gravity assist opportunity has been challenging, although Mars as gravity-assist partner has resulted in a mission Δ*v* of 9,763 m/s. This improvement over the actual mission can be explained by the fact that here only the planar trajectory proportion has been considered and also *Dawn* used the gravity assist for improving its mission margin, not as an enabler. Therefore, it likely has not been an optimal solution, but a solution which also provided the opportunity of scientific measurements at Mars.

7.8 Comparison with *Hayabusa 2*

As a second mission, *Hayabusa 2* has been investigated, also with a set of ten calculations. The respective settings are listed in Table 7-19. For these calculations the thrust has been assumed as 28 mN (assuming one thruster operating), resulting in a maximum thrust acceleration of 5.6×10^{-5} m/s^2 for a mass of 500 kg (Tsuda et al., 2013). While the mission has been launched in December 2014, the actual interplanetary operation started on 3rd March 2015 (MJD 57084). Therefore, this is set as mission launch date +/- 20 days. *Hayabusa 2*'s planned arrival date at $1999JU_3$ is 11th June 2018 (Tsuda et al., 2012), i.e. MJD 58280. Thus, the flight time is 1196 days. For the optimization,

Table 7-19: Listing of properties as used for the optimization with Differential Evolution for re-calculating the *Hayabusa 2* mission.

Property	Value
Stopping Criterion	Max. number of generations (1,000) or 5 times in a row without replacement
Population Size:	200
Crossover Constant	0.75
Differential Weight	1
Launch Date Window:	57064 plus 40 days
Time of Flight Interval:	1,100 to 1,400 days
Maximum Number of Gravity Assists	1
Maximum Number of Revolutions	3
Maximum Thrust Acceleration	5.6×10^{-5} m/s²
Allowed pericenter distance d_{peri}	0.2
Allowed velocity difference d_{velo}	0.2
Partner pool constrained	1 outward, 1 inward

the constraints as tested in Chapter 7.6 have been used. For d_{peri} and d_{velo} the values have been set to 0.2 respectively and the partner pool has been constrained by 1 into each direction.

The GA partner has been Earth for the original trajectory design and supplied a Δv of 1.6 km/s, flying by Earth at an altitude of 3000 km on 3rd December 2015 (MJD 57359). The original trajectory has been calculated with the help of Nonlinear Sequential Quadratic Programming, by dividing it into short nodes and the control variables have been the boundary conditions of each node (Tsuda et al. 2013), i.e. by assessing the fitness of neighboring solution candidates, a direction of where new solution candidates should be investigated, is determined.

7.8.1 Results

The results provided by the calculations do not closely resemble the actual mission and are summarized in Table 7-20. The launch date has a difference of ca. one day with a calculated result of 57085.5 on average and a standard deviation of about 10 days. The gravity assist occurs approximately 300 days later than in the actual mission, on average at MJD 57669 with a standard deviation of approximately 85 days, so almost three months. However, the GA partner is always Earth, just as for the actual mission trajectory. The flight time is on average 1252.4 days, i.e. about 56 days larger than in the original trajectory.

7.8.2 Discussion

As a comparison, the described method is not able to fully reproduce the trajectory of *Hayabusa 2* but also not to improve it, which is mainly caused by the divergence regarding the gravity assist

Table 7-20: Summary of the optimization outcome for *Hayabusa 2*'s trajectory . Mission Δv in m/s, Launch Date in MJD, Time of Flight in days. Original Launch Date: MJD 57084 and Flight Time: 1196 d.

Parameter/No. of legs	1 (0 GA)	2 (1 GA)
$\Delta v_{mission}$ [average/standard deviation]	8,269.3/ 0	12,470.3/1493.8
# of Generations [av./standard dev.]	440.7/ 57.2	265.8/ 58.7
$LD_{mission}$ [av./standard dev.]	57064/ 0	57085.5/ 10.1
$ToF_{mission}$ [av./standard dev.]	1,146/ 0	1,252.4/ 67.1
$N_{rev,mission}$ [av./standard dev.]	2/ 0	2.1 / 0.3
1st GA Partner [# of times occurring]	-	Earth: 10
GA Encounter Date [av./ standard dev.]	-	57669/ 85.8
$ToF_{1st\ leg}$ [av./ standard dev.]	-	583.5/ 89.9
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	1.1/ 0.3

encounter date. The literature does not provide information about the actual required mission Δ*v*, but in no instance the gravity-assist version of the trajectory as calculated can improve the mission Δ*v* as needed without gravity assist. The best gravity-assist result is 9,939 m/s, which is about 1.7 km/s larger than the non-gravity-assist result.

However, the overall flight time has a difference of 4.7% when compared to the original flight time and 19% of the flight time window as provided for the optimization. The overall flight time therefore fits the original mission quite well. The gravity-assist partner is always identical, but the encounter date is dissimilar. This difference in the gravity assist is likely the reason for the non-optimal mission performance when comparing the mission Δ*v* of the non-gravity-assist and gravity-assist cases.

7.9 Comparison with *BepiColombo*

As a third comparison *BepiColombo* has been investigated. BepiColombo is a multi-gravity assist mission targeted at Mercury and is currently still in development. The respective trajectory design has changed several times, e.g. from an arrival date in 2017 (García Yárnoz and Jehn, 2006) over late 2020 (Benkhoff et al., 2010) to 2024 (Jehn and Langevin, 2016). The design has evolved over the past decade and has also been extensively optimized. As measure for risk reduction, 30 day coast arcs are included before a gravity assist (García Yárnoz and Jehn, 2006). In total a series of seven gravity assists are planned at Venus and Mercury and electrical propulsion is used to achieve a mission Δ*v* of 5 to 5.8 km/s (Benkhoff et al., 2010), later refined to 4.14 km/s (Jehn and Langevin, 2016). Due to Mercury's inner position in the solar system, a Venus flyby is considered as mandatory by the mission designers, which can also be linked to the Tisserand graphs in Figure 2-3. It can be seen that a transfer from Earth to Mercury with gravity-assists is otherwise difficult. These gravity assists are obtained applying coasting arcs, i.e. no propulsion is assumed and the associated Lambert's problem is solved to provide the results (Jehn and Langevin, 2016).

Due to its completeness, the trajectory data for calculations is taken from an early trajectory design (Jehn at al., 2008), but the principal trajectory is not differing from later designs. The settings used for the calculations are listed in Table 7-21. The reference launch date is 1st August 2013 (MJD 56517), which is taken with a +/- 50 days frame, thus setting the launch date to MJD 56455 plus 100 days as window. The original arrival date is 4th June 2019 (MJD 58638), i.e. flight time is 2121 days, therefore, the interval is set to 1900 to 2300. With the limitations of previous

Table 7-21: Listing of properties as used for the optimization with Differential Evolution for re-calculating the *BepiColombo* mission.

Property	Value
Stopping Criterion	Max. number of generations (1,000) or 5 times in a row without replacement
Population Size:	200
Crossover Constant	0.75
Differential Weight	1
Launch Date Window:	56455 plus 100 days
Time of Flight Interval:	1,900 to 2,300 days
Maximum Number of Gravity Assists	2
Maximum Number of Revolutions	8
Maximum Thrust Acceleration	1.45×10^{-4} m/s²
Allowed pericenter distance d_{peri}	0.05
Allowed velocity difference d_{velo}	0.2
Partner pool constrained	1 outward, 1 inward

Table 7-22: Summary of the optimization outcome for *BepiColombo*'s trajectory . Mission Δ*v* in m/s, Launch Date in MJD, Time of Flight in days. Original Launch Date: MJD 56517, Flight Time: 2121 d.

Parameter/No. of legs	1 (0 GA)	2 (1 GA)	3 (2 GA)
$\Delta v_{mission}$ [average/standard deviation]	36,250.5/ 1	27,316.4/ 1381.4	40,539.6/ 3188.3
$LD_{mission}$ [av./standard dev.]	56524/ 0	56485.9/ 24.5	56494.3/ 27.47
$ToF_{mission}$ [av./standard dev.]	1921/ 0	2033.6/ 77.9	2,098.25/ 117.3
$N_{rev,mission}$ [av./standard dev.]	2/ 0	4.1/ 0.6	3.75/ 0.6
1st GA Partner [# of times occurring]	-	Venus: 1, Earth: 6, Mars: 3	Earth: 7, Mars: 3
GA Encounter Date [av./ standard dev.]	-	57903.3/ 361	57380.5/ 477
$ToF_{1st\ leg}$ [av./ standard dev.]	-	1417.4/ 351	886.3/ 473.3
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	3.2/ 0.4	1.63/ 0.5
2nd GA Partner [# of times occurring]	-	-	Earth: 2, Mars: 8
GA Encounter Date [av./ standard dev.]	-	-	58039.5/ 482.5
$ToF_{1st\ leg}$ [av./ standard dev.]	-	-	659/ 177.9
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	-	1/ 0

calculations in mind, only 2 gravity-assist maneuvers are set as maximum number. Assuming a thrust of 290 mN at a dry mass of 2000 kg, a maximum thrust acceleration of 1.45×10^{-4} m/s² is set.

The original trajectory has been calculated with combinations of various software such as DITAN and MANTRA, specialized for different aspects of the optimization, including navigation constraints, which are not part of the method investigated in this thesis (Jehn and Schoenmaekers, 2014). Partly the trajectory design involves coasting arcs, where no propulsion is applied, mostly to exploit orbit period resonances (Jehn and Schoenmakers 2014).

7.9.1 Results

The results given in Table 7-22 show two things: Based on simple approximations, the Δ*v* needed for a low-thrust transfer can be estimated as the difference between the respective circular velocity (Messerschmid and Fasoulas, 2005). For the case of Earth and Mercury, this results in ca. 16 km/s, which is also the limit for the original trajectory calculations as given by (Jehn and Schoenmaekers, 2014). The non-gravity-assist solution is 20 km/s larger than this limit.

The single gravity-assist solution does not show an improvement over these mentioned 16 km/s. Its outcome is below the calculated the non-gravity-assist solution by on average ca. 25%, i.e. about 9 km/s. Yet, its outcome is on average more than 10 km/s larger than the mentioned limit. The flight time for the mission is similar to the original with a difference of about three months, i.e. 4% of the original flight time.

Also in no cases is the sequence of gravity-assist partners similar to the original mission design. Earth occurs as first gravity-assist partner in several cases, but never in combination with Venus as second, which has been expected based on the original design. The multi-gravity-assist trajectory is only slightly longer in terms of flight time.

7.9.2 Discussion

The gravity-assist solutions obtainable within these calculations do not resemble those calculated by Jehn and Langevin (Jehn and Langevin, 2016) or others found in the literature (see p. 114). Instead, the solutions produced within this work exceed the given results of mission Δ*v* significantly. While the single gravity-assist improves over the non-gravity-assist solution, the

results are in every case larger than the theoretical limit of 16 km/s as non-gravity-assist benchmark (Messerschmid and Fasoulas, 2005). The optimization has for all runs only produced Venus as gravity-assist partner once and preferred Mars as second gravity-assist partner. These effects are all attributed to the long flight time. The significant flight time introduced here is sufficient to allow the seven gravity assists of the original trajectory, but also means a significant increase in required mission Δv as coast arcs cannot be modelled by shape based models, unless they are very close to a circular orbit. Using smaller flight time, the shape based model produces a mission Δv around 18 km/s (see Section 7.10), which is reasonable, taking into account that Mercury has a very eccentric orbit and thus requires a larger mission Δv than the theory prescribes. The same is true for the gravity assist partner. Associated with the long flight time and thrust history, outer planets are more useful and can be easily targeted due to their smaller velocity and thus less dynamic phasing.

The lack of such good solution candidates suggests that the path assumed for the original trajectory, i.e. first a gravity-assist at Earth and then several at Venus before arriving at Mercury and conducting flybys there, is prudent and needs no further refinement.

7.10 Test Calculations to Mercury

To test if a different window of flight time produces better results for a Mercury mission, ten further calculations have been set up. The properties used for the calculations are listed in Table 7-23, this time the possible flight time is between 500 and 1,500 days, as opposed to 1,900 to 2,300 before during the calculations concerning *BepiColombo*. The spacecraft is assumed to be similar to *BepiColombo*. Only a single gravity assist has been intended.

7.10.1 Results

The results for the calculations are provided in Table 7-24; the encounter dates for the respective gravity-assist partner are given in Figure 7-19. It can be seen that the 1-leg mission results are identical in Δv, which is 18.177 km/s, and thus about 20 km/s smaller than in the previous calculations regarding *BepiColombo*. The standard deviation for all non-gravity-assist results is in all cases zero, due to the analytical shape-based model and the discrete variables (see Eq. (7-3)).

The best Δv result for the gravity-assist missions has been 15.726 km/s, an improvement of 13.5% in comparison to the non-gravity-assist solution. On average the mission Δv is 17.651 km/s, the

Table 7-23: Listing of properties as used for the optimization with Differential Evolution for the recalculation of the Earth to Mercury mission.

Property	Value
Stopping Criterion	Max. number of generations (1000) or 5 times in a row without replacement
Population Size:	300
Crossover Constant	0.75
Differential Weight	1
Launch Date Window:	56400 plus 720 days
Time of Flight Interval:	500 to 1,500 days
Maximum Number of Gravity Assists	1
Maximum Number of Revolutions	8
Maximum Thrust Acceleration	1.45×10^{-4} m/s²
Allowed pericenter distance d_{peri}	0.2
Allowed velocity difference d_{velo}	0.2
Partner pool constrained	1 outward, 1 inward

Table 7-24: Summary of the optimization outcome for the Earth to Mercury calculations. Mission Δ*v* in m/s, Launch Date in MJD, Time of Flight in days.

Parameter/No. of legs	1 (0 GA)	2 (1 GA)
$\Delta v_{mission}$ [average/standard deviation]	18,177/ 0	17,651.4/ 1,442.2
# of Generations [av./standard dev.]	184.2/ 7.2	340/ 67.4
$LD_{mission}$ [av./standard dev.]	56967/ 0	56691.8/ 183.9
$ToF_{mission}$ [av./standard dev.]	508/ 0	1,099/ 182.6
$N_{rev,mission}$ [av./standard dev.]	2/ 0	3.3/ 0.48
1st GA Partner [# of times occurring]	-	Venus: 2, Earth: 8
GA Encounter Date [av./ standard dev.]	-	57591.6/ 347
$ToF_{1st\ leg}$ [av./ standard dev.]	-	894.8/ 214.3
$N_{rev,1st\ leg}$ [av./ standard dev.]	-	2.3/ 0.7

associated standard deviation is 1.44 km/s. Earth (8 times) and Venus (twice) occur as gravity-assist partners. For Earth three different encounter dates occur more than once, around 57350 MJD, 57550 MJD and 57750 MJD and in each of these, solutions with less mission Δ*v* than the non-gravity-assist solutions occur. For Venus the encounter dates were quite similar, 57996 MJD and 57999 MJD, but only the first of which produced a result below the non-gravity-assist mission, i.e. 16.5 km/s, the latter produced a mission Δ*v* of 18.5 km/s.

7.10.2 Discussion

As anticipated, the reduction of the allowed flight time leads to better results for non-gravity-assist and gravity-assist missions alike. The occurrence of three repetitive encounter dates are attributed to the faster orbital period of Mercury and the more dynamic phasing, leading to more gravity-assist opportunities, as described in (García Yárnoz and Jehn, 2006). Meaningful differences in mission Δ*v* cannot be detected, which supports this assumption. While the result is not similar in magnitude to those of the original *BepiColombo* mission, in difference to before, gravity-assist

Figure 7-19: Single gravity-assist encounter dates for the Earth to Mercry calculations shown for the respective gravity-assist partners.

partners similar to *BepiColombo* are determined by the optimizer. The previously selected gravity-assist partner Mars does not occur in these calculations at all as a solution. This is also linked to the flight time fitting a single gravity-assist mission, as opposed to before using a flight time selected for a seven-gravity-assist mission.

The larger flight time in the previous calculation set forced the optimizer to adopt a trajectory that requires a large mission Δv, which is associated with the trajectory model. The shape based model allows thrusting with small thrust magnitudes, but still dedicated coasting arcs are not possible at any segment.

It can be observed that the optimizer is able to identify useful gravity-assist partners and encounters for this mission to the inner solar system.

7.11 Overall Optimization Performance

The basic goal of including the gravity-assist partner as a variable in the optimization for a low-thrust mission has been successful and dominating variables for the problem have been identified. The method should be improved regarding refinement of solutions, reproduction of results and convergence to the global optimum, however. Nevertheless, identification of the beneficial gravity-assist partners is possible, allowing the subsequent usage and further refinement of such a solution in a dedicated optimizer. Finding the best gravity-assist partner does not require an experienced mission analyst when applying the derived method, with the limitations discussed in Chapter 7.12. Gravity-assist partners can be identified for missions to the outer solar system (as shown for a mission targeting Jupiter) and the inner solar system (as shown for a mission targeting Mercury). Generally, Differential Evolution has outperformed the other algorithms and proved to be the most robust algorithm, which is attributed to the nature of the variables. There is a risk of losing valuable, beneficial information by modification of the solutions during the evolution. The obstacle that due to reinitiating the local variables for every solution candidate this beneficial information cannot be detected as such, due to a bad combination with local variable values obscuring the beneficial nature of the global variable settings, can be mitigated by the populations associated with Differential Evolution. There is a larger chance for retaining beneficial information in a population than for a single solution.

7.11.1 Astrodynamical Implications

For the calculations concerning the rendezvous transfer from Earth to Jupiter, in all instances of Differential Evolution calculations with 200 population members and more, the GA-partners have been identical to the launch body (Earth) for all single-gravity assist solutions. The same is the case for 70% of the 50 member calculations. This suggests that reusing gravity-assist partners similar to the launch body for this kind of mission is advantageous. This can likely be attributed to the fact that it is easier to reencounter an already encountered planet with small changes of the respective spacecraft orbit. This is visible in the thrust profiles showing minor thrusting in the first segments. This is not possible for impulsive missions, where once a trajectory is set, it can only be modified by the flyby itself.

Similarly, the inner body (Earth) has been produced as best result in all cases containing 500 population members for 2 legs and while considered as the second gravity-assist partner for 200 population members and 3 legs, this did not occur for 500 population members, where only Mars and Earth occur 3 resp. 7 times. In combination with the tendency to "reencounter" the gravity-assist partner, it can be assumed that it is easier to reach an inner body a second time, due to the smaller orbital period of the respective body, which allows it to obtain another beneficial position more quickly. In no instance has Jupiter been used more than once and in no instance has the body for the second gravity assist been at a more inward position than the first one (e.g. Earth vs. Mars).

A similar effect can be seen for the calculations targeting Mercury, where Earth has been the most occurring gravity-assist partner, which supports this explained observation. At the same time the small change of the orbital position of Jupiter in case of the first investigated mission scenario, allows the optimizer to identify gravity assist encounters which are very similar, as opposed to those used for the Mercury mission. This is to be expected and likewise related to the change of positions which occurs faster in the inner solar system than the outer solar system.

7.11.2 Dominance of Optimization Variables and Search Space Topology

The major influence on the optimization results of the Earth-Jupiter calculations for single-gravity-assist trajectories are clearly the global variables of mission flight time, mission numbers of revolutions, mission launch date and the gravity-assist partner. Without evolutionary influence, a random search is capable of finding useful gravity-assist partners but not in the same quality as an evolutionary search. This is even more evidenced for multi-gravity-assist missions.

Concerning Differential Evolution, for 3 legs, the search results using 50 population members, ran for about 94 generations on average, i.e. on average 4,700 samples of the search space were evaluated, resulting in an average Δv of 29 km/s. Using 63,000 samples in a random search resulted in an average mission Δv of 26.2 km/s. An increase of factor 13.4 in regards of samples taken, resulted only in 10% of improvement regarding the mission outcome. A similar picture is painted for single-gravity-assist missions. Only for 50 population members, which on average mean 4,300 samples when multiplied with the respective average generation number, the outcome is worse than the best outcome obtained with a random search and using 63,000 samples. In this case the improvement is 7%, i.e. 13.74 km/s vs. 14.72 km/s. Even increasing the sample size for Differential Evolution to an average of 50,260 evaluated candidates leads to a better solution than found with 63,000 random samples, i.e. 13.19 km/s.

The fact, that a significantly larger sample size cannot produce a significantly better result when comparing random and evolutionary search, also provides an insight on the search space topology. It can be seen that the optimal regions are distributed over the search space and their area is not very large compared to the overall search space otherwise finding them randomly would be easy and random search would perform better than it actually does. The global optimum's area is not large as random search rarely reproduces similar results and shows large diversity of the final solutions. However, the fact that using the mission global variables for evolution produces better results shows that their influence outweighs the randomness of the local variables. The derivation of the local variables, e.g. the turning angle δ, is identical in both search types: random. And while the areas of beneficial values of the global variables are not large enough to be easily found by a random search, they are still large enough to be effectively used for evolving towards a better solution and not so small that a modification of the variables always leads away from better results. Otherwise, the optimization would not stop at generations of several hundred but earlier.

This dominance even occurs for multi-gravity-assist missions. It is clear that the areas of beneficial values become smaller, for that reason the overall results are of poorer quality than compared to single- or non-gravity-assist missions. Still, evolution via Differential Evolution leads to better results than random search, meaning areas do exist, which an evolutionary approach can use. A single solution for the Earth-Jupiter calculations even provided a better result than the non-gravity-assist mission solution.

In general, it can be said that the topology is relatively wide (and thus optimal regions are easier to find) for smaller numbers of gravity assists and becomes more spiked for larger numbers of gravity assists. At least theoretically it can be assumed that the more spiky the search space topology becomes, the larger also the possible fitness, because more gravity assists have the potential for more Δv savings. This is illustrated in Figure 7-20, showing a sketch of a possible two-dimensional projection of a search space topology. The difference between small numbers of gravity assists vs.

large gravity assist numbers can be seen. The potential fitness is larger for the latter, due to more potential for energy gains, but the optimal regions are narrower, due to the more complex variable situation. However, even for single-gravity-assist missions somewhat spiky terrain occurs, evidenced by the inability of Simulated Annealing to find solutions better than non-gravity-assist missions. The spiky topology is confirmed by the behavior of the two different temperature drops when viewing Simulated Annealing. While the faster converging method, exponential, is capable to outperform the linear drop for 2-leg-missions, this is not the case for 3-leg-missions. This indicates that even with a faster convergence it is difficult to find even a local optimum. Convergence is not the decisive factor when determining the outcome of a multi-gravity assist.

In addition, the areas of good results of certain variable values within the search space seem to be different. Especially the influence of certain gravity-assist partners seems to be larger. While apparently Earth can by application of Differential Evolution easily be identified for single-gravity-assist missions as best gravity-assist partner, it also is favored as the first partner in case of multi-gravity assist solutions. In all three regimes of the evolutionary search for the Earth-Jupiter mission, Earth occurs as first gravity-assist partner in 70% of the calculations. The second partner differs.

Concerning distance between optima, apparently the optima are closer in distance for inner solar system bodies as can be seen for the calculations concerning Mercury. The more dynamical situation in the inner solar system (García Yárnoz and Jehn, 2006), provides more opportunities for gravity assists, this is why not a single encounter but several are provided by the calculations targeting Mercury.

Although the control variables as used for the optimization are different, the spiky nature of search space topology fits the results described in (Carnelli, 2005). There the gravity-assist partner was not a control variable as in the approach presented here, but had to be found when calculating any

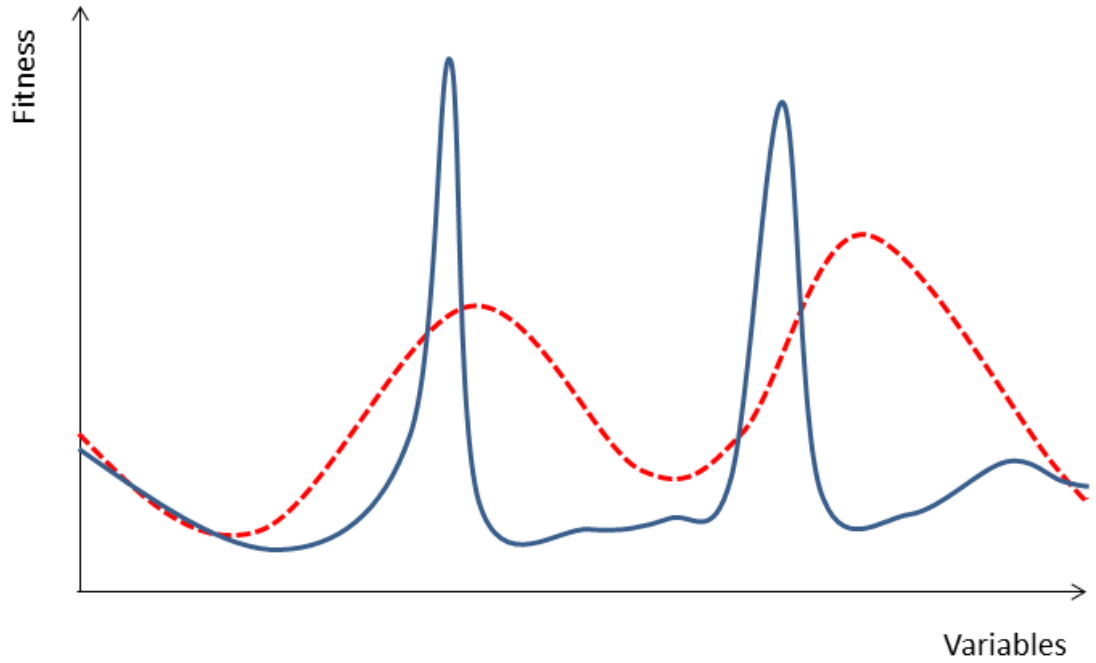

Figure 7-20: A sketch of a projection of an approximate search space topology regarding fitness over possible variable values, showing differences in steepness based on small numbers of gravity-assists (red/ dashed line) and large numbers of gravity-assists (blue/ solid line).

trajectory to the respective target. As more non-gravity-assist solutions exist, these were easier to optimize and prevailed in the respective population of solution candidates. This is related to the spikey nature of the search space topology, which is even intensified in cases where the trajectory is not forced to include a gravity assist. The forcing already excludes all trajectories without gravity-assist and therefore removes parts of the search space. The separation of the evaluation of gravity-assist trajectories and non-gravity-assist trajectories allows a focused evaluation of the first.

7.11.3 Comparisons with Existing Missions

In all instances, the attempted reproduction of the existing mission trajectories has led to different optima than the original calculations found in the literature and with few exceptions for *Dawn*, no better results have been produced.

For *Dawn* the optimization has been partially successful in producing some solutions which had a better (smaller) mission Δv than the original one and even on average the results were quite similar. However, the gravity-assist partner has been Earth instead of originally Mars. This is likely attributed to the fact that Mars has not been an optimal partner, as the gravity-assist has only been applied to increase the mission margin and furthermore has also been target of scientific experimentation during the mission. For that reason there have been also solutions which performed better than the original one. Proof in the literature for these assumptions has not been obtainable however.

Results for *Hayabusa* 2 have been diverging. While the overall mission has been similar in its properties, the gravity-assist encounter was not in its encounter date. The gravity-assist partner has been selected as Earth, but the date was different, which is possibly the reason for the gravity-assist trajectory to perform worse than the non-gravity-assist trajectory.

A similar, resp. even worse picture is painted for the calculations regarding *BepiColombo*. For this analysis the trajectories have been worse performing, including the non-gravity-assist solutions. It is apparent that the gravity-assist strategy obtained by the search method contradicts the strategy of the mission's original trajectory of first using Earth, then Venus as gravity-assist partner, but fails to produce better results with its own strategy. In contrast, Mars occurs both as first and second gravity-assist partner. The difficulties for reproducing *Hayabusa 2* and *BepiColombo* are likely linked to the following three aspects:

First, the inner solar system is very dynamic, due to the larger angular velocities of the respective bodies compared to the outer solar system, i.e. there is a quicker change of positions. While this is beneficial from concerning the number of gravity-assist opportunities (García Yárnoz and Jehn, 2006), this large number makes it problematic for the search to identify beneficial combinations resp. handle the phasing adequately. Slight changes of mission variables (local and global) to previously evaluated solution candidates can have a strong influence on the phasing and thus benefit of the gravity assist than for missions in the outer solar system. With a reduced amount of useful variable values, it is less likely for the algorithm to identify global optima for a given number of solution candidates, i.e. hindering the effect described in the schema theorem as mentioned in Chapter 5.1.5 (Mahfoud, 1995). At the same time, this reduces the dominance of the mission's global variables and more emphasis is put on the local variables, describing the individual segments.

Second, both *Hayabusa 2'*s and *BepiColombo*'s trajectory were derived using dedicated coasting arcs, and methods that actively use them and use respective optimization schemes. A coast arc is defined by its starting velocity and position vectors and the time allowed for it, i.e. in difference to thrust-arcs of a trajectory coast arcs do not add large numbers of variables. The downside is that they are defined by the starting condition and cannot be changed afterwards. However, the better

Table 7-25: List of orbit parameters of the target bodies for *Dawn*, *Hayabusa 2* and *BepiColombo*.

Value/ Body	$1999JU_3$*	Vesta*	Mercury**
Eccentricity	0.1903	0.0892	0.2056
Inclination	5.8839°	7.14°	7.0°
Semi-Major Axis [AU]	1.1896	2.3618	0.3871

*NASA's small body database (https://ssd.jpl.nasa.gov/sbdb.cgi) **(Prussing and Conway, 1998)

results of the optimization effort for the original solution show their usefulness and that the trajectory parts with very small thrust accelerations in the trajectory thrust profiles seen in the previous subchapters are no sufficient substitute.

Third, the optimization calculations have used the same properties as the original missions, especially for flight time, but a smaller number of gravity assists (based on the performance during the previous calculations concerning more than one gravity assist). Large flight times result in worse performing solution candidates and mislead the optimizer to other gravity-assist partners. Smaller flight times, fitting missions with a smaller number of gravity assists produce better results, as evidenced in the calculations dedicated to an Earth-Mercury mission.

The outset of this thesis has been to investigate the usefulness of Tisserand's Criterion and especially its graphical representation, Tisserand graphs, as method for optimization low-thrust missions using gravity-assists. In front of that backdrop, it has to be mentioned that besides the graphs' limitations regarding application in missions with continuous thrust, they are also not usable for missions with target bodies that have a large eccentricity or inclination, especially not asteroids and comets (Vasile and De Pascale, 2006). The respective bodies of the calculations concerning actual missions, each fall into this category, see Table 7-25. Mercury has the largest planetary eccentricity and an inclination of 7°, Vesta's inclination is 7.14° and 1999 JU_3's inclination is close to 6° and it has an eccentricity of almost 0.2. As it has been the goal to find a method comparably easy and quick as Tisserand graphs it is regarded as acceptable that such bodies cannot easily be used as targets for missions that are to be optimized first with the presented method. To further align the perspective on the optimization outcome, especially regarding *BepiColombo's* trajectory, it should be noted that the method presented in this thesis has been intended as a first step, allowing further optimization and evaluation of feasibility, not to create a final mission trajectory. It can therefore not be expected that its results can compete with an actual mission trajectory that has been optimized using multiple tools, specialized in different aspects of the optimization, and in a timeframe of more than a decade ((García Yárnoz and Jehn, 2006) and (Jehn and Langevin, 2016)).

7.12 Tisserand Graph Considerations

The initial idea for optimizing the gravity-assist sequence has been application of Tisserand's Criterion. Also, Chen et al. (Chen et al.,2008) have depicted low-thrust trajectories in Tisserand graph diagrams, representing a single search in a "branch search" (as implicitely executed for one branch). It becomes apparent that especially concerning multi-gravity-assist trajectories, their approach is more successful than the method presented in this thesis in finding trajectories with a Δv below that of a mission without gravity assist, at the expense of universiality of their search. They are not searching for a globally optimal solution, but one solution, which they evaluate a posteriori. By selecting fixed values for thrust and gravity-assist partner, they are able to identify less Δv consuming trajectories than obtained with the method presented in this thesis. While limiting the search to fixed values restricts the chances of finding the actual global optimum, it allows finding a solution. However, one objective of this thesis has been to actually find a gravity-assist partner (or a series) and thus allow identification of non-common solutions. A search with

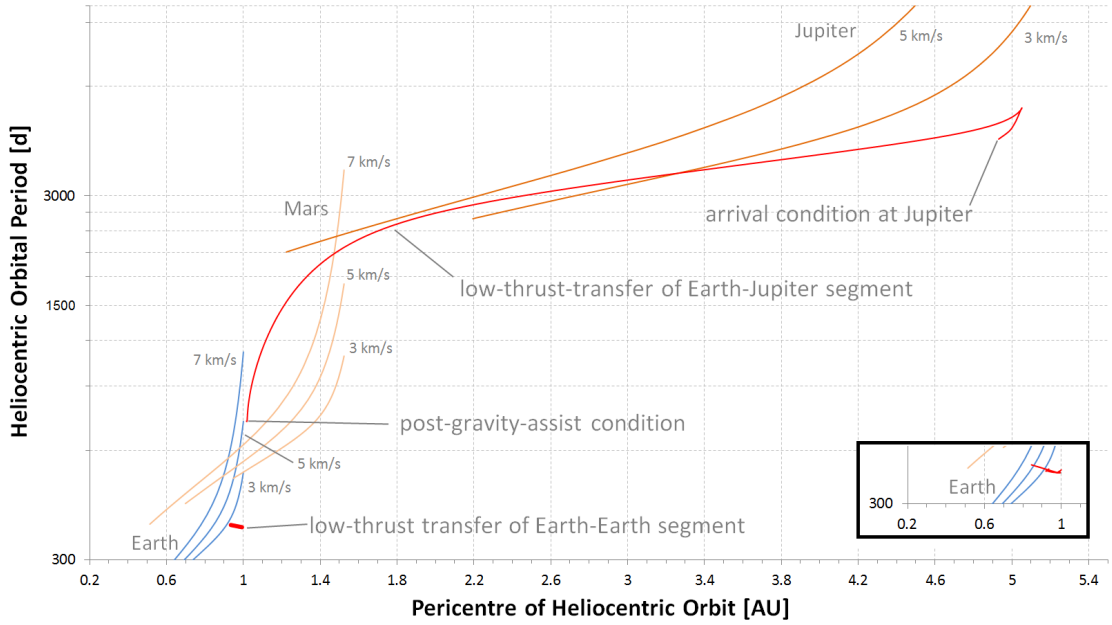

Figure 7-21: The trajectory properties of a low-thrust transfer depicted in a Tisserand graph diagram for the Earth-Jupiter mission seen in Figure 7-5, with a hyperbolic excess velocity of 5.21 km/s at Earth. Two parts of the transfer are visible. Tisserand graphs are shown for Earth, Mars and Jupiter with the designated hyperbolic excess velocities. The transfer from Earth to Earth contains the initial condition (right end of graph) and the final condition before the gravity assist (left). The frame on the right shows the complete history of the properties for the first segment.

branches being created in a similar manner as demonstrated by Chen et al. by by varying the gravity-assist partner could allow reaching this goal, but with the limitation of a stepwise variation of the remaining values with the drawbacks discussed in Chapter 6.

Chen et al. have visualized low-thrust trajectories in a Tisserand graph diagram (using circular planetary orbits). While in this thesis non-circular orbits have been used, it is nevertheless attempted to show a similar depiction. This can be seen in Figure 7-21 for the trajectory shown in Figure 7-5 (with a v_∞ at the gravity assist of 5.2 km/s).

It has to be noted that the graph depicting the first segment of the transfer, i.e. from Earth to Earth occurs from right to left, i.e. the spacecraft's pericentre radius is reduced. It does not connect to the 5 km/s graph. The subsequent shift of pericentre radius and orbital period is visible at the point marked with *post-gravity-assist condition*. Especially at this position it is visible that the Tisserand graphs in this realistic situation should be shifted (to the right) to accommodate for the non-circular planetary orbits. This shift has to occur depending on the actual meeting point of the spacecraft and planet (i.e. the orbital distance of the planet at the encounter date).

The tendencies in this picture fit what is generally to be expected, their precision however is in doubt. The first part of the transfer does not reach up to the point of the respective (ideal) Tisserand graph. At this point it cannot be ascertained fully why this is the case. As mentioned before, the Tisserand graphs for the real planetary orbits will shift to accommodate changes in their own orbital properties with time. This would change the position and form of the Tisserand graphs as depicted in Figure 7-21. In addition the trajectory model used is an approximation. Three aspects have to be noted in that regard.

First, Wall and Conway originally derived the inverse polynomial by a parametric comparison with solutions for a simplified rendezvous problem (e.g. only a fixed flight time and circular target orbit

have been used). They deliberately did not include escape trajectories and neither did they include flyby trajectories containing changes of velocity direction (i.e. deflection due to a gravity-assist) into account (although they apply their model for asteroid flybys). This might attribute to a reduced quality and realism of the solution for such a situation as seen in Figure 7-21.

Second, Wall and Conway also discuss the unusual and unrealistic thrust history associated with trajectories containing less than 3 π rad revolutions about the origin, which is not achieved for the given example trajectory. The behavior, pointing to a more impulsive nature of the thrust, as discussed in (Wall and Conway, 2009) can also be seen in the thrust histories depicted for missions calculated within this thesis, e.g. in Figure 7-5 to Figure 7-11.

Third, to analyse the suitability of the trajectory model after discovering the less than expected realistic properties, a simple transfer from Earth to Earth has been tested. For initial and final condition no hyperbolic excess velocity has been considered and the goal has been a rendezvous. The trajectory model has not been able to determine the trivial solution of not thrusting at all, although the desired target condition, i.e. a relative velocity of 0 has already been fulfilled initially. Results vary between small amounts of Δv (some hundred m/s) and large amounts (some ten thousand m/s). As the initial leg of the trajectory described in Figure 7-21 is also a transfer from Earth to Earth, it is possible that the model is suffering from the same inaccuracies in this case.

Eq. (5-9) describes the trajectory. The trivial solution for the above mentioned case, i.e. a rendezvous transfer from Earth to Earth would be a trajectory identical to Earth's solar orbit without thrusting at all, i.e. for Earth the radius would be almost constant, due to Earth's eccentricity of 0.0167. However, Eq. (5-9) cannot become constant for a progressing θ, only the values for the coefficients can become very close to 0. Despite application of "double" variables in the code, this nonetheless bears the risk of numerical inaccuracies, due to the closeness to 0. These inaccuracies can progress within the calculation and lead to inaccurate results, as shown in Figure 7-21 for the proportion of the transfer close to a circular orbit. These inaccuracies reduce the reliability and usefulness of the presented trajectories and add weight to the recommendation of changing the trajectory model, in addition to the inability of coast arc modeling.

To test this two further calculations have been conducted with the same properties as for the trajectory depicted in Figure 7-5, as given in Table 7-6 (50 population members), but with two modifications:

a. Forcing (via coding) the optimizer to create a trajectory incorporating at least two revolutions for the first segment
b. Increasing the steps size (see Chapter 5.2.2, stage 7.) from 100 to 1000.

The results are summarized in Table 7-26 (column '0' stating the values of the original trajectory). It can be seen that the hyperbolic excess velocity has been reduced for both modifications and the Δv has been increased. The most significant change in coefficients can be see in the coefficient b of the respective first legs, where the order of magnitude is a factor of 10 larger for the trajectory with a gravity assist at Mars. This could have repercussions on the coefficients d to g as b is part of the equations to determine these coefficients (see Eq. ((5-16) resp. Eq. (5-19)). Similarly the coefficients for the second segment (the transfer from the gravity-assist partner to Jupiter) has coefficients factor 10 larger than the respective first leg. The trajectories for both modifications are depicted in Figure 7-22 and the Tisserand graph diagram is shown in Figure 7-23 for modification a and in Figure 7-24 for modification b. In Figure 7-23 the shift caused by the gravity assist is less severe than for the original solution (Figure 7-21), although the final condition before the gravity assist is still not touching a Tisserand graph. The decrease in the mismatch to the expected picture is fitting the larger diversion from a circular orbit for the trajectory presented in Figure 7-22 (left) and the original trajectory in Figure 7-5, but the trajectory is still close to a circle.

Table 7-26: Summary of the outcome for modified optimization for Earth-Jupiter rendezvous transfer (a: forcing minimum 2 revolutions for the first segment, b: increasing the step size from 100 to 1000) and comparison with the original trajectory (option 0). All trajectories have one gravity assist.

Parameter/Calculation	a	b	0
$\Delta v_{mission}$	19,832 m/s	15,209 m/s	12,470 m/s
$LD_{mission}$	56858	56353	56097
$ToF_{mission}$	2672 d	3331 d	3177 d
$N_{rev,mission}$	3	2	2
GA Partner	Earth	Mars	Earth
v_∞	4063 m/s	2167 m/s	5211 m/s
$ToF_{1st\ leg}$	751 d	760 d	413 d
$N_{rev,1st\ leg}$	2	1	1
Coefficients 1st leg	a: 0.981768924055507	a: 1.00894877000302	a: 0.982098159405825
	b: 0.00259204462911067	b: -0.017074962052682	b: -0.00222859701317472
	c: 0.00638912911111542	c: -0.00407261067349965	c: 0.00621707878845862
	d: 0.00612973816258259	d: -0.0071604102021672	d: 0.0038377095396913
	e: -0.00200471046589204	e: 0.0017111863416211	e: -0.0023812916566918
	f: 0.000185985102779659	f: -0.000113304268224268	f: 0.000374202917520971
	g: -0.000005463331306713	g: 0.000001629024763565	g: -0.000018494761153966
Coefficients 2nd leg	a: 0.985717972995677	a: 0.685265342246449	a: 0.984067721671785
	b: 0.0471725493787239	b: 0.0560099042899776	b: 0.00825295859272204
	c: -0.0499409315003193	c: -0.0570259673070226	c: -0.128654784135837
	d: -0.0207822030756614	d: -0.00376426650974786	d: -0.00181783325977502
	e: 0.00828633500410758	e: 0.00385944177923437	e: 0.0122662837031488
	f: -0.00088854802996852	f: -0.000498485726113357	f: -0.00220569142965493
	g: 0.0000297379042197855	g: 0.000019721953498491	g: 0.00011658188715424

This is different for the calculation concerning modification b. Figure 7-22 (right) shows that the deviation from a circle is significant, due to the flyby at Mars (semi-major axis of 1.524 AU). Corresponding, the Tisserand graph diagram fits more what is to be expected, although the fit is not exact due to the real Mars orbit properties vs. an ideal circular Mars orbit. In general the lack of plausibility concerning the Tisserand graph diagram occurs for the first leg only, which fits that for the second leg, the respective coefficients are factor 10 larger than those of the first leg.

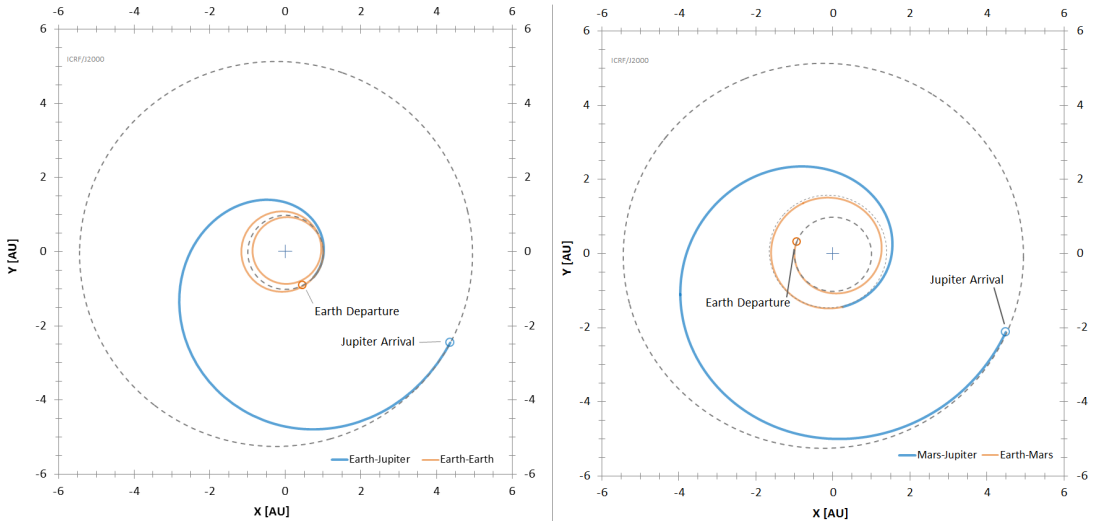

Figure 7-22: The trajectory for a two revolution first segment (modification a) with an Earth gravity assist (left) and for the increased step size (modification b) with a Mars gravity assist (right). The coordinates are in the International Celestial Reference Frame and scaled in Astronomical Units (AU).

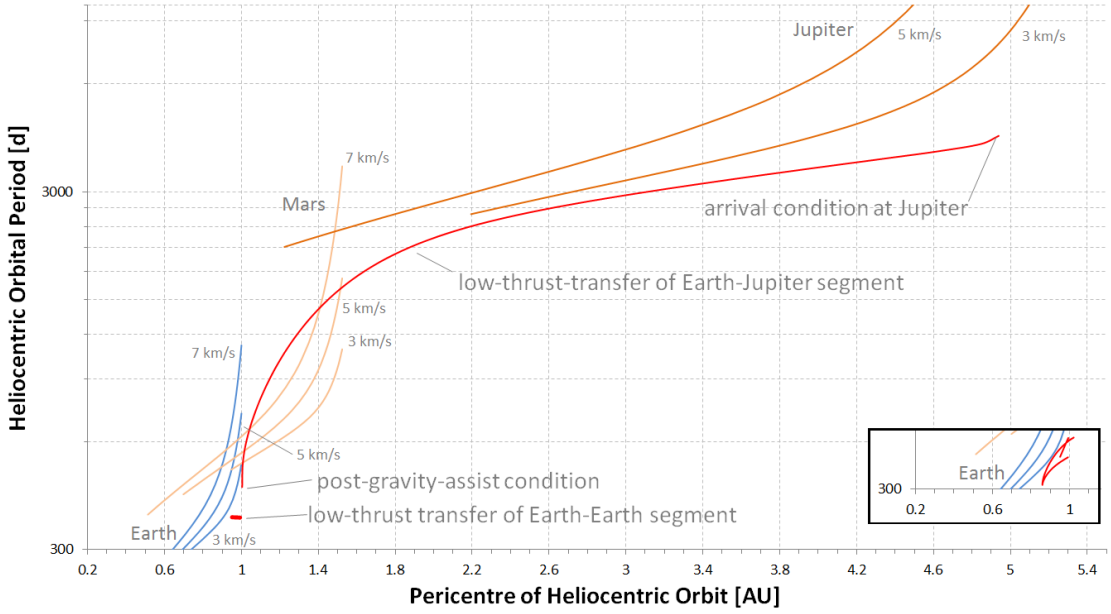

Figure 7-23: The trajectory properties of a low-thrust transfer depicted in a Tisserand graph diagram for the Earth-Jupiter mission with a forced number of revolutions of 2 for the first segment (modification a). Tisserand graphs are shown for Earth, Mars and Jupiter with the designated hyperbolic excess velocities. The transfer from Earth to Earth contains the initial condition (right end of graph) and the final condition before the gravity assist (left end) with a v_∞ of 4 km/s. The frame on the right shows the complete history of the properties for the first segment.

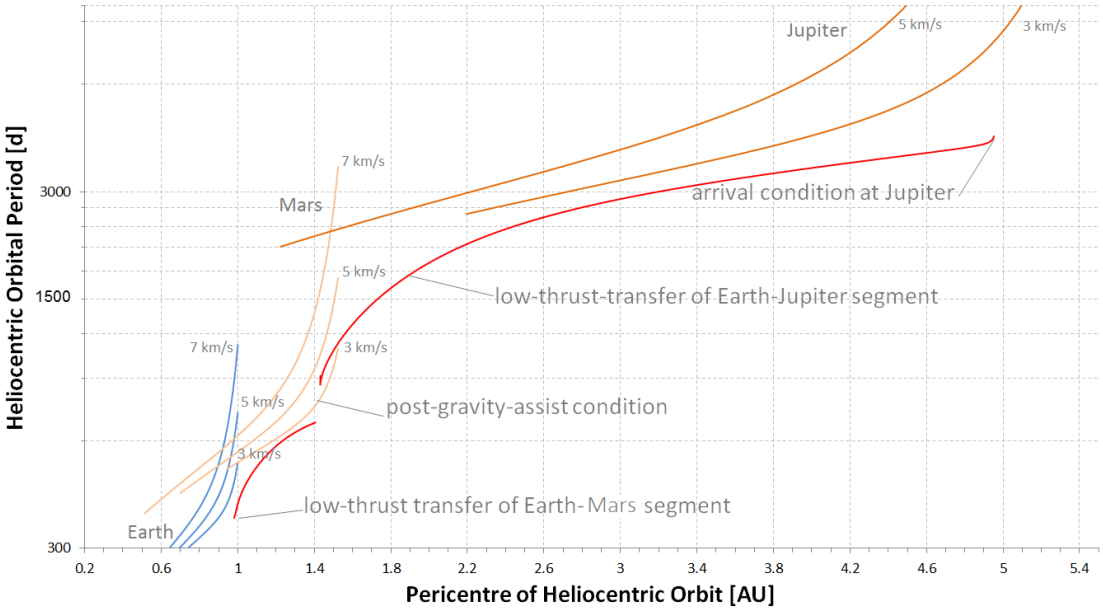

Figure 7-24: The trajectory properties of a low-thrust transfer depicted in a Tisserand graph diagram for the Earth-Jupiter mission with a step number of 1000 (modification b). Tisserand graphs are shown for Earth, Mars and Jupiter with the designated hyperbolic excess velocities. The gravity assist at Mars occurs with a hyperbolic excess velocity of 2 km/s.

The number of modified calculations is not sufficient to make general statements, especially not concerning why the modification b produced a Mars gravity-assist trajectory instead of Earth (it is likely only a variation as occurred before for 50 population members, see Table 7-8). However, a tendency can be seen that near-circular trajectories result in a less plausible progression of the

orbital period and pericentre distance in a Tisserand graph diagram than a transfer to Mars, with a less circular transfer trajectory.

As the trajectory model's drawbacks concerning coast arcs and the corresponding inability of adequately calculating existing trajectories (see Chapter 7.7 to 7.9 and Chapter 7.11.3) already mandate a change of the trajectory model for future application of the method, the issues concerning Tisserand graphs and the implausible trajectory behavior for near-circular trajectories are not further investigated in this thesis.

While other shape-based models exist and have been used to calculate gravity-assist low-thrust trajectories with *pre-determined* partner sequences (see Chapter 2.4.2), they are "computationally slow" (Wall and Conway, 2009) and produce trajectories, which are "far from being optimal" (Wall and Conway, 2009).

Figure 7-21 presents another noteworthy issue. The graph depicting the second mission segment, from Earth to Jupiter, crosses one of Jupiter's Tisserand graphs (note again that this graph represents the case of a circular orbit). While this appears to indicate an opportunity for a gravity assist, matters are less straightforward. First, the phasing is not regarded in the graph, i.e. to achieve a gravity assist at Jupiter, Jupiter's actual position has to be included. As per diagram, the orbital period of the spacecraft is more than 8 years, Jupiter's is approximately 11.8 years, i.e. a significant amount of time could be required to actually meet Jupiter, even when coasting. However, coasting has been identified as lacking ability for the presented method. Second, this crossing point has been obtained for a trajectory optimized for a different final condition, i.e. it is unlikely that it is also optimal for a further gravity assist at Jupiter and a subsequent rendezvous. The goal of the work presented in this thesis however has been to not just find *a* solution, but a (near) optimal. Exploiting this crossing point would be identical of calculating the correction term in Eq. (4-81) with the calculated values for the velocity and thrust acceleration and derive a gravity-assist opportunity with that solution. However, the product of velocity and acceleration are optimized for a solution not containing such a gravity assist and thus including this crossing point in the consideration would be diverting from the optimal solution. While depending on the quality of the previous search this could still lead to an improvement (especially considering the lack of success for applying the presented method to multi-gravity-assist solutions), this cannot be generalized.

7.13 Open Issues and Outlook

For testing the overall approach, the trajectory model has been limited to two dimensions; analogously to the restrictions of the Tisserand graphs (see Chapter 2.3). It has been regarded as an approach sufficient for analysis of this thesis' method's suitability as well as the trajectory model's suitability. As planets are typically the gravity-assist partners (including missions with multiple gravity assists) with only small inclination differences between partners, this limitation is not severe when aiming to analyse sequences of such flybys. Only the last gravity assist would change the inclination significantly in case a mission is targeted outside the ecliptic. Limiting the perspective to the ecliptic naturally rules out missions targeted outside the ecliptic. Thus, extending the method to three dimensions allows analyzing a more complete and thus realistic catalogue of mission scenarios. This extension can be achieved by adapting the shape-based trajectory model to three dimensions, which will enable the inclusion of inclination changes into the evaluation of the mission candidates.

At the same time with the third dimension, more variables are introduced, e.g. another turning angle (one angle for changes parallel to the ecliptic and one for perpendicular changes) and e.g. one vector component for velocity. This increase in variables will decrease the search success rate,

especially since they are also real valued, similar as the increase in variables for a multi-gravity-assist mission has reduced the optimization quality. A similar effect is to be expected for including the non-tangential thrust in the optimization.

Strategies of constraining e.g. the inclination change of a gravity assist into likely useful regions, could improve the search quality. A mission targeting Jupiter is likely not improved by a large increase of inclination in regards to the ecliptic, especial not if further planetary flybys are intended. A trajectory leading away from the ecliptic will also lead away from further planets as gravity-assist partners. Therefore, constraining the change of direction perpendicular to the ecliptic (or another reference plane) can be used to remove likely non-beneficial candidates. On the other hand extending the optimization into three dimensions would allow application in scenarios, which involve targets with a significant inclination, e.g. asteroids, dwarf planets or an application in the Jupiter or Saturn system when using its more inclined moons as gravity-assist partners.

Besides further investigating usage of the Differential Evolution, e.g. by modifying properties like the mutation rate of the solution candidates, other algorithms can be investigated, possibly a combination of several algorithms could be used. While Simulated Annealing did not proof very useful for this kind of search, others might provide better results, e.g. Swarm Optimization or Threshold Accepting. In addition, other replacement schemes and more variations of the algorithm properties could be investigated. Similarly, certain algorithms might produce better results for different mission scenarios – depending on the respective search space topology. Further investigating the search space topology of various typical mission scenarios, e.g. targeting Mercury or Mars, might provide indications of suitability of specific algorithms and provide information for other mission design methods.

Eventually, with sufficient understanding of the search space a two-step optimization as described in the section 7.4.3 could be implemented, if considered useful. A thorough analysis of the variables labeled as local for this current method could provide information about patterns of the search space topology.

In addition, a way of including dedicated coasting arcs for optimization and to regard operational constraints as for *BepiColombo* (see Chapter 7.9) could be included. This would also allow actual application of Tisserand graphs for portions of the mission without thrust. In any case a different trajectory model should be employed for future work, due to the limitations of the shape-based model presented within this work concerning coast arcs and realism.

Besides the before mentioned updates, a combination of an optimizer for gravity-assist sequences and radiation models would allow finding trajectories, that provide a trajectory optimized based on minimum radiation exposure and with the help of gravity assists still keep a reasonable amount of Δv. This would especially be interesting for missions involving Jupiter's moons, as Jupiter's vicinity is coined by its radiation. Similarly, Earth's vicinity is subject to radiation in the Van-Allen Belts and missions involving low-thrust gravity assists at the Moon could also be optimized regarding minimal radiation obtained on a trajectory.

8 Summary and Conclusion

The present work has investigated the gravity-assist optimization for low-thrust trajectories under the application of an energy relation called Tisserand's Criterion resp. its visual representation in the form of Tisserand graphs.

Low-thrust missions benefit from the large specific impulse of low-thrust engines (typically some 1000 s). At the same time, gravity-assist maneuvers have proven to be an enabling technique for many mission designs, including missions like *Cassini, Voyager 1* and *2, Dawn*, *Hayabusa* and *Hayabusa 2*. It is therefore a reasonable intention to combine these two options for mission design improvement and further investigate and evaluate the possibility to optimize such trajectories in with both aspects being part of the optimization process.

Especially early mission design studies as conducted in the Concurrent Engineering Facility of the German Aerospace Center (DLR) rely on fast and robust mission analysis and thus trajectory optimization. It is not possible to spend long computation times during such mission design studies.

However, present methods of optimizing low-thrust gravity-assist trajectories focus exclusively on the low-thrust optimization. Methods of including the sequence of gravity-assist partners in the optimization have not been published yet. This thesis aimed to close that gap.

In this work, it has been shown that an energy-based method applying Tisserand's Criterion in so-called Tisserand graphs is currently prominently used for gravity-assist sequencing for missions involving impulsive propulsion maneuvers. However, as Tisserand's Criterion is based on the circular restricted three-body system and most importantly the premise that gravitation is the only acting force in the system, a direct transfer of this method to apply it for continuous low-thrust is not possible. The very nature of low-thrust means that for a considerable amount of mission time the spacecraft is thrusting, i.e. the thrust force is acting on the spacecraft in addition to the gravitational forces.

As a first step in this work, the magnitude of deviation regarding Tisserand's Criterion has been analysed for a low-thrust scenario. Such an analysis has not been published before. To understand the magnitude and therefore the impact of the applied thrust it has further been related to errors that naturally occur due to discrepancies between the circular restricted three-body problem and the actual solar system, i.e. the theoretical case Tisserand's Criterion has been derived for and the practical case where it is used.

With the help of sample calculations and simulations it has been shown that errors in the value of the Tisserand's Criterion occur mostly only with a magnitude of some percent. For example, the error introduced due to a small distance between the second mass (i.e. the gravity-assist partner) and the spacecraft has in extreme cases a magnitude of about 1.5%, i.e. it is not significant.

The thrust force and gravitational force have further been related to each other for the case of a typical mission, in this case based on *Dawn*. It has been shown that for a realistic case the gravity force of the Sun usually exceeds the thrust magnitude by several ten thousand and even in the most favoring situation for thrust, does not reach a smaller ratio than factor 100 up to a solar distance of 20 AU. Consequently, it has been concluded that in the balance of forces the thrust is inferior to the gravitational force.

The largest "natural" error in Tisserand's Criterion occurs due to the non-circular orbits of the potential gravity-assist partners, as has been shown in this thesis. For numerical simulations, two solar system models have been used to calculate the effect of a gravity assist on a given spacecraft and the resulting value of Tisserand's Criterion has been compared. Due to the nature of the situation only part of the infinite solution space has been sampled for the simulations, but even in this part a discrepancy between the respective values of Tisserand's Criterion of up to 25% occurred. Therefore, showing that neglecting the non-circular orbits of the planets can have a considerable effect on the magnitude of the value of Tisserand's Criterion, which consequently also influences the usage of Tisserand's Criterion for sequencing gravity-assist missions.

Since Tisserand's Criterion is an energy relation, however the energy change due to thrust has been considered more relevant for this analysis. The simple fact that low thrust has an enabling quality for missions i.e. transfers a spacecraft from one orbit to another, even for interplanetary missions like *Dawn*, already shows that the energy contribution is not negligible. The change of orbits is tantamount to a change of specific energy. In difference to impulsive missions, this change of energy is continuously occurring. Therefore, the thrust should not be disregarded, when applying Tisserand's Criterion.

Thus, as a next step a correction term has been derived, which modifies Tisserand's Criterion, based on the energy input caused by thrust. Similar corrections have not been published before outside the frame of this thesis. The disadvantage of this term is that it is not a state quantity and needs to be integrated over time to receive a correct value for the modified Tisserand's Criterion. However, with this term it is possible to extend the usage of Tisserand's Criterion to missions under (low) thrust and even consider the effect of outgassing of comets on their trajectory, if Tisserand's Criterion is used for cometary observation, a purpose, which it has originally been derived for.

Reviewing existing benchmarks and low-thrust trajectory optimization, three algorithms have been selected for further usage with the method presented in this work: two evolutionary algorithms and random search as benchmark. Their suitability has been investigated via numerical experimentation. To further facilitate the optimization process of the gravity-assist sequence, methods to trim the solution space have been investigated and two have been presented in this work:

1. Restricting trajectories to areas with the largest Δv change, and
2. restricting the pool of possible gravity-assist partners.

To combine trajectory and sequence optimization, an objective function has been used, which is based on mission Δv, where necessary penalized for violations of maximum allowed thrust and minimum allowed distance to the central body. To address the necessary phasing of the respective bodies shape-based representations have been used, which allow quickly finding sufficiently accurate solutions, by assuming a polynomial formulation for a trajectory.

Subsequently, a method has been derived that calculates the mission legs in a stepwise manner and then optimizes all mission candidates, but also optimizes non-gravity-assist trajectories (which is quickly achievable via the shape-based method) as benchmarks.

This method has been based on the idea of forcing the gravity assist into the optimization by defining each leg of the trajectory by its initial and final body. The actual body is a control variable for the optimization and thus allows the finding of sequences of GA partners. The definition of trajectory segments as being located between two bodies and the usage of the fast shape-based trajectory model allows the evaluation of a large number of gravity-assist trajectories, resp. sequences. This leads to a thorough search through the search space. By including gravity-assist partners as variables in the search, it is also unbiased by the mission analyst's experience.

Furthermore, it has been shown that, while not unconditionally true, evolutionary strategies can be exploited when optimizing the mission sequence with the proposed method. For an Earth to Jupiter mission example, Differential Evolution could produce improvements of up to 25% for the selected objective function and was also able to reliably identify an optimal gravity-assist partner and encounter date. For a mission example from Earth to Mercury, the improvement reached up to 13.5%, although not a single, but three gravity-assist opportunities have been identified. From the three initially considered search patterns, as originally anticipated, the sequential path was taken, i.e. GA and trajectory optimization happen stepwise, resp. for each trajectory segment. Using a population for optimization has shown to be valuable, which is attributed to an increased chance of retaining beneficial information of variable settings and thus beneficial regions of the search space.

The method should be improved in reliability and performance regarding multi-gravity-assist missions and regarding inner-solar-system missions, as evidenced by its inability to reproduce results as given for the missions *Hayabusa 2* and *BepiColombo*, which each had prescribed gravity-assist partners. Once such an improvement is achieved, it should also be enhanced to incorporate all three dimensions or e.g. a radiation model. Further calculations can also allow improving the understanding of the search space topology and thus evaluation of algorithm suitability.

Comparison of examples of the calculated trajectories has shown a mismatch of the transfer trajectories to the expected behavior in Tisserand graph diagrams. This is tentatively explained by numerical issues caused by the trajectory model for near-circular transfer trajectories, limiting the quality of the results further and adding weight to the need of changing the trajectory model for future work. The results obtained with this method incorporating the complete search space have also not exceeded the quality of the results obtained by a simple trial and error search using Tisserand graphs, which means that the shape-based model is of limited suitability for the problem addressed in this thesis. Other trajectory models should be considered in the future.

In der vorliegenden Arbeit sind Ergebnisse enthalten, die im Rahmen der Betreuung folgender studentischer Arbeiten entstanden sind
(the present work contains results obtained via the supervision of the following student theses):

Hansen, H.; *Analysis about Constraints for Low-Thrust Gravity-assist Sequencing*, University of Bremen, 2017

Herrmann, M.; *Recherche und Bewertung von Formbasierten Verfahren im Bereich der Niedrigschubbahnoptimierung*, University of Bremen, 2011

Kuntikal Doddi Mahadevaiah, N.; *Analysis and Evaluation of Algorithms for Low-Thrust Gravity-Assist Sequencing*, University of Bremen, 2017

Srongprapa, P.; *Modelling of Three-Body System in C++ for Tisserand Error Estimation*, Technical University of Delft, 2015

References

Accomazzo, A., Ferri, P., Hubault, A., Lodiot, S., Pellon-Bailon, J.L., Porta, R.; *Rosetta visits asteroid (21) Lutetia*, Acta Astronautica Vol. 72, pp. 178-184, March-April 2012

Alles, W.; *Raumflugmechanik II*, Lecture Script, Chair of Flight Dynamics, RWTH Aachen University, 3rd Edition, 2005

Ashlock, D.; *Evolutionary Computation for Modelling and Optimization*, Springer, 2006

Bakich, M.E., *The Cambridge Planetary Handbook*, Cambridge University Press, 2000

Bate, R.R., Mueller, D.D., White, J.E.; *Fundamentals of Astrodynamics*, Dover Publication, 1971

Benkhoff, J., van Casteren, J., Hayakawa, H., Fujimoto, M., Laakso, H., Novara, M., Ferri, P., Middleton, H. R., Ziethe, R.; *BepiColombo – Comprehensive exploration of Mercury: Mission overview and science goals*, Planetary and Space Science, Vol. 58, pp. 2-20, 2010

BIPM (Bureau International des Poids et Mesures); *The International System of Units*; 8th Edition, 2006

Brise, Y.; *Lipschitzian Optimization, DIRECT Algorithm, and Applications*, Presentation, Institute of Theoretical Computer Science, ETH Zürich, 2008

Bronstein, I.N., Semendjajew, K.A., Musiol, G., Mühlig, H.; *Taschenbuch der Mathematik*, Harri Deutsch Verlag, 5th Edition, 2000

Carnelli, I.; *Optimization of Interplanetary Trajectories combining Low-Thrust and Gravity Assists with Evolutionary Neurocontrol*, master thesis, Politecnico di Milano Facolt`a di Ingegneria Dipartimento di Ingegneria Aerospaziale, 2005

Carnelli, I., Dachwald, B., Vasile, M.; *Evolutionary Neurocontrol: A Novel Method for Low-Thrust Gravity-Assist Trajectory Optimization*, Journal of Guidance, Control, and Dynamics, Vol. 32, No. 2, March–April 2009

Chen, K.J., Kloster, K.W., Longuski, J.M.; *A Graphical Method for Preliminary Design of Low-Thrust Gravity-Assist Trajectories*, AIAA/AAS Astrodynamics Specialist Conference and Exhibit, 18-21 August, Honolulu, Hawaii, 2008

Conway, B.A. (editor); *Spacecraft Trajectory Optimization;* Cambridge University Press, 2010

Crain, T., Bishop, R.H., Fowler, W.; *Interplanetary Flyby Mission Optimization Using a Hybrid Global/Local Search Method*, Journal of Spacecraft and Rockets, Vol. 37, No. 4, July–August 2000

Dachwald, B.; *Low-Thrust Trajectory Optimization and Interplanetary Mission Analysis Using Evolutionary Neurocontrol*, Dissertation, Universität der Bundeswehr München, 2003

Debban, T.J., McConaghy, T.T., Longuski, J.M.; *Design and Optimization of low-thrust gravity-assist trajectories to selected planets*, Astrodynamics Specialist Conference and Exhibit, 5-8 August 2002, Monterey, California

Diniega, S., Sayanagi, K.M., Balcerski, J., Carande, B., Diaz-Silva, R.A., Fraeman, A.A., Guzewich, S.D., Hudson, J., Nahm, A.L., Potter-McIntyre, S., Route, M, Urban, K.D., Vasisht, S., Benneke, B., Gil, S., Livi, R, Williams, B., Budney, C.J., Lowes, L.L., *Mission to the Trojan asteroids: Lessons learned during a JPL Planetary Science Summer School mission design exercise*, Planetary and Space Science, Volume 76, pp. 68-82, February 2013

Dueck, G., Scheuer, T.; *Threshold Accepting: A General Purpose Optimization Algorithm Appearing Superior to Simulated Annealing*, Journal of Computational Physics, Vol. 90, pp. 161-175, 1990

El-Beltagy, M.A., Keane, A.J.; *A comparison of various optimization algorithms on a multilevel problem*, Engineering Applications of Artificial Intelligence 12, pp. 639 – 654, 1999

Flandro, G.A., *Fast Reconnaissance Missions to the Outer Solar System Utilizing Energy Derived from the Gravitational Field of Jupiter*, Acta Astronautica, Vol. 12, No. 4, pp. 329-337, 1966

García Yarnoz, D.G., Jehn, R., Croon, M.; *Interplanetary navigation along the low-thrust trajectory of BepiColombo*, Acta Astronautica, Vol. 59, pp. 284-293, 2006

Goldberg, D.E.; *Genetic Algorithms in search, optimization and machine learning*, Addison-Wesley, 1989

Gurfil, P., Kasdin M.J.; *Characterization and design of out-of-ecliptic trajectories using deterministic crowding genetic algorithms*, Computer Methods in Applied Mechanics and Engineering, Vol. 191, pp 2141–2158, 2002

Hartmann, J.W., Coverstone-Carroll, V., Williams, S.N.; *Optimal Interplanetary Spacecraft Trajectories via a Pareto Genetic Algorithm*, The Journal of the Astronautical Sciences, Vol. 46, No. 3, pp.267-282, 1998

Heaton, A.F., Strange, N.J., Longuski, J.M., Bonfiglio, E.P.; *Automated Design of the Europa Orbiter Tour; Journal of Spacecraft and Rockets*, Vol. 39, No. 1, pp 17 -22, January-February 2002

IAU (International Astronomical Union); *Resolution B2 – on the redefinition of the astronomical unit of length;* XXVIII General Assembly of International Astronomical Union, 2012

Jehn, R., Companys, V, Corrall, C, García Yárnoz, D, Sánchez, N.; *Navigating BepiColombo during the weak-stability capture at Mercury*, Advances in Space Research, Vol. 42, pp. 1364-1369, 2008

Jehn, R., Schoenmaekers, J.; *BepiColombo trajectory options to Mercury in 2016 and 2017*, 24th International Symposium on Spaceflight Dynamics, 5-9 May, Laurel, Maryland, USA, 2014

Jehn, R., Langevin, Y.; *BepiColombo trajectory options to Mercury in 2018 and 2019*, 26th AAS/ AIAA Space Flight Mechanics Meeting, 14-18 February, Napa, California, USA, 2016

Kemble, S.; *Interplanetary Mission Analysis and Design*; Springer, 2006

Kim, M.; *Continuous Low-Thrust Trajectory Optimization: Techniques and Applications*, Dissertation, Virginia State University, 2005

Kloster, K.W., Petropoulos, A.E., Longuski, J.M.; *Europa Orbiter Tour Design with Io Gravity-assists*, Acta Astronautica, Vol. 68, pp. 931-946, 2011

Labunsky, A.V., Papkov, O.V., Suhkanov, K.G.; *Multiple Gravity-assist Interplanetary Trajectories*, 1st ed. Gordon and Breach Science Publishers, 1998

Löb, H., Schartner, K., Seboldt, W., Dachwald, B., Ströppel, J., Ohndorf, A., *CONSEP Contributions to Solar-Electric Propulsion Final Report*, 2007

Mahfoud, S.W.; *Niching Methods for Genetic Algorithms*, Doctoral Thesis, University of Illinois, 1995

Maiwald, V.; *Applicability of Tisserand's Criterion for Optimization of Gravity-assist Sequences for Low-thrust Missions*, 66th International Astronautical Congress (IAC), 12-16 October, Jerusalem, Israel, 2015

Maiwald, V.; *About Combining Tisserand Graph Gravity-assist Sequencing with Low-thrust Trajectory Optimization*, 6th International Conference on Astrodynamics Tools and Techniques (ICATT), 14-17 March, Darmstadt, Germany, 2016

Maiwald, V.; *A New Method for Optimization of Low-Thrust Gravity-assist Sequences*, CEAS Space Journal, 9(3), pp. 243-256, DOI :10.1007/s12567-017-0147-7, 2017, a

Maiwald, V.; *Initial Results of a New Method for Optimizing Low-Thrust Gravity-Assist Missions*, 31st International Symposium of Space Technology and Science, 3-9 June, Matsuyama, Japan, 2017, b

McConaghy, T.T., Debban, T.J., Petropoulos, A.E., Longuski, J.M.; *Design and Optimization of low-thrust with gravity assist*, Journal of Spacecraft and Rockets, Vol 40, No. 3, pp 380 – 387, 2003

Mengshoel, O.J., Goldberg, D.E.; *The Crowding Approach to Niching in Genetic Algorithms*, Evolutionary Computation, Vol. 16, No. 3, pp. 315 – 354, 2008

Messerschmid, E., Fasoulas, S.; *Raumfahrtsysteme – Eine Einführung mit Übungen und Lösungen*, 2nd Edition, Springer, 2005

Miller, J.K., Weeks, C.J.; *Application of Tisserand's Criterion to the Design of Gravity-assist Trajectories*, AIAA/AAS Astrodynamics Specialist Conference and Exhibit, August 5-8, Monterey, USA, 2002

Mühlenbein, H., *Parallel genetic algorithms in combinatorial optimization*, Computer Science and Operations Research: New Developments in Their Interfaces, 8 - 10 January 1992

Murray, C.D., Dermott, S.F.; *Solar System Dynamics*, Cambridge University Press, 2008

Myatt, D.R., Becerra, V.M., Nasuto, S.J., Bishop, J.M.; *Advanced Global Optimization for Mission Analysis and Design*, Ariadna report id: 03/4101, 2004

NASA; *Dawn Launch Mission to Ceres and Vesta Press Kit*, 2007

NIST (National Institute of Standards and Technology); *Newtonian constant of gravitation*; https://physics.nist.gov/cgi-bin/cuu/Value?bg; accessed 26th July 2015

Nocedal, J., Wright, S.J.; *Numerical Optimization*, Second Edition, Springer, 2006

Nourani, Y., Andresen, B.; A comparison of simulated annealing cooling strategies, Journal of Physics A: Mathematics and General, 31, p. 8373-8385, 1998

Oberth, H.; *Ways to Spaceflight*; NASA TT F-622, Translation of "Wege zur Raumschiffahrt", 1970

Petropoulos, A.E., Kowalkowski, T.D., Vavrina, M.A., Parcher, D.W., Finlayson, P.A., Whiffen, G.J., Sims, J.A.; *1st ACT global trajectory optimisation competition: Results found at the Jet Propulsion Laboratory*, Acta Astronautica Vol. 61, No. 9, pp. 806-815, November 2007

Pirlot, M.; *General local search methods*, European Journal of Operational Research, 92, p.493-511, 1996

Prussing, J.E., Conway, B.A.; *Orbital Mechanics*, Oxford University Press, 1993

Rayman, M.D., Fraschetti, T.C., Raymond, C.A., Russel, C.T.; *Dawn: A mission in development for exploration of main belt asteroids Vesta and Ceres*, Acta Astronautica Vol. 58 No. 11, pp 605-616, June 2006

Rayman, M.D., Mase, R.A.; *The second year of Dawn mission operations: Mars gravity assist and onwards to Vesta*, Acta Astronautica Vol. 67 No. 3-4, pp 483-488, August-September 2010

Russel, S., Norvig, P., *Artificial Intelligence – A Modern Approach*, Prentice Hall, Pearson Education, Second Edition, pp. 111-115, 2003

Settea, S., Boullart, L.; *An implementation of genetic algorithms for rule based machine Learning*, Engineering Applications of Artificial Intelligence, Vol. 13, pp 381 – 390, 2000

Sims, J.A., Flanagan, S.N.; *Preliminary Design of Low-thrust Interplanetary Missions*, AAS/ AIAA Astrodynamics Specialist Conference, August 1999

Storn, R., Price, K.; *Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces*, Journal of Global Optimization, Vol. 11, pp 341-359, 1997

Strange, N.J., Longuski, J.M.; *Graphical Method for Gravity-assist Trajectory Design*, Journal of Spacecraft and Rockets, Vol. 39, No. 1, pp 9 -16, January-February 2002

Sutton, G., Biblarz, O., *Rocket Propulsion Elements*, Wiley-Interscience, 2001

Tsuda, Y., Saiki, T., Ogawa, N., Morimoto, M.; *Trajectory Design for Japanese New Asteroid Sample Return Mission Hayabusa-2*, 23rd ISSFD, Pasadena, California, USA, Oct 29- Nov. 2 2012

Tsuda, Y., Yoshikawa, M., Abe, M., Minamino, H., Nakazawa, S.; *System design of the Hayabusa 2 – Asteroid sample return mission to 1999 JU3*, Acta Astronautica, Vol. 91, pp 356-362, 2013

Tsuda, Y., Kato, T., Shiraishi, M., Matsuoka, M.; *Trajectory Navigation and Guidance Operation Toward Earth Swing-By of Asteroid Sample Return Mission "Hayabusa2"*, 25th International Symposium of Spaceflight Dynamics, Munich, Germany, 19 to 23 Oct., 2015

Vasile, M., De Pascale, P.; *Preliminary Design of Multiple Gravity-Assist Trajectories*, Journal of Spacecraft and Rockets, Vol. 43, No. 4, July – August, pp. 794-805, 2006

Vasile, M., Campagnola, S.; *Design of Low-Thrust Multi-Gravity Assist Trajectories to Europa*, Journal of the British Interplanetary Society, Vol. 62, p. 15 – 31, 2009

Vasile, M., Minisci, E., Locatelli, M; *Analysis of Some Global Optimization Algorithms for Space Trajectory Design*, Journal of Spacecraft and Rockets, Vol. 47, No. 2, March – April 2010, pp. 334-344

Venter, G.; *Review of Optimization Techniques*, Encyclopedia of Aerospace Engineering, Editor: R. Blockley, W. Shyy, John Wiley and Sons, Ltd., 2010

Wakker, K.F.; *Astrodynamics – I*, Lecture Script, Technical University of Delft, August 2007

Wall, B.; Conway, B.; *Shape-Based Approach to Low-Thrust Rendezvous Trajectory Design*, Journal of Guidance, Control, and Dynamics; Vol. 32, No. 1, 2009

Wall, B.; Novak, D.; *A 3D Shape-Based Approximation Method For Low-Thrust Trajectory Design*; Astrodynamics 2011, AAS/AIAA Astrodynamics Specialist Conference, Girdwood, Alaska, USA, July 31-August 4, 2011

Weise, T.; *Global Optimization Algorithms – Theory and Application*; Ebook, www.it-weise.de (Associate Professor, University of Science and Technology of China), 2008

Yam, C.H., McConaghy, T.T., Chen, K.J., Longuski, J.; *Design of Low-Thrust Gravity-assist Trajectories to the Outer Planets*, 55th International Astronautical Congress, Vancouver, Canada, 4-8 October 2004

Yang, D., Xu, B., Lei, H.L., *Multi-objective Detection Trajectory Optimization Design In Solar System*, 64th International Astronomical Congress (IAC), Beijing, China, 23-27 September, 2013

Zhao, Z., Shang, H., Cui, P., Huang, X.; *Low Thrust Gravity Assist Optimization Based on Hybrid Method*, 2012 IEEE fifth International Conference on Advanced Computational Intelligence(ICACI), Nanjing, Jiangsu, China, 18-20 October, 2012

Appendix

1. Code Description and Manual

The code for GOLT is divided into two parts. The first contains the trajectory model based on the shape-based methods by Wall and Conway (2009). The second is the search method developed within this thesis, using the trajectory model to quickly asses the planets' phasing and transfer of the spacecraft.

The shape-based method has been coded on basis of the above quoted reference and the whole program is controlled via an input file, containing relevant information about the to be optimized problem. The results of the optimization are recorded in a "results" file, as well as a csv-file for easier printing in plotting software.

The input file contains information about the intended mission, such as the starting body and target body but also about the optimization as such, e.g. the number of maximum generations. The parameters contained in the input file are listed in the following:

- Start_Body: Body at which the spacecraft launches
- End_Body: Target of the mission
- R_Min: Minimum allowed distance to barycenter (AU)
- Launch_Date: Minimum launch date for the mission (MJD)
- Launch_Frame: Extension of launch window (days)
- Flight_Time_Max: maximum allowed flight time (d)
- Flight_Time_Min: Minimum allowed flight time (d)
- Accuracy: As described by Wall and Conway (2009), the trajectory is calculated with an assumed flight time first, later the parameter d is iterated until it matches the intended flight time; this value describes the allowed relative difference between the two flight time values
- Max_Thrust: maximum allowed thrust acceleration (m/s^2)
- Nrev_Max: Maximum allowed number of full revolutions around barycenter (see Wall and Conway, 2009)
- Step_No: The number of integration steps used for integrating Δv and flight time
- Fitness_Weight: Factor k in Eq. (5-34), giving the weight for the dominance of the Δv in the evaluation
- N_pop: Size of the population
- Gen_max: Maximum number of generations after which the search stops, the search also stops if for four generations in a row no replacement occurred, i.e. no solution improvement occurs
- Crossover_Constant: Used for differential evolution, setting the probability of crossing over of the trial vector's value to the new solution
- Diff_Weight: the Differential Weight, determining the impact of the vector difference on the mutation vector (see Section 5.1.5)

The trajectory code has been tested and verified with examples given by Wall and Novak (2011) for a 3D variant of the method and a Mars transfer. Wall and Novak report a trajectory with a launch date at 58969 MJD, a flight time of 520 days and a Δv of 5,640 m/s for a mission with a maximum thrust acceleration of 0.0004 m/s^2. Setting the same maximum allowed acceleration, a launch window from 58900 to 59100 and an allowed flight time between 450 to 600 days, the code created for this thesis produces a flight time of 546 days and a Δv of 5,546 m/s. The slight differences are likely to be attributed to:

i. The heuristic nature of the search, i.e. it is unlikely to exactly reproduce the results,
ii. the fact that in this thesis' work no inclination was regarded, whereas Mars has an inclination of 1.9° which was reckoned with in the example given by Wall and Novak, and
iii. possible differences in the evaluation of the solutions and the weight given on Δv by the author, which is not clearly documented for the given example and thus cannot be wholly reproduced.

A difference of less than 1.6% wrt the original Δv result and of 5% wrt the flight time is considered to be accurate enough for verification of the code, as the result is intended to be a preliminary mission analysis and no high-fidelity trajectory calculation.

To calculate planetary position data, the code uses ephemeris data from NASA Horizons database[2]. The ephemeris data is generated in advance and saved as file with the name of the corresponding body, e.g. "Earth.txt". To be used by the coded they are to be stored in a folder called "//files/eph_files".

2. Class Diagram

Figure A-1 shows the class diagram for the code. Each search is composed of a calculation object, which contains all relevant information as e.g. the launch body and target body. The number of maximum allowed gravity assists, from which the maximum number of segments is derived. This calculation associates with an optimizer, e.g. Differential Evolution or Random Search. Furthermore, the object Calculation contains Sequence Candidates of a minimum number of population, which is given by the optimization algorithm (for Differential Evolution it is four, based in its evolution principles, see 5.1.6) up to a maximum number as desired by the user. Each candidate has attached a mission object, which during calculation administers and organizes all variables and other relevant information. For each leg, the mission contains one shaping object, which basically is the software realization of the shape-based trajectory model. In turn each sequence candidate consists of k trajectories, which represent each respective mission leg.

[2] JPL Horizons online solar system data ephemeris computation system, http://ssd.jpl.nasa.gov/?horizons

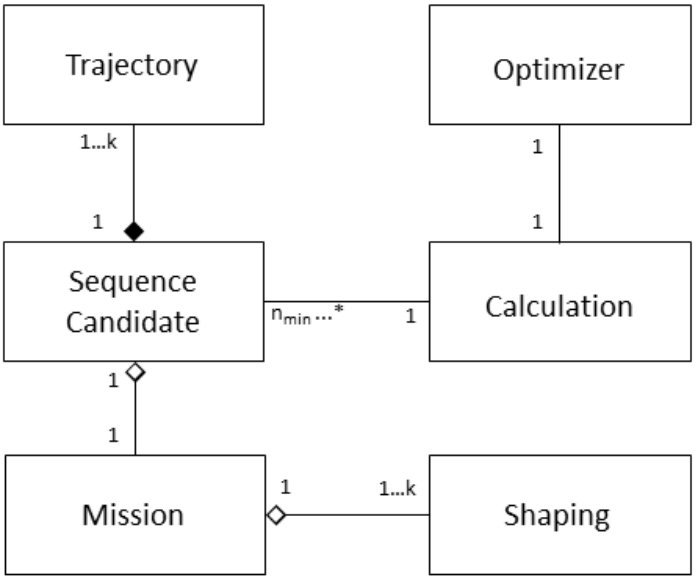

Figure A-1: The class diagram of the code used in this thesis with *k* as the number of legs for each missions candidate and n_{min} as the minimum population number required for optimization. For Differential Evolution this is four.