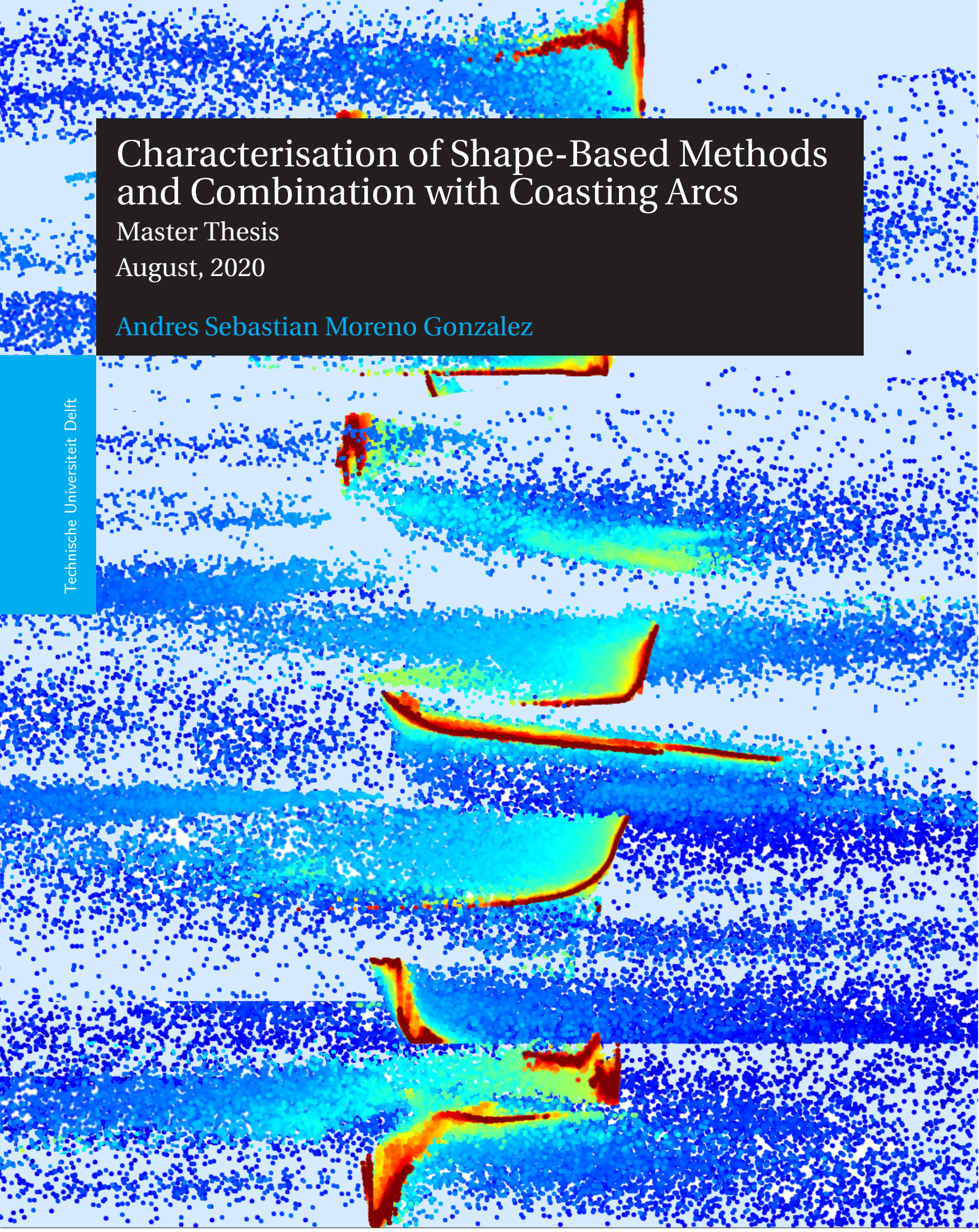
# Characterisation of Shape-Based Methods and Combination with Coasting Arcs

Master Thesis

August, 2020

Andres Sebastian Moreno Gonzalez

Technische Universiteit Delft



TU Delft Delft University of Technology

Challenge the future

# Characterisation of Shape-Based Methods and Combination with Coasting Arcs

Master Thesis
August, 2020

by

**Andres Sebastian Moreno Gonzalez**

in partial fulfillment of the requirements for the degree of

**Master of Science**

in Aerospace Engineering

at the Delft University of Technology,

to be defended publicly on Wednesday September 30, 2020 at 14:00.

| | | |
|---|---|---|
| Student number: | 4377761 | |
| Project duration: | September 16, 2019 – September 30, 2020 | |
| Supervisor: | Ir. R. Noomen | |
| Thesis committee: | Prof. dr. ir. P.N.A.M. Visser | TU Delft, Chair |
| | Ir. R. Noomen | TU Delft, Supervisor |
| | Ir. B.T.C. Zandbergen | TU Delft |

An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft Delft University of Technology

# PREFACE

This thesis is the final element of my Master Programme within the Space Flight master track and the Space Exploration profile at the Faculty of Aerospace Engineering at Delft University of Technology. After six years of living and studying in Delft, all things must come to an end. And I wouldn't have made it without the help of some people I want to thank.

I want to thank my supervisor Ron Noomen, he has listened to me week after week and asked me questions and provided insights I hadn't thought of, which I used for the thesis.

I would like to thank Marie Chambe who made the framework for shape-based methods in TUDAT, and thanks to which, I could spend more time on applying the methods due to a reduction in required implementation effort.

Thank you to all my room mates over the years, who I could talk to for many evenings. Thank you to my friends in Delft who I spent movie nights and game nights with and will continue to do so. Thank you to my old high school friends who I could always find online.

And finally, a big thanks to my parents and sister who supported me every week when I came home to Belgium. Thank you dad for driving me to Delft in the first year and accompanying me to the train station for the rest of the years. Thank you mom for the good food I missed during the week and the books you bought for me. And thank you all three of you for your love and support over the years.

*Andres Sebastian Moreno Gonzalez*
*Delft, August 2020*

# SUMMARY

Low-thrust propulsion's main advantage is its efficient propellant usage. This holds for manoeuvres as well as for major orbital changes, both for LEO/MEO/GEO satellites and for interplanetary missions. In designing such missions, extensive numerical models as well as less demanding analytic first-order approximations can be used. A specific kind of the latter category of methods are called shape-based methods. TU Delft's Astrodynamics Toolbox (TUDAT) contains the framework for shape-based methods and spherical shaping and hodographic shaping have already been implemented for use by students and staff. Having more shape-based method available allows for a more versatile toolbox and combining several shapes with for example coasting would lead to more attractive and efficient transfers.

The research question and its sub-questions that are to be answered in this master thesis are as follows:

***What is the benefit of using analytical low-thrust modelling in the design of interplanetary transfers?***

1. *What current methods for analytically modelling low-thrust transfer orbits in 2D and 3D can be implemented and verified in TUDAT?*
2. *How do the methods compare when applied to the same design mission to an interplanetary transfer?*
3. *Would the inclusion of a coasting arc in the design of the total transfer improve the results with respect to required propellant mass, thrust and time of flight?*

This report contains a summary of the literature study that was performed prior to this thesis. This summary provides the basics of orbital mechanics, the reference frame, coordinate systems and an argumentation that for interplanetary transfers all perturbations, except the thrust of the spacecraft can be neglected in a two-body problem. A summary of the review of shape-based methods is provided where exponential sinusoids and inverse polynomials were selected to be implemented in TUDAT for their simplicity yet versatility. Together with spherical shaping and hodographic shaping, these four methods were selected to be compared in performance using TUDAT. Reviewing past flown spacecraft, a thrust acceleration has been achieved between 0.037 and 0.37 mm/s$^2$. When analysing multiple interplanetary transfers and observing Tempel-1's eccentric orbit and relative distance from the Sun, the Earth-Tempel-1 interplanetary transfer was chosen to be used for both a non-coasting and a coasting transfer. The Runge-Kutta 4 integrator was selected for its simplicity and relatively good accuracy. Gaussian quadrature was selected for its better performance per computation. Cowell propagation was selected, as the coasting phase does not have too large changes in trajectory. Differential evolution (DE), particle swarm optimisation (PSO), simple genetic algorithm (SGA) and multi-objective evolutionary algorithm (MSOED) were selected to be used as optimisers for their performance.

This report describes the theory required for the implementation of both exponential sinusoids (ES) and the sixth-order inverse polynomials (IP) with 3D capabilities. These methods have been implemented into the TUDAT shape-based framework with the used inputs and outputs defined. For the IP, an extra derivation is shown to find the boundaries of its free coefficient. The successful verification and validation of both methods by reproducing the data provided in their source articles is presented in this report. This report describes the already implemented spherical shaping (SS) and hodographic shaping (HS) methods. For SS, an additional derivation is provided which automatically computes the bounds for the free coefficient. These bounds have been implemented and the verification has been presented.

When applying all four methods to the same grid search with start date, time of flight (TOF) and number of revolutions as common input, a similar pattern in $\Delta V$ was found that repeated for start date and TOF inputs. It was found to be linked to the relative position of Earth and Tempel-1; whereby certain relative positions yielded more optimal trajectories in terms of $\Delta V$. However due to the eccentricity of Tempel-1's orbit the pattern cannot fully repeat for low time of flights. It was found overall that the TOF's of the solutions with the lowest TOFs correspond to the Hohmann transfer time from Earth to Tempel-1, multiplied if more revolutions are used.

Both ES and IP had only a limited number of low $\Delta V$ regions below 15 km/s and usually below a TOF of 2500 days. For SS there is a single $\Delta V$ region, but it stretches over the full start date range and up to TOFs of 4000 days. The HS results are poor, as the lowest $\Delta V$ found is 26.8 km/s, which is due to the use of the default base functions, while proper base functions could achieve $\Delta V$s of 13 km/s as found in literature. While IP produced the single best result of 10.68 km/s, it did so for a limited number of inputs and was combined with high peak thrust accelerations. SS on the other hand produced consistently low $\Delta V$s for a very wide range of input values and still relatively low peak thrusts.

The same transfer was optimised, but now a global optimiser was used, which aims to find the best trajectory in terms of $\Delta V$ per shape-based method and per number of revolutions. Using the data from the grid search the input ranges were altered and the possible number of revolutions was increased. Multiple optimisers and seeds were used to determine their influence. The DE optimiser was found to produce the best results in terms of $\Delta V$ and without influence of seed number, compared to PSO or SGA.

Comparing the four methods it is found that the following order can be established with respect to their lowest $\Delta V$, based on both an unoptimised grid search and global optimisation: Inverse polynomial (10.51 km/s), Spherical shaping (11.71 km/s), Exponential sinusoid (13.09 km/s), Hodographic shaping (26.22 km/s).

The Earth-Tempel-1 transfer with coasting consists of a powered phase, a coasting phase, and a powered phase again, whereby the powered phases have a duration of one full revolution at most. Using this set-up, it was found that besides the start date and TOF, eight input variables are added: the state and the TOF of the phases.

For optimisation five analysis groups were set-up: SS-C-SS, IP-C-IP, IP-C-IP-MOO, IP-C-SS-MOO, and IP-C-IP-MOO, with C for coasting and MOO for multi-objective optimisation. The first two groups use single-objective optimisation (SOO) for $\Delta V$ only, whereas the last three are multi-objective optimisation (MOO) accounting for $\Delta V$ and peak thrust, and the last one is distinct from the third one (note that they have the same name) as it uses normalised fitness values for the optimisation. The optimiser settings were experimented with and the best ones selected. The best optimiser was found to be DE and MSOED for SOO and MOO respectively. MOO was found to be faster in convergence and required less manually changing the search area.

For all optimisations (SOO and MOO) with a focus on $\Delta V$ five to six distinct (in start date) trajectories were found. For SS-C-SS, IP-C-IP, IP-C-IP between 11.9-12.5, 10.3-10.7 and 10.6-11.0 km/s were found respectively. The peak thrusts were 0.97-1.31, 1.6-2.175 and 1.13-2.3 mm/s$^2$ respectively. For MOO, with a better balance between $\Delta V$ and peak thrust, no single optimal trajectory was found, rather a collection of trajectories with $\Delta V$ ranging between 10.62-12.2 km/s and peak thrust between 1.99-0.21 mm/s$^2$ respectively. The trajectories per group were usually very similar in terms of relative position of Earth and Tempel-1 and TOF.

For the thrust profiles for both coasting and non-coasting transfers it was found that they usually consisted of one or two peaks, and these usually represent a specific manoeuvre in the orbit. For example, increasing the transfer orbit radius, meeting the velocity of Tempel-1, and changing the inclination, the latter was either combined with another manoeuvre or was executed more gradually over time. The orbit raising was often done close at high speeds or close to the central body as it is more efficient; and inclination change was done usually at lower speeds or farther away from the central body. It is observed that for an increasing number of revolutions the best found $\Delta V$ increases, while the peak thrust decreases. Shorter travel times require higher peak thrust but less overall integrated thrust levels and vice versa.

The answer to the main research question and conclusion of this thesis is as follows:

ES and IP have been successfully implemented and verified in TUDAT. ES, IP, SS, and HS have been applied to an Earth-Tempel-1 transfer for a wide range of start dates and time of flights. The $\Delta V$s found are comparable to those found in literature for all methods besides HS, as it was applied in a standard setting. IP and SS produced the lowest required $\Delta V$s, with SS having slightly higher required $\Delta V$s, but with a lower peak acceleration thrust. The inclusion of a coasting arc between two powered arcs for IP and SS compared to the standard transfer, results in trajectories with similar and lower propellant mass, similar and lower peak thrust and similar and longer time of flights. It is recommended to use both IP and SS without any coasting when doing a first-order design of an interplanetary transfer, however the inclusion of coasting and the use of multi-objective optimisation with normalised fitness values could lead to improved results for peak acceleration thrust, for a limited increase in computation time.

# NOMENCLATURE

**General - Greek Symbols**

| | | |
|---|---|---|
| $\alpha$ | Polar angle | rad |
| $\Upsilon$ | First point of Aries | - |
| $\Delta A$ | Change in variable A | various |
| $\delta$ | Elevation angle | rad |
| $\eta$ | Intermediate parameters for transformation from Keplerian to Cartesian elements | - |
| $\gamma$ | Flight path angle | rad |
| $\mathscr{E}$ | Specific energy | $m^2/s^2$ |
| $\mu$ | Standard gravitational parameter | $m^3/s^2$ |
| $\Omega$ | Right Ascension of the Ascending Node | rad |
| $\omega$ | Argument of pericenter | rad |
| $\Psi$ | Second velocity angle | rad |
| $\Psi$ | Transfer angle | rad |
| $\tau$ | Time of pericenter passage | s |
| $\theta$ | True anomaly | rad |
| $\theta$ | Polar angle | rad |
| $\theta_f$ | Final polar angle | rad |
| $\varphi$ | Elevational velocity | m/s |
| $\varphi$ | Elevation angle | rad |
| $\xi$ | Intermediate parameters for transformation from Keplerian to Cartesian elements | - |

**General - Roman Symbols**

| | | |
|---|---|---|
| **A** | Vector of variable 'A' | |
| $a$ | Acceleration | $m/s^2$ |
| $a$ | Normalised Acceleration | - |
| $a$ | Semi-major axis | m |
| $c$ | Speed of light in vacuum | m/s |
| $E$ | Eccentric anomaly | rad |
| $e$ | Eccentricity | - |
| $f$ | Acceleration | $m/s^2$ |
| $g_0$ | Gravitational constant | $m/s^2$ |
| $H$ | Specific angular momentum | $m^2/s$ |
| $h$ | Specific angular momentum | $m^2/s$ |
| $h$ | Planck constant | $m^2$ kg/s |

| | | |
|---|---|---|
| $i$ | Inclination | rad |
| $I$ | Impulse | s |
| $l$ | Intermediate parameters for transformation from Keplerian to Cartesian elements | - |
| $m$ | Mass | kg |
| $M$ | Mean anomaly | rad |
| $m$ | Intermediate parameters for transformation from Keplerian to Cartesian elements | - |
| $N$ | Nodal Vector | - |
| $N$ | Number of revolutions | - |
| $n$ | Intermediate parameters for transformation from Keplerian to Cartesian elements | - |
| $P$ | Power | W |
| $p$ | Semi-latus rectum | m |
| $r$ | Radial coordinate | m |
| $T$ | Thrust | N |
| $t$ | Time | s |
| $V$ | Velocity | m/s |
| $v$ | Velocity | m/s |
| $x$ | x-coordinate | m |
| $y$ | y-coordinate | m |
| $z$ | Axial distance | m |
| $z$ | z-coordinate | m |

**Exponential Sinusoids**

| | | |
|---|---|---|
| $\phi$ | Phase angle | rad |
| $c$ | Abbreviation for $\cos(k_2\theta+\phi)$ | - |
| $k_0$ | Scaling factor | - |
| $k_1$ | Dynamic range parameter | - |
| $k_2$ | Winding parameter | - |
| $n$ | Direction of thrust | - |
| $q$ | Constant | $\text{rad}^{-1}$ |
| $s$ | Abbreviation for $\sin(k_2\theta+\phi)$ | - |
| $S_{k_2}$ | Class of trajectories | - |

**Hodographic Shaping**

| | | |
|---|---|---|
| $c_i$ | Coefficient for summation of functions | - |
| $f$ | Total thrust acceleration | $\text{m/s}^2$ |
| $n$ | Number of base functions | - |
| $R$ | Radial velocity function | m/rad |
| $s$ | Distance from central body | m |
| $T$ | Time-evolution function | s/rad |
| $u$ | Variable representing t or $\theta$ | m,rad |

| | | |
|---|---|---|
| $V$ | Velocity function | m/s |
| $v_i$ | Base functions for velocity profile | - |
| $Z$ | Axial-velocity function | m/rad |
| **Inverse Polynomials** | | |
| $a-g$ | Constant coefficients | - |
| $a_z-d_z$ | Constant coefficients | - |
| $q$ | Variable, integer, larger or equal to 3 | - |
| $T_{a_{in}}$ | Total in-plane thrust acceleration | m/s$^2$ |
| $T_{a_z}$ | Total out-of-plane thrust acceleration | m/s$^2$ |
| $T_a$ | Total thrust acceleration | m/s$^2$ |
| **Numerical Integration/Quadrature/Propagator** | | |
| $\alpha$ | Thrust angle | rad |
| $\mathbf{\Phi}$ | Increment function for numerical integration | - |
| $\gamma$ | Flight path angle | rad |
| $\mathbf{F}$ | Central body force and perturbing force | N |
| $\mathbf{r}$ | Cowell Propagator state | - |
| $a$ | Start of interval | - |
| $b$ | End of interval | - |
| $b_i$ | Weights | - |
| $c_i, c_i^*$ | Coefficients for integrators | - |
| $F$ | Specific thrust or acceleration | m/s$^2$ |
| $f(y,t)$ | Function of y(t) and time | various |
| $h$ | Step size | s |
| $j$ | Index for data points | - |
| $k_i$ | Function evaluations | - |
| $m$ | Number of data points | - |
| $m$ | mass of spacecraft | kg |
| $N$ | Number of variables (integration) | - |
| $N$ | Order of Runge-Kutta | - |
| $n$ | Index for data points | - |
| $y(t)$ | Function of time | various |
| **Spherical Shaping** | | |
| $\mathbf{\Phi}_k$ | Vector with terms for elevation shape function | various |
| $\mathbf{a_k}$ | Vector with coefficients for radial shape function | - |
| $\mathbf{A_{a_2}}$ | Part associated with the free coefficient $a_2$ | - |
| $\mathbf{A}$ | Homogeneous part in matrix form | - |
| $\mathbf{b_k}$ | Vector with coefficients for elevation shape function | - |
| $\mathbf{B}$ | Non-homogeneous part | - |

| | | |
|---|---|---|
| $\mathbf{R_k}$ | Vector with terms for radial shape function | various |
| $\Phi(\theta)$ | Elevation shape function | rad |
| $a_k$ | k$^{th}$ coefficient for radial shape function | - |
| $b_k$ | k$^{th}$ coefficient for elevation shape function | - |
| $C$ | Auxiliary variable | - |
| $c$ | Abbreviation representing cos(·) | - |
| $D$ | Time equation scalar function | m/rad$^2$ |
| $F_{1/2/3/4}$ | Auxiliary variables | various |
| $p_k$ | k$^{th}$ coefficient for radial shape function | - |
| $R(\theta)$ | Radial shape function | m |
| $s$ | Abbreviation representing sin(·) | - |
| $T(\theta)$ | Time shape function | s |
| $u$ | Control acceleration | m/s$^2$ |
| $Z$ | Inverse of the radial shape function | m$^{-1}$ |

**Subscript**

| | |
|---|---|
| $\square_0$ | Initial value |
| $\square_c$ | Central body |
| $\square_e$ | Exhaust, effective |
| $\square_f$ | Final value |
| $\square_h$ | Out-of-plane component |
| $\square_i$ | i$^{th}$ value |
| $\square_n$ | Normal component |
| $\square_r$ | Radial component |
| $\square_s$ | Sun |
| $\square_t$ | Tangential component |
| $\square_\theta$ | Cylindrical $\theta$-component |
| $\square_\theta$ | Spherical $\theta$-component |
| $\square_\varphi$ | Spherical $\varphi$-component |
| $\square_f$ | final value |
| $\square_i$ | Initial value |
| $\square_{LT}$ | Low Thrust |
| $\square_{max}$ | Maximum value of variable |
| $\square_{req}$ | Required value |
| $\square_{RK4}$ | Runge-Kutta 4 |
| $\square_r$ | Cylindrical r-component |
| $\square_r$ | Spherical r-component |
| $\square_{sp}$ | Specific |
| $\square_x$ | Cartesian x-component |

$\square_y$ Cartesian y-component

$\square_z$ Cartesian z-component

$\square_z$ Cylindrical z-component

**Superscript**

$\hat{\square}$ Unit vector

$\hat{\square}$ Integral w.r.t. the polar angle (hodographic shaping)

$\square'$ Derivative w.r.t. the polar angle

$\square''$ Second derivative w.r.t. the polar angle

$\square'''$ Third derivative w.r.t. the polar angle

$\tilde{\square}$ Approximate value (numerical integration)

$\tilde{\square}$ Integral w.r.t. time (hodographic shaping)

$\vec{\square}$ Vector

$\dot{\square}$ Derivative w.r.t. time

$\ddot{\square}$ Second derivative w.r.t. time

# ABBREVIATIONS

| | |
|---|---|
| AU | Astronomical Unit |
| DE | Differential Evolution |
| DO | Detailed Optimisation |
| EOM | Equations Of Motion |
| ES | Exponential Sinusoids |
| GO | General Optimisation |
| HS | Hodographic Shaping |
| IID | Input Identification |
| IP | Inverse Polynomial |
| IP-C-IP | Inverse Polynomial - Coasting - Inverse Polynomial |
| IP-C-IP-MOO | Inverse Polynomial - Coasting - Inverse Polynomial - Multi-Objective Optimisation |
| LT | Low Thrust |
| MJD | Modified Julian Date |
| MJD2000 | Modified Julian Date since 01/Jan/2000 at 12:00 |
| MOEA/D-DE | Multi-objective Evolutionary Algorithm by Decomposition DE |
| MOO | Multi-Objective Optimisation |
| MSOED | Multi-objective Evolutionary Algorithm by Decomposition DE |
| PaGMO | Parallel Global Multi-Objective Optimizer |
| PSO | Particle Swarm Optimization |
| RK(4) | Runge-Kutta (4) |
| SGA | Simple Genetic Algorithm |
| SOO | Single-Objective Optimisation |
| SS | Spherical Shaping |
| SS-C-IP | Spherical Shaping - Coasting - Inverse Polynomial |
| SS-C-IP-MOO | Spherical Shaping - Coasting - Inverse Polynomial - Multi-Objective Optimisation |
| SS-C-SS | Spherical Shaping - Coasting - Spherical Shaping |
| TOF | Time Of Flight |
| TUDAT | Technical University Delft Astrodynamics Toolbox |

# CONTENTS

# 1

# INTRODUCTION

Low-thrust propulsion is an attractive means to propel a space vehicle, because of its attractive efficiency (as expressed by the high $I_{sp}$) in terms of propellant mass usage. This holds for manoeuvres as well as for major orbital changes, both for LEO/MEO/GEO satellites and for interplanetary missions. To design such missions; fast, first-order and efficient representations of transfer orbits are typically used before a more detailed and more exact numerical model is applied. An elegant way to do so is to use so-called shape-based methods. While TU Delft's Astrodynamics Toolbox (TUDAT) contains a few shape-based methods like spherical shaping and hodographic shaping, more can be added, like exponential sinusoids and inverse polynomials. A not too often explored aspect is the addition of a coasting or non-powered phase, or the combination of multiple shape-based methods into one transfer. The ability to use a variety of different shape-based methods within TUDAT, would allow a more widespread use in trajectory analysis, and the fast analytic tool could eventually lead to both innovative transfers with various phases and eventually propellant savings for interplanetary transfers.

The research question and its sub-questions that are to be answered in this master thesis are as follows:

***What is the benefit of using analytical low-thrust modelling in the design of interplanetary transfers?***

1. *What current methods for analytically modelling low-thrust transfer orbits in 2D and 3D can be implemented and verified in TUDAT?*
2. *How do the methods compare when applied to the same design mission to an interplanetary transfer?*
3. *Would the inclusion of a coasting arc in the design of the total transfer improve the results with respect to required propellant mass, thrust and time of flight?*

The thesis starts with the implementation and verification of two shape-based methods in TUDAT, written in C++. Afterwards these two shape-based methods and two existing ones in TUDAT will be applied to the interplanetary Earth-Tempel-1 transfer case, where performance will be compared between the four. The two most promising ones will be used in the same transfer case, but now with the addition of a coasting phase, whereby the results will be compared with each other and the non-coasting transfer.

In order to answer the main question of this report, Chapter 2 provides the fundamentals, which contains information on the basics of both orbital mechanics including reference frames, the basics of low thrust, shape-based methods and current low-thrust technology, information on the Earth-Tempel-1 test case, the used numerical integration and propagation and finally the optimisers. Chapter 3 explains the theory of exponential sinusoids, its implementation, verification and validation. Chapter 4 uses the same structure, but now for the second shape-based method, the inverse polynomials. Chapter 5 provides information on the spherical shaping method and additions made in TUDAT for extended functionality. it also provides information on the hodographic shaping method and the used base functions. Chapter 6 contains the grid-search and optimisation results for the non-coasting Earth-Tempel-1 transfer case, and the four shape-based methods are compared to each other. Chapter 7 has the results for the Earth-Tempel-1 transfer case with the inclusion of a coasting arc, where inverse polynomials and spherical shaping are used separately and in combination. With all experiments performed, Chapter 8 provides the answer to the main research question and the final conclusions. Chapter 9 concludes the master thesis with the recommendations.

# 2

# FUNDAMENTALS

This chapter provides the relevant theory and decisions to deal with this topic, a full evaluation was done prior to this thesis project [1]. This chapter starts with the basics of orbital mechanics like reference frames, coordinate systems, the two-body problem and its perturbations (Section 2.1). Then low thrust is addressed, consisting of the basic equations, review of existing shape-based methods and current low-thrust technology (Section 2.2). After that, the Earth-Tempel-1 test case that is chosen for the shape-based methods to be applied to is explained, including its selection and relevant technical information (Section 2.3). This is followed by the presentation of the chosen integrators, quadrature and propagators. (Section 2.4). The chapter concludes with the optimisers to be used in the test case (Section 2.5).

## 2.1. ORBITAL MECHANICS

In this section theory about reference frames, coordinate systems, the two-body problem and perturbations are explained as they are required for modelling, representing and analysing the computed trajectories. The main source for this section is Wakker [2] unless stated otherwise.

### 2.1.1. REFERENCE FRAME

The state of a vehicle's position and velocity is presented within a reference frame. Using these, the motion of the spacecraft can be described, analysed and optimised. The research question contains the problem of an interplanetary transfer, thus the need for a heliocentric reference frame is clear.

An inertial or Newtonian reference frame is defined as:

"An inertial reference frame is a reference frame with respect to which a particle remains at rest or in uniform rectilinear motion if no resultant force acts upon that particle." [2, p.3]

This means that an inertial reference frame is not under the influence of a force and is not accelerating. Although the Solar System is under the influence of the Milky Way galaxy, as the relative size of the Solar System compared to the distances within the galaxy are small, it can be assumed the Solar System is pseudo-inertial.

The (non-rotating ecliptic) heliocentric reference frame has the Sun in its centre, the XY-plane is defined as the ecliptic plane with the +X-axis also going through $\Upsilon$. The +Z-axis has an angle between itself and the +Z-axis in the geocentric frame of 23°27', which is the obliquity of the ecliptic. As with the geocentric frame the +Y-axis completes the set such that an orthogonal set is formed. This frame can be seen in Figure 2.1.

The reference frame that will be used, is the one as defined in TUDAT with the Solar System barycentre (SSB) as central body and 'ECLIPJ2000' as the frame orientation. For timing purposes, TUDAT works with the Modified Julian Date 2000 (MJD2000) starting at astronomical epoch J2000.

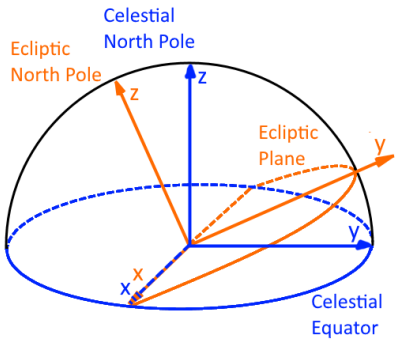



Figure 2.1: Heliocentric and Geocentric Reference Frame, figure adjusted from [2, Fig 11.3].

### 2.1.2. COORDINATE SYSTEMS

The state of a vehicle in space can be described with various coordinate systems. The coordinates systems used in computation and plotting are the Cartesian, cylindrical, spherical and Keplerian coordinate system. As TUDAT has the functionality to transform between the various coordinate systems, the common conversions are only given in Appendix A.

#### CARTESIAN COORDINATES

These are rectangular coordinates and are represented using: $x, y, z, \dot{x}, \dot{y}, \dot{z}$. as can be seen in Figure 2.2a. These coordinates are used often to propagate the orbit numerically as it is very straightforward to apply the laws of Newton. However these are not always the best for numerical accuracy or for understanding the motion of the spacecraft.

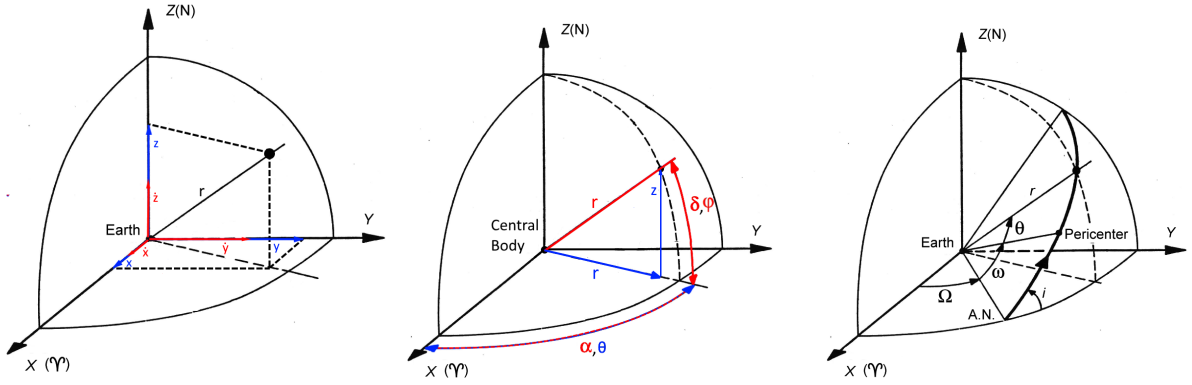
#### CYLINDRICAL AND SPHERICAL COORDINATES

Cylindrical coordinates are represented using $r, \theta, z, \dot{r}, \dot{\theta}, \dot{z}$. Thus the position is defined by a distance $r$ from the origin and the angle $\theta$ from the X-axis in the XY-plane. The third dimension is simply a z-coordinate as seen with Cartesian coordinates. The velocities are represented in the radial direction, the angular direction and the vertical direction. This representation is used mostly for two-dimensional representation of low-thrust trajectories.

Spherical coordinates do not use a 'z' to describe its third dimension but use elevation $\delta$ (sometimes denoted as $\varphi$). While the second element is defined in the same way, this angle is represented using $\alpha$ (sometimes denoted as $\theta$). For spherical coordinates, the velocity is represented using the magnitude of the velocity vector V and its orientation is defined using angles $\gamma$ and $\Psi$. Thus the complete element set uses: $r, \alpha, \delta, V, \gamma, \Psi$. Both coordinate systems are shown in Figure 2.2b.

#### ORBITAL/KEPLERIAN ELEMENTS

This representation, also called Keplerian elements, is different from the aforementioned coordinate systems as it describes the full motion at once, meaning the elements do not vary over time assuming no perturbations, except for $\theta$ which describes the position of the satellite within this orbital elements representation at a certain epoch. $a$ is the semi-major axis, $e$ the eccentricity, these indicate the shape of the orbit. $\omega$ is the argument of periapsis and indicates the orientation of the orbit with respect the body within the orbital plane. $\tau$ is the time of pericenter passage and allows for the orientation of the orbit and the position of the spacecraft in the orbit to be connected. With the two-dimensional orbit defined, its orientation in 3D space needs to be defined. The crossing of the orbital plane and the reference plane (equatorial or ecliptic plane) creates a line. The position where this line intersects with the orbit is called the nodes. The ascending node and descending node are where the spacecraft goes South-to-North and North-to-South respectively. The angle between the +X-axis (♈) and the line connecting the nodes is $\Omega$ and is called the right ascension of the ascending node. The last angle is $i$, the inclination which goes from the reference plane to the orbital plane at the ascending node. All these parameters can also be seen in Figure 2.2c.




(a) Cartesian coordinates

(b) Cylindrical and spherical coordinates

(c) Keplerian coordinates

Figure 2.2: Various coordinate systems to represent a spacecraft position around a central body. Figures adjusted from [2, Fig. 11.7].

### 2.1.3. TWO-BODY PROBLEM AND PERTURBATIONS

#### TWO-BODY PROBLEM

The motion of a vehicle around a fully symmetric central body with mass $m_c$ is governed by Equation 2.1, and $r$ is as defined in the spherical coordinate system. Only the central body exerts a gravitational force. Both laws of conservation of specific momentum $h$ and specific energy $\mathscr{E}$ are presented in Equations 2.2 and 2.3.

$$\frac{\mathrm{d}^2\vec{r}}{\mathrm{d}t^2} = -\frac{\mu}{r^3}\vec{r}, \quad \mu = Gm_c \tag{2.1}$$

$$\vec{h} = \vec{r} \times \vec{V} = \text{constant} \tag{2.2}$$

$$\frac{1}{2}V^2 - \frac{\mu}{r} = \mathscr{E} = \text{constant} \tag{2.3}$$

The trajectory equation is the solution to the governing equations and is shown in Equation 2.4. This equation can be translated into three shapes: An ellipse (e<1), a parabola (e=1) and a hyperbola (e>1); $p$ is called the semi-latus rectum.

$$r = \frac{h^2/\mu}{1 + e\cos(\varphi - \omega)} = \frac{p}{1 + e\cos\theta} = \frac{a(1-e^2)}{1 + e\cos\theta} \tag{2.4}$$

#### PERTURBATIONS

The two-body problem as discussed before is a simplification; besides the central gravity of the main body, there are various other effects/perturbations working on the spacecraft.

In interplanetary transfers, there are a few perturbations that could be important. The first one is the third-body effect, the effect of gravity of other planets besides the central body. Then there is the solar radiation pressure. Finally the thrust force, which is technically also a perturbation, but one that is controlled.

The most important perturbation as seen in Figure 2.3 is the solar radiation pressure. For this graph it was assumed the planets are at the same side of the Sun, aligned along one axis, in reality they could be on the other side of the Sun. For interplanetary trajectories the assumption of a two-body problem with thrust as the single perturbation is valid for distances larger than that of Mercury's orbit, due to the magnitude of the thrust acceleration.

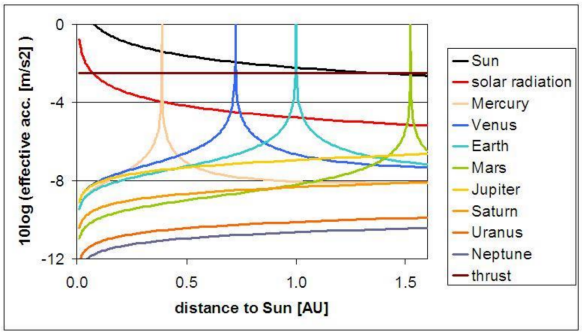



Figure 2.3: Magnitude of perturbations on heliocentric orbits [3, slide 60].

## 2.2. LOW THRUST

This section starts with the explanation of the low-thrust basic equations. It is then followed by a summary of the existing shape-based methods. Finally the existing low-thrust technology and thrust levels of past missions are presented.

### 2.2.1. LOW THRUST ORBITAL MECHANICS BASICS

The governing equation of motion for the two-body problem was already shown in Equation 2.1. To this equation a specific thrust force (acceleration) can be added, on top of this, the equation can be represented in polar coordinates as shown in Equation 2.5. The definition of the variables in this equation is shown in Figure 2.4. $\gamma$ is the flight path angle, $F$ is the specific thrust or acceleration, and $\alpha$ is the thrust angle. It should be noted that these equations of motion are valid for a particle with negligible mass compared to the central body with parameter $\mu$ and only thrust and gravity are the dominant forces. Thus these equations are valid for interplanetary and Earth-centred trajectories.

$$\begin{cases} \ddot{r} - r\dot{\theta}^2 + \frac{\mu}{r^2} = F\sin\alpha \\ \frac{1}{r}\frac{d}{dt}(r^2\dot{\theta}) = F\cos\alpha \\ \ddot{\theta}r + 2\dot{\theta}\dot{r} = F\cos\alpha \end{cases} \tag{2.5}$$

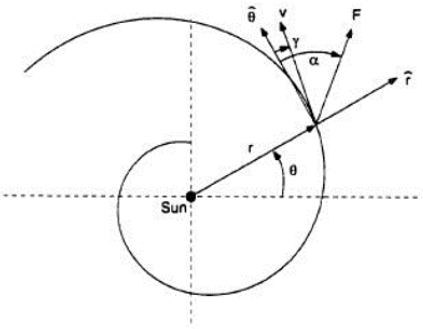



Figure 2.4: Sketch for modelling low-thrust in a two-body problem. Figure taken from [4, Fig. 1].

### 2.2.2. REVIEW OF EXISTING SHAPE-BASED METHODS

There exist many analytic methods to solve the low-thrust problem. A limited number of them can be implemented in TUDAT and used in the Earth-Tempel-1 test cases.

The methods that are considered are called shape-based methods and are analytic. This means that the problem of finding trajectories is solved not by trying to find a numerical solution to the equations of motion, but to assume the analytic solution (trajectory) exists. The assumed solution or 'shape' of the trajectory is then plugged into the equations of motion to determine what kind of acceleration is required to fly such a shape. Depending on the class of shapes (extra) parameters can be tuned to allow the trajectory to fit boundary conditions such as position and velocity. The advantage of shape-based methods is that they are analytic, meaning they can be solved quickly and thus can be used in large analyses with several thousands of variations in trajectories. The downside is the limited flexibility of the shapes and that they are unoptimised.

In the literature study an extensive review was performed, here the summary with Table 2.1 and a brief reasoning for the selected methods will be provided.

First of all **Exponential Sinusoids** are given. Even though they can satisfy few boundary condition and are only 2D; they are not complex and can be considered as the father of the shape-based approach. Thus exponential sinusoids can be a good reference and are selected here. The choice was made to use the exponential sinusoids expressed in the form of a Lambert type of problem, as it uses similar inputs as the other shape-based methods.

As exponential sinusoids cannot satisfy the velocity boundary condition, **Inverse Polynomials** are chosen to be implemented as it can solve for all main boundary conditions, is straightforward to implement and is used as a reference by other methods. The addition of 3D is included such that all methods can be applied to the Earth-Tempel-1 test case.

With 2D methods selected, a 3D method is chosen as well. **Spherical Shaping** is one of the few 3D methods, this method was found to produce better results than pseudo-equinoctial, and is better documented. Fourth, **Hodographic Shaping** has been chosen, it can also satisfy both position and velocity boundary conditions. It uses velocity profiles at its core, thus it is a different kind compared to exponential sinusoids. The main reasons these methods were chosen is for their good documentation. After the literature study it was discovered that another MSc student, Marie Chambe, had already implemented these two methods in TUDAT together with a shape-based method framework, meaning only the 2D methods needed to be implemented and all four methods can be applied to the Earth-Tempel-1 test case.

Table 2.1: Comparison of shape-based methods for low-thrust trajectory design. Table is based on [5, Table 1] and supplemented with other methods. The name of the method is given in standard text format, in italic the additions to the methods are indicated. In blue the chosen methods are shown.

| | BC | | | | | | |
|---|---|---|---|---|---|---|---|
| **Method** | **Position** | **Velocity** | **TOF** | **3D** | **Thrust Limited** | **Number of Revolutions** | **Ref.** |
| Exponential Sinusoids | Yes | No | No | No | No | Multi | [6] |
| *Lambert Targeting* | | | Yes | | | | [7] |
| *Spirals* | | | | | No[a] | | [8] |
| *CR3BP* | | | | | Yes | | [9] |
| Inverse Polynomial | Yes | Yes | Yes | No | No | Few | [10] |
| *Addition of 3D* | | | | Yes[b] | | | [11] |
| *Thrust limitation* | | | | | Yes | Multi | [12] |
| Pseudo-Equinoctial | Yes | Yes | Yes | Yes | Yes | Multi | [13] |
| Spherical Shaping | Yes | Yes | Yes | Yes | No | Multi | [14] |
| *Improvement of 3D* | | | | Yes | | | [15] |
| Fourier Series | Yes | Yes | Yes | No | Yes | Multi | [16] |
| *On-Off Thrust* | | | | No[c] | | | [17] |
| *Optimal Control* | | | No | | | | [18] |
| Hodographic Shaping | Yes | Yes | Yes | Yes | No | Multi | [5] |
| Pseudo-Spectral | Yes | Yes | Yes | Yes | Yes | Few | [19] |
| Trajectory equation | Yes | Yes | Yes | No | Yes | Multi/few | [20] |
| *Addition of 3D* | | | | Yes | | | [21] |
| Finite Time Polynomial | Yes | Yes | Yes | No | Thrust | Multi | [22] |

[a] Different thrust profiles are investigated
[b] Only valid results for up to 15° plane change
[c] Thrust profile is simply on/off

### 2.2.3. CURRENT LOW-THRUST TECHNOLOGY

In Table 2.2, the main characteristics of various types of electric propulsion are shown. It is seen there exists a wide range of thrusters with high specific impulses, yet relatively low thrust values. In comparison, standard chemical engines have relatively low specific impulses in the hundreds of seconds and have several kilo to Mega Newtons of thrust. In addition these low-thrust engines convert electrical power into thrust rather than with a combustion process as with chemical engines. In Table 2.3 missions that have flown are presented so a sense of achievable accelerations is obtained. It is seen that the thrust acceleration levels are between 0.037 and 0.37 mm/s$^2$ .

One of the most important equations to relate required $\Delta V$ and propellant mass is the Tsiolkovsky rocket equation as shown in Equation 2.6. With $m_0$ as the total mass before the rocket burn, $m_f$ as the (final) mass after the burn. and the difference between the two being the propellant mass. Here the exhaust velocity plays a role, as the higher it is, the less propellant is required for the same required change in velocity, which is the main advantage of low-thrust propulsion. For comparison chemical rocket engines have exhaust velocities between 2 and 4 km/s.

$$\Delta V = v_e \ln \frac{m_0}{m_f} \tag{2.6}$$

Table 2.2: Characteristics of electric propulsion [23, Table 17-4,7].

| | | Isp vac [s] | Thrust T [N] | T/Power [mN/kW] |
|---|---|---|---|---|
| **Electrothermal** | **Resistojet** | 150-700 | 0.005-0.5 | 743-905 |
| | **Arcjet** | 450-1500 | 0.05-5 | 113-135 |
| **Electrostatic** | **Magnetoplasmadynamic** | 2000-6000 | $5{\cdot}10^{-6}$-0.5 | |
| | **Pulsed Plasma Thruster** | 1200 | $5{\cdot}10^{-6}$-0.05 | 16.1-20.8 |
| | **Pulsed Inductive** | 1500-2500 | $5{\cdot}10^{-6}$-0.1 | |
| **Electromagnetic** | **Hall Effect Thruster** | 2000 | 25-200 | 54.3-55 |
| | **Colloid** | 1500 | $5{\cdot}10^{-6}$-0.005 | |
| | **Ion Thruster** | 2500-400 | 2-200 | 25.6-41 |

Table 2.3: Thrust and mass specifications of flown low-thrust missions (Table from [2, Tab. 19.1], masses from [24–27]).

| | | | | Mass [kg] | | Acceleration [m/s$^2$] | |
|---|---|---|---|---|---|---|---|
| **Mission** | **# Thrusters** | **Total Thrust [mN]** | $v_e$ **[km/s]** | **Wet** | **Dry** | **Minimum** | **Maximum** |
| Hayabusa | 4 | 32 | 31 | 510 | 377 | 0.063 | 0.085 |
| SMART-1 | 1 | 68 | 16 | 367 | 287 | 0.185 | 0.237 |
| Dawn | 3 | 276 | 30 | 1218 | 747 | 0.227 | 0.369 |
| GOCE | 2 | 40 | 40 | 1077 | 872 | 0.037 | 0.046 |

## 2.3. EARTH-TEMPEL-1 TRANSFER

This section provides information on coasting arcs, why the Earth-Tempel-1 transfer case was selected and the relevant properties of the Tempel-1 orbit.

### 2.3.1. NON-COASTING AND COASTING

Coasting is un-powered flight, for trajectories this means that there is no thrust and that the spacecraft only moves due to the central gravity in the two-body problem. In high-thrust transfers, coasting is used constantly with Hohmann transfers; thrust is used at departure and arrival only. For low-thrust transfers, coasting might provide more useful positions of the departure and arrival bodies and result in more attractive $\Delta V$ values; thrust is used during the whole powered arc.

An interplanetary transfer including coasting should be considered as a three-phase transfer with a coasting phase between two arcs of powered flight, described with the shape-based methods. For example first an exponential sinusoid arc is used, then a coasting phase and then another exposin arc. Practically this adds eight inputs to the non-coasting problem: six for the state when starting to coast and two to define the moments in time at which coasting starts and stops. This means that there cannot be too many coasting phases as this would severely increase the problem difficulty. An alternative to investigate is the combination of multiple shape-based methods in one transfer.

### 2.3.2. TEST CASE SELECTION

During the literature study, the interplanetary cases the shape-based methods had been applied to in their original paper were inspected, which produces Table 2.5. A transfer case was selected that was not too far from the central body to allow multiple full revolutions within a reasonable time frame (Neptune is too far from the Sun) and also a target that was not already applied to by all methods (Mars has been used extensively).

In order to be able to compare coasting and non-coasting transfers, the target should remain the same, this means the test case should allow for both types of transfers. An eccentric target orbit would prove the most challenging and hence most interesting for a coasting transfer. Of the three remaining targets, Mercury and 1989ML have the least eccentric orbits, and 1989ML has an orbit too close to Earth to be an interesting target. The Earth-Tempel-1 transfer as highlighted in blue in the table has been selected because it accounts for the listed considerations.

Table 2.4: Test cases used per method, chosen target in blue, test cases used in reference literature marked with a check mark. (table based on [28]).

| | **Target** | | | | |
|---|---|---|---|---|---|
| **Method** | **Mars** | **Mercury** | **Tempel-1** | **1989ML** | **Neptune** |
| Exponential Sinusoids | ✓ | | | | |
| Inverse Polynomial | ✓ | | | | |
| Hodographic Shaping | ✓ | ✓ | ✓ | ✓ | |
| Spherical Shaping | ✓ | | ✓ | ✓ | ✓ |

### 2.3.3. TEMPEL-1 ORBIT

As the orbit of Tempel-1 is not defined in TUDAT, it needed to be implemented. While a spice kernel exists for its orbit [29], it was found it only worked from the year 2000 to 2020. Therefore a more rudimentary yet fast and reasonably accurate method was used. Using its orbital elements provided in Table 2.5, the two-body problem was simulated and it was found the differences are within the bounds as listed in Table 2.6. All differences are acceptable, however the error in true anomaly $\theta$ is relatively large and is shown in Figure 2.5 together with the instantaneous radius of the orbit. It can be seen that the error depends on the location within the orbit; it is largest near the pericenter and for each pericenter passage the error 'hill' increases. As overall the error is limited to the position within the orbit near the pericenter, it is deemed acceptable for use in preliminary mission design as done in this thesis. The larger peaks at the tips of the 'hills' is due to the true anomaly going from 0 to 360 degrees and when calculating the difference near this switch of value, a peak appears. The orbit of Tempel-1 is implicitly verified in Subsection 5.1.3 as part of a main verification of spherical shaping. The orbital and synodic period are provided in Table 2.7 for quick reference.

Table 2.5: Tempel-1 Keplerian elements, with respect to the mean ecliptic and equinox of J2000 at Epoch 2457539.5 (2016-May-31.0). Taken from JPL Small-Body Data Browser [30].

| **a [AU]** | **e [-]** | **i [deg]** | **L/M [deg]** | **$\omega$ [deg]** | **$\Omega$ [deg]** |
|---|---|---|---|---|---|
| 3.1456923552 | 0.50963079493 | 10.473864146 | 348.76829861 | 179.2035808 | 68.749598031 |

Table 2.6: Maximum difference in orbital elements between using spice kernel position from 2000-2020 and the two-body representation using the orbital elements and reference epoch in Table 2.5, the values were taken for a difference of 5994.5 days from the reference epoch.

| $\Delta$ **a [AU]** | $\Delta e$ [-] | $\Delta i$ **[deg]** | $\Delta\ \boldsymbol{\theta}$ **[deg]** | $\Delta\ \boldsymbol{\omega}$ **[deg]** | $\Delta\ \boldsymbol{\Omega}$ **[deg]** |
|---|---|---|---|---|---|
| 0.03 | 0.01 | 0.07 | 33 | 0.4 | 0.25 |

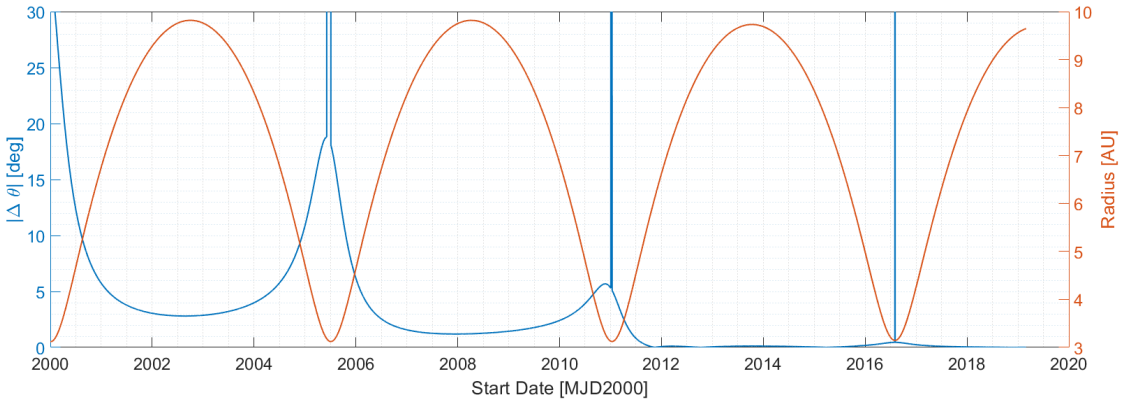



Figure 2.5: Absolute difference between true anomaly of orbit of Tempel-1 computed with the spice kernel [29] and the two-body equations (Table 2.5) (Left y-axis, blue). Radius of the spacecraft to the Sun over time (Right y-axis, blue).

Table 2.7: Orbit properties of Tempel-1 [28, Table 10.4].

| **Orbit Period [days]** | **Synodic period [days]** |
|---|---|
| 2016 | 446 |

## 2.4. INTEGRATION AND PROPAGATION

The shape-based methods have some parameters which need to be solved numerically. In the following subsections these will be explained and the chosen methods given. The coasting phase is a two-body problem, which could be solved using the analytic methods as shown in Subsection 2.1.3. However as TUDAT is used, its readily available numerical tools were chosen to be used in this thesis.

### 2.4.1. NUMERICAL INTEGRATION

The differential equation (Equation 2.1) of the two-body problem is numerically integrated for modelling the coasting phase. Using Equation 2.7 the numerical integration is performed [31]. The equation shows the state vector $\mathbf{y}$, the initial state $\mathbf{y}_0$, the step-size $h$, the increment function $\Phi$ and the approximate solution $\bar{y}(t_0, \mathbf{y}_0)$.

$$\mathbf{y}(t_0 + h) \approx \mathbf{y}_0 + h \cdot \mathbf{\Phi} = \bar{y}(t_0, \mathbf{y}_0) \tag{2.7}$$

This increment function is dependent on the selected numerical integrator. In TUDAT several are available. Runge-Kutta 4 is the default integrator with fixed step-size and is a multi-stage method, it evaluates the differential equation multiple times per increment. It was chosen as it is simple to use and moderately accurate and fast. Additionally the shape-based method framework uses it by default. The step-size for modelling the coasting phase is 1 day as it allows for a small enough resolution during optimisation, note that this is valid for coasting around the Sun.

### 2.4.2. NUMERICAL QUADRATURE

Several parameters, like $\Delta V$ and TOF, are computed using numerical computation of an integral and integrand [32]. These kinds of methods are called 'Numerical Quadrature'.

Numerical quadrature takes the sum of weighted function evaluations at points (=adjoints) along the interval of integration as shown in Equation 2.8. $y(t)$ represents the integrand, a known function, $a, b$ is the start and end of the interval, $k_i$ are the weights for the summation, $y(t_i)$ are the function evaluations at the adjoints and $m$ is the number of adjoints. For this equation and the next equations the variable $t$ for time is used, however this could be any variable.

$$\int_a^b y(t)\mathrm{d}t = \sum_{i=1}^{m} k_i y(t_i) \tag{2.8}$$

The accuracy of this method is dependent on the step-size, the spacing of the adjoints. The smaller the step-size, the more accurate the results, but more function evaluations are required. In TUDAT two functions for quadrature are implemented, the Trapezoid rule and Gaussian Quadrature.

The trapezoid rule uses equally spaced adjoints and uses weights of 1. At the boundaries the weights are set to 0.5. Gaussian Quadrature is more complex than the trapezoidal rule as it uses the Legendre polynomials for the weights and the adjoints are not fixed. It is a more accurate method for the same number of nodes, but requires more computational effort due to the weights. Gaussian quadrature is selected for its faster and more accurate results and lower truncation error. During the implementation the required number of nodes will be determined.

### 2.4.3. PROPAGATION

For solving differential equations, numerical integrators are used for evaluating the function while the propagators represent the governing equations of motion. There exist several sets of equations of motion in TUDAT like the Encke method and modified equinoctial equations, in order to reduce complexity and remove the need for many conversions of the state, the default method, the Cowell method is selected.

The Cowell method refers to direct numerical integration or equations in the form as shown in Equation 2.9 [33], with $\ddot{\mathbf{r}}$ the second derivative of the state, $\mathbf{F}$ the central body force and perturbing forces and $m$ the mass of the spacecraft. Conventionally the state is the Cartesian state with $x, y, z, \dot{x}, \dot{y}, \dot{z}$.

$$\ddot{\mathbf{r}} = \frac{\sum \mathbf{F}}{m} \tag{2.9}$$

## 2.5. OPTIMISATION

First the performance parameters are presented, it is followed by the working principle of the optimisers and finally the chosen optimisers are presented.

### 2.5.1. PERFORMANCE PARAMETERS

Some of the four methods allow for a certain degree of freedom in terms of possible trajectories. For example, the exponential sinusoids allow for many trajectories which go from one point to the other but with different times of flight. This can thus be minimised and it becomes an optimisation problem. The shape-based approach for low-thrust trajectories aims to find feasible solutions, and with some degrees of freedom, this initial guess can be made near-optimal. Near-optimal depends on the chosen performance parameters, for low-thrust problems these are as listed below.

1. Time Of Flight (minimise)
2. $\Delta V$ (minimise)
3. Max Acceleration (minimise)

### 2.5.2. GLOBAL OPTIMISER

For optimisation, the optimisers available in TUDAT will be used. Normally a combination of global and local optimisers would be used, however TUDAT does not yet have the local optimisers fully implemented, such that it works for all computers, as such global optimisers with more constricted input ranges have been used to optimise locally. Below a short description is given of the available global optimisers in TUDAT.

A meta-heuristic global optimiser starts with an initial population of individuals (input variables). Then for these individuals the fitness (performance parameters) is calculated. Based on those values the population is changed (mixed/copied/discarded/...) to create a new population. The population manipulation should be such that it leads to a population with better fitness values. Other manipulations to the populations are also applied such as immigration (importing completely new individuals) and mutation (altering individuals based on chance).

These optimisers are meta-heuristic, meaning no information on the gradients are required which make them more robust. For all the optimisers presented here there are extra steps, less steps, different schemes for each of the steps, etc.

### 2.5.3. SELECTED OPTIMISER ALGORITHMS

Below a brief overview of the selected ones is provided.

The main algorithm that has been used is Differential Evolution (DE). It is a type of genetic algorithm and changes the population of vectors using the differences between the vectors [34]. It is a widely used, well performing and robust optimiser. The implementation in PaGMO includes box constraints.

A second algorithm that has been used is Particle Swarm Optimization (PSO). It is based on how birds in swarms look for and go to locations with the most food [35]. This means that for a given bird (individual) its next movement and location is influenced by the fitness it encounters at the location it has travelled and also by the fitness as found by other birds.

The third and last single-objective optimiser that has been used is Simple Genetic Algorithm (SGA). It is one of the first evolutionary algorithms [36]. The main aspects of the algorithm are Selection, Crossover, Mutation, Reinsertion (Elitism, keeping the best individuals).

Finally one multi-objective optimiser has been used: Multi-Objective Evolutionary Algorithm (MOEA/D-DE). The optimiser makes use of problem decomposition, this means it will make the overarching problem into multiple smaller scalar problems which can be more easily optimised [37]. The optimisation of the so-called sub-problems is done with evolutionary operators like DE. This algorithm has been tried and tested and considering its success a good candidate.

# 3

# EXPONENTIAL SINUSOIDS

The exponential sinusoids are one of the two methods to be implemented in the existing TUDAT shape-based method framework. This chapter starts with the relevant theory used for implementation (Section 3.1). Then the implementation itself is presented (Section 3.2). After that, the verification of the implementation is done (Section 3.3). Finally the validation performed using the cases in the original papers is shown (Section 3.4).

## 3.1. THEORY

The following section is based on the papers by Petropoulos [4, 6, 38] unless indicated otherwise. This section describes the main and derivative equations for exponential sinusoids for the specific tangential thrust case.

### 3.1.1. BASIC EQUATIONS

The shape equations are given in Equation 3.1, with the variables as defined before. There, $r$ is the radial distance, $k_0$ the scaling factor, $k_1$ the dynamic range parameter, $k_2$ the winding parameter, $\theta$ the true anomaly and $\phi$ the phase angle.

$$\begin{cases} r = k_0 \exp\left[k_1 \sin k_2\theta + \phi\right] \\ \dot{r} = r\dot{\theta}\tan\gamma \\ \dot{\theta}^2 = \left(\frac{\mu}{r^3}\right)\frac{1}{\tan^2\gamma + k_1 k_2^2 s + 1} \end{cases} \quad \begin{cases} s \equiv \sin\left(k_2\theta + \phi\right) \\ c \equiv \cos\left(k_2\theta + \phi\right) \\ \tan\gamma \equiv k_1 k_2 c \end{cases} \tag{3.1}$$

In Equation 3.2, $a$, the acceleration normalised with respect to the local gravitational acceleration, is shown.

$$a = \frac{(-1)^n \tan\gamma}{2\cos\gamma}\left[\frac{1}{\tan^2\gamma + k_1 k_2^2 + 1} - \frac{k_2^2(1 - 2k_1 s)}{\left(\tan^2\gamma + k_1 k_2^2 + 1\right)^2}\right] \tag{3.2}$$

Values for the design parameters for which $k_1 k_2^2$ approaches the value '1' should be avoided as near the periapsis ($s = -1$) this can lead to values of infinity for the thrust. Values which lead to $k_1 k_2^2$ being larger than one can lead to a negative $\dot{\theta}$ which should also be avoided as this can lead to a thrust profile that cannot be performed using tangential thrust (when assuming prograde motion).

### 3.1.2. LAMBERT TARGETING

Petropoulos used the theory of exponential sinusoids in a shooting type of problem. Izzo reformulated this for a Lambert type of problem, and this will be the basis for this subsection and implementation in TUDAT [7]. An initial position $r_1$, final position $r_2$, and traversed angle $\Psi$ are required as input, with this a group of solutions can be determined that satisfies these conditions. The variable $k_2$ is fixed and $k_0, k_1$ and $\phi$ need to be found, $\theta_{initial}$ is also set to zero. This leads to the boundary equations as seen in Equation 3.3.

$$\begin{cases} r_1 = k_0 \exp\left[k_1 \sin\left(\phi\right)\right] \\ r_2 = k_0 \exp\left[k_1 \sin\left(k_2\theta_f + \phi\right)\right] \end{cases} \tag{3.3}$$

$$\theta_f = \Psi + 2\pi N \tag{3.4}$$

In Equation 3.4 the final azimuth, $\theta_f$ is calculated, also assuming the initial angle is zero. A value for $k_2$ is assumed. $\theta_f$ is used to limit the range of $\gamma_1$ as seen in Equations 3.5 and 3.6. The other parameters for the exponential sinusoid are calculated in Equations 3.7, 3.8 and 3.9.

$$\tan\gamma_{1,min,max} = \frac{k_2}{2}\left[-\ln\left(\frac{r_1}{r_2}\right)\cot\left(\frac{k_2\theta_f}{2}\right) \mp \sqrt{\Delta}\right], \tag{3.5}$$

$$\Delta = \frac{2(1-\cos\left(k_2\theta_f\right))}{k_4^4} - \ln^2\left(\frac{r_1}{r_2}\right) \tag{3.6}$$

$$k_1^2 = \left[\frac{\ln\left(\frac{r_1}{r_2}\right) + \frac{\tan\left(\gamma_1\right)}{k_2}\sin\left(k_2\theta_f\right)}{1-\cos\left(k_2\theta_f\right)}\right]^2 + \frac{\tan^2\left(\gamma_1\right)}{k_2^2} \tag{3.7}$$

$$sign(k_1) = sign\left[\ln\left(\frac{r_1}{r_2}\right) + \frac{\tan\left(\gamma_1\right)}{k_2}\sin\left(k_2\theta_f\right)\right] \tag{3.8}$$

$$\phi = \arccos\left(\frac{\tan\left(\gamma_1\right)}{k_1 k_2}\right), \quad k_0 = \frac{r_1}{\exp k_1 \sin\left(\phi\right)} \tag{3.9}$$

In order to evaluate the time of flight, Equation 3.10 is used by a summation, with values to be evaluated at the beginning of each sub-interval. $\theta_i$ ranges between 0 and $\theta_f$, and from this and the other exponential sinusoid parameters, the sub-parameters (with subscript $i$) are determined as seen in Equation 3.11.

$$TOF = \sum_{i=1}^{i_{max}} \Delta\theta_i \sqrt{r_i^3(\tan^2\gamma_i + k_1 k_2^2 s_i + 1)/\mu} \tag{3.10}$$

$$\gamma_i = \arctan\left(k_1 k_2 c_i\right), \quad r_i = k_0 \exp\left(k_1 s_i\right), \quad c_i = \cos\left(k_2\theta_i + \phi\right), \quad s_i = \sin\left(k_2\theta_i + \phi\right) \tag{3.11}$$

In Equation 3.12 the equation for the required $\Delta V$ is shown for a given exponential sinusoid.

$$\Delta V = \int_0^{\theta_f} \frac{|a|}{\dot{\theta}} \mathrm{d}\theta \tag{3.12}$$

In Equation 3.13 the way to compute the radial, angular and total velocity is shown.

$$\vec{V}_r = \dot{r}\cdot\hat{r}, \quad \vec{V}_\theta = r\dot{\theta}[(\hat{r}_i \times \hat{r}_f) \times \hat{r}], \quad \vec{V} = \vec{V}_r + \vec{V}_\theta \tag{3.13}$$

Exponential sinusoids are normally 2D trajectories, thus a plane of transfer must be found from the departure to the arrival body. This plane can be found by using the cross product of the position vectors of the target and arrival body, which results in the normal to the transfer plane. The traversed azimuthal angle (in-plane) can be found using the great circle arc formula as shown in Equation 3.14, this computes the angle between two points on a sphere.

$$\Psi = \arctan \frac{|\hat{r}_1 \times \hat{r}_2|}{\hat{r}_1 \bullet \hat{r}_2} \tag{3.14}$$

As a summary for 'Izzo targeting' the next steps are done: (1) Given $r_1, r_2, \Psi, TOF_{req}$ (2) Assume $k_2$ (3) Assume $N$ (4) Compute $\gamma_{1,min/max}$ (5) Assume $\gamma_1$ (6) Compute $k_0, k_1, \phi, TOF$ (7) Iterate until $TOF \approx TOF_{req}$. With the shape the required forces, $\Delta V$ and velocities can be computed.

## 3.2. IMPLEMENTATION

Implementing exponential sinusoids into TUDAT was made easier by the existing hodographic and spherical shaping methods implemented by Chambe. Due to the presence of a framework within TUDAT, many functions were already in place and it was mostly adding or removing some inputs, replacing the contents of the existing functions with the unique equations and variables presented in the section before, and finally rewriting the contents of the output equations. Therefore only the used inputs and outputs will be discussed here.

The inputs are as shown in Table 3.1; the first column shows the name of the variables in TUDAT, the second column gives a short description of the variables, and the last column shows the type of variable. The content is mostly taken and transformed into tabular form, from the TUDAT documentation [39]. TUDAT as variable type means it is a combination of types, that form for example the settings for integration. Compared to the existing framework, one input has been added, the 'windingParameter'. It is an input specific for the exponential sinusoids and a possible parameter for optimisation.

Table 3.1: Inputs for both the exponential sinusoid and inverse polynomial shape-based method class in TUDAT. Inputs taken from [39]. windingParameter* is the input unique to exponential sinusoids.

| Input | Description | Type |
|---|---|---|
| initialState | State of the spacecraft at departure. | 6D vector |
| finalState | State of the spacecraft at arrival. | 6D vector |
| requiredTimeOfFlight | Time of flight required for the shape-based trajectory. | double |
| numberOfRevolutions | Required number of revolutions before the spacecraft reaches its final state. | double |
| bodyMap | Map of pointers to Body objects involved in the low-thrust trajectory. | TUDAT |
| bodyToPropagate | Name of the spacecraft to be propagated. | string |
| centralBody | Name of the central body of the low-thrust trajectory. | string |
| windingParameter* | An adjustable coefficient of the exponential sinusoid function | double |
| rootFinderSettings | Settings that define the root finder algorithm, used to find the proper gamma coefficient value for which the required time-of-flight would be achieved. | TUDAT |
| integratorSettings | Integrator settings (empty by default), used to propagate the spacecraft mass or thrust profiles, or to numerically propagate the fully perturbed trajectory (as a means to assess the quality of the analytical shaped-based preliminary design). | TUDAT |

As discussed in the fundamentals chapter, the Gaussian quadrature is used to calculate integrals numerically. An internal setting that was changed in TUDAT is the number of nodes used for Gaussian quadrature, originally set to 16. However after computing the test cases it was found to be insufficient during the validation phase and 64 nodes are used instead. In the validation section (Section 3.4), more information will be provided.

The TUDAT framework allows for many outputs, however more than required for the purpose of this thesis, as such these functions were not rewritten and unused. In Table 3.2 the outputs that were defined and used are shown. The given inputs are always able to be requested again, an output that was added to the framework is the 'infeasibleTOF'. Instead of letting the shape method result in an error, it finishes normally, however with an extra variable indicating the inputs failed to produce a shape adhering to the given inputs. The remainder of the outputs have been used to plot the trajectories and to compute the performance parameters required for the test cases. Note that for the thrust acceleration only the magnitude is computed and not the full thrust vector.

Table 3.2: Outputs for both the exponential sinusoid and inverse polynomial shape-based method class in TUDAT.

| **Output** | **Description** | **Type** |
|---|---|---|
| Inputs | The inputs used for the output computation can be requested | TUDAT |
| InfeasibleTOF | The shape-based method cannot compute a valid trajectory for the given inputs | bool |
| coefficients | The coefficients that define the shape function(s) | XD vector |
| stateVector | The current state vector for a given azimuth angle or time | 6D vector |
| deltaV | Delta V | double |
| thrustAccelerationMagnitude | The thrust acceleration magnitude for a given azimuth angle or time | double |

## 3.3. VERIFICATION

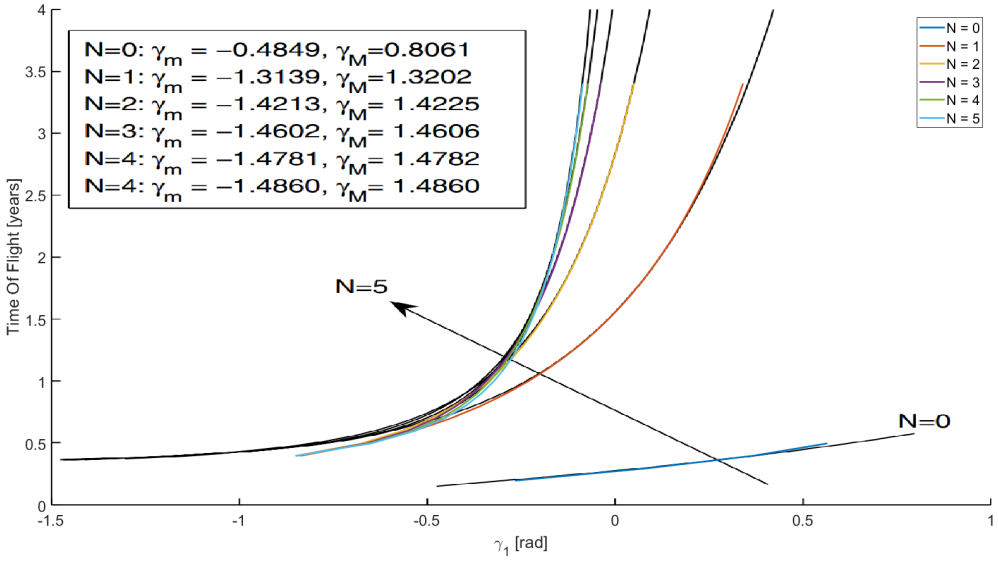
There are two main methods of verification that have been used: unit testing of the functions and system testing of the shape class.

Unit testing was performed by comparing the intermediate results of the functions with the results found in [40]. This comparison of radius, acceleration and angular acceleration for a given azimuth angle is shown in Table 3.3, with the inputs required for the Izzo method as given in the caption. The shape coefficients $k_0, k_1$ and $\varphi$ were computed as well and found to be almost equal to those in the reference material. It can be seen that the difference in results is very small (<0.005%), which indicates the functions and shape method has been implemented correctly. The remaining difference can be attributed to using a slightly different central body gravitational parameter and constant for the astronomical unit to SI conversion.

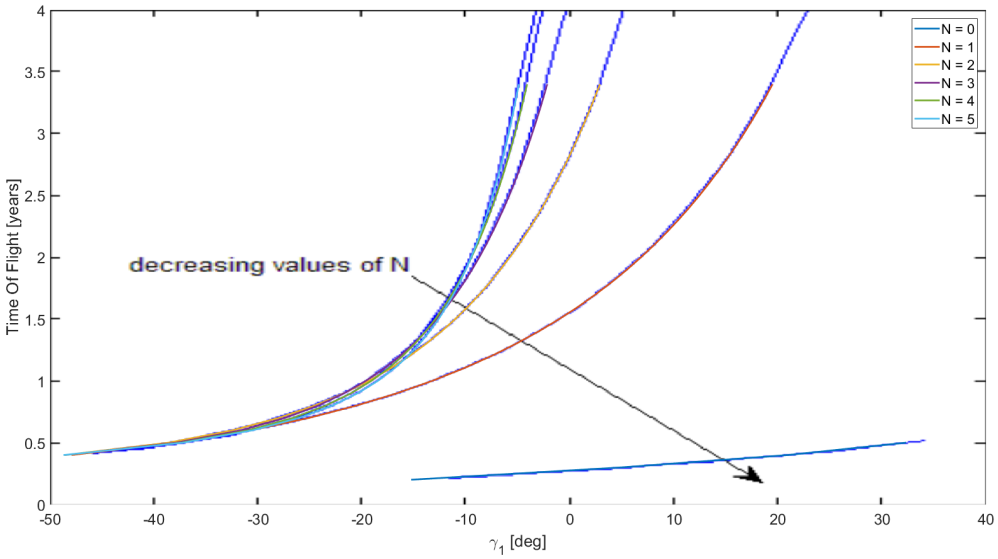
Table 3.3: Intermediate results for $r$, $a$ and $(\mathbf{d}t/\mathbf{d}\theta)^2$ with the given inputs: $r_1 = 1, r_2 = 1.5, k_2 = 1/12, \Psi = 90°, N_{rev} = 2, \gamma_= - 80°$ and computed shape coefficients: $k_0 = 4.9410^{55} m$, $k_1 = -123.04$ and $\varphi = 0.9847$ rad. Reference results taken from [40, Slide 67,69]

| | $r$ **[km]** | | | $a$ **[km/s$^2$]** | | | $(\mathbf{d}t/\mathbf{d}\theta)^2$ **[(s/rad)$^2$]** | | |
|---|---|---|---|---|---|---|---|---|---|
| $\boldsymbol{\theta}$ [rad] | **Computed** | **Reference** | **Diff [%]** | **Computed** | **Reference** | **Diff [%]** | **Computed** | **Reference** | **Diff [%]** |
| 0 | 1.50E+08 | 1.496E+08 | 0.00007 | -2.852E-06 | -2.853E-06 | -0.00417 | 8.187E+14 | 8.186E+14 | 0.00406 |
| 0.7068583 | 3.25E+06 | 3.252E+06 | 0.00009 | -5.980E-03 | -5.980E-03 | -0.00406 | 6.964E+09 | 6.963E+09 | 0.00411 |
| 1.4137167 | 1.02E+05 | 1.022E+05 | -0.00029 | -5.970E+00 | -5.971E+00 | -0.00407 | 1.743E+05 | 1.743E+05 | 0.00401 |
| 2.120575 | 4.70E+03 | 4.703E+03 | 0.00004 | -2.764E+03 | -2.765E+03 | -0.00412 | 1.323E+01 | 1.323E+01 | 0.00392 |

A system test that was performed is reproducing the results found in [7], where the time of flight is shown for a given $\gamma_1$. This graph is the culmination of the Izzo method for exponential sinusoids, and if the method is correctly implemented the same results should be achieved. The comparison is shown in Figure 3.1a, by overlapping the computed (coloured) and reference (black) graph. The values computed for the minimum and maximum of the range for $\gamma_1$ are the same as in the reference material. It is seen that for a higher $\gamma_1$ and TOF the two are very similar, however not so much for the lower values. After more unit tests no error could be found which could explain the visible difference. Paulino has reproduced the graph as well in her thesis [8], using the same manner of comparison Figure 3.1b can be made. It is seen that for all values of $\gamma_1$ the resulting TOF is similar. Either both Paulino's and the current implementation are incorrect or the Izzo's is. The latter is reinforced by the error in the legend of the original graph (N=4 shown twice) and that for the validation process as shown in Section 3.4, Izzo's given inputs are incorrect.




(a) Comparison between TUDAT ES implementation and results from Izzo [7, Fig. 2] shown in black.

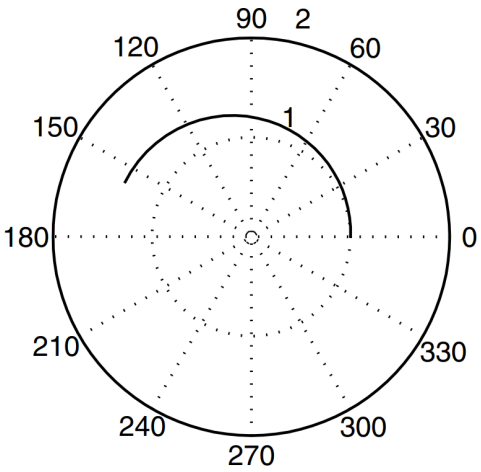



(b) Comparison between TUDAT ES implementation and results from Paulino [8, Fig. 5.6a] shown in blue.

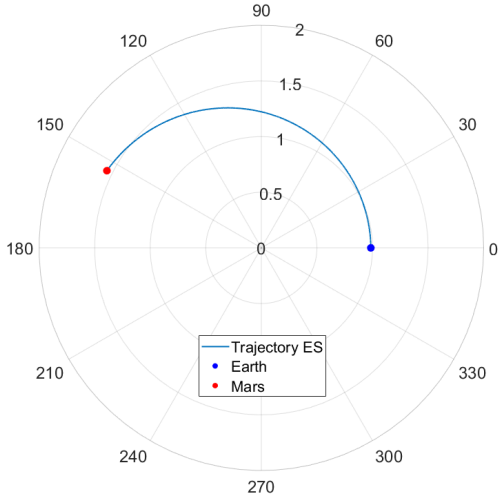
Figure 3.1: Time of flight versus the initial flight-path angle curves for the class $S_{1/12}$[1,1.5,$\pi$/2,N]. Comparison between TUDAT ES implementation and reference literature.

## 3.4. VALIDATION

Validation of the exponential sinusoids is done using the main results of Izzo's paper [7]. Izzo found the most efficient trajectories in terms of $\Delta V$ for an Earth-Mars transfer, excluding $\Delta V$ required for satisfying the velocity requirements at the departure and arrival body. He investigated start time, time of flight and winding parameter, for zero, one and two revolutions. No direct comparison of $\Delta V$ or any other parameters can be made as Izzo only provides the inputs for the most optimum cases. He does provide the trajectory when zero full revolutions are used. When comparing Izzo and the new computation, it was found they do not match. More specifically the initial and final position of the bodies do not match, indicating that the values provided are incorrect. After changing the start time of the transfer manually a new one can be found such that the trajectory can be reproduced, this reference and computed trajectory is shown in Figure 3.2a and 3.2b respectively. It is seen they match correctly when the start date is altered. Note that the start dates here are in Modified Julian Date (MJD) without the '2000', however the difference in trajectories could be due to improper notation of the start date. For both the MJD and the MJD2000 notation the trajectories were computed, neither trajectory was the one found by Izzo.




(a) Reference trajectory using start date MJD 4115. Figure taken from [7, Fig. 3].




(b) Computed trajectory using start date MJD 4401.

Figure 3.2: Earth–Mars optimum in the trajectory class of the exponential spirals $N = 0$, $TOF = 206$ days, and $k_2 = 0.928$. Units are all non-dimensional.

While the computed $\Delta V$ is not given by Izzo, it is claimed the solution is optimal, which can be checked. By varying the start time and time of flight around the given optimal values and then computing the $\Delta V$ it is expected Izzo's values are in an optimum. The results of this analysis are shown in Figure 3.3. It is seen that the used inputs are indeed around the minimum $\Delta V$, the figure also includes a zoom-in of the optimal area. As the start date was found manually such that the trajectories match visually, some error in the optimal start date was expected (error of +35 days) for the given TOF (yellow line). Strangely Izzo's optimal value for TOF seems to be too short, as the $\Delta V$ could be even lower for a longer TOF (+50 days) (lower blue line).

In the implementation section (Section 3.2), the change from 16 to 64 nodes for the Gaussian quadrature was mentioned, here this change will be justified. The results shown in Figure 3.3 were computed with 64 nodes, in Figure 3.4 the same results but using 16, 32 and 48 nodes for Gaussian quadrature are shown. It can be observed that using a lower number of nodes results in less smooth results with seemingly unnatural 'kinks'. It was found that when using too few nodes, the integral cannot be properly computed; the more nodes are used the more the solution to the integral converges to a single value. A decision was made to use 64 nodes, not only for the exponential sinusoids, but for the other shape-based methods as well.

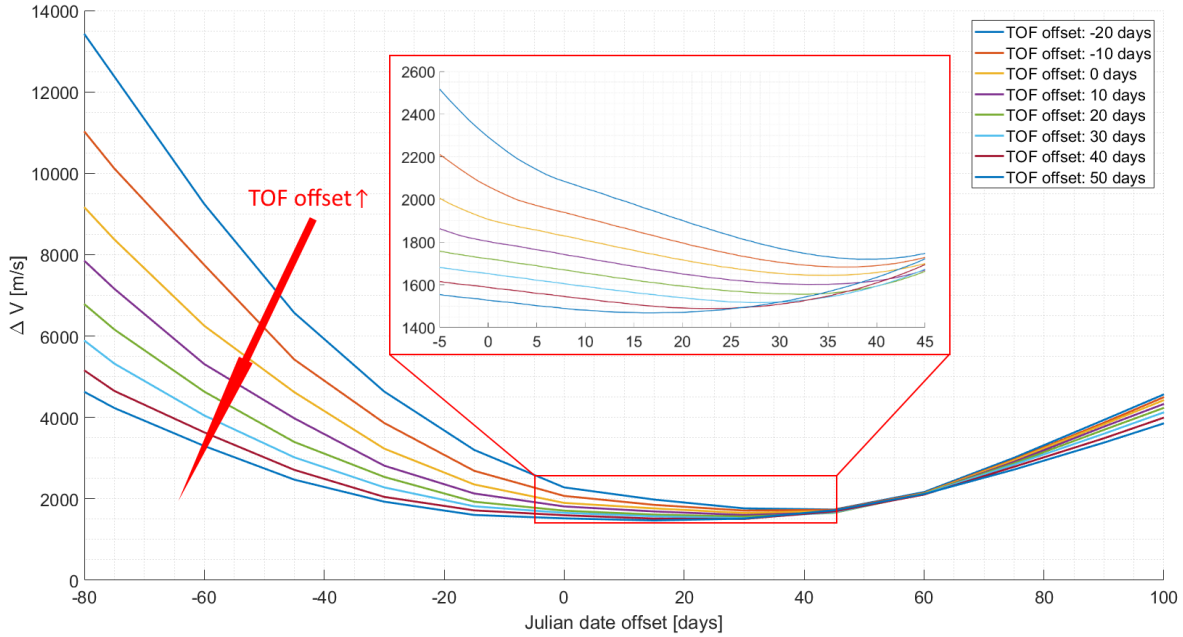



Figure 3.3: Required $\Delta V$ for Exponential Sinusoid trajectories with deviations in time of flight and start date, using a nominal Earth-Mars transfer starting on MJD 4401.5 and a time of flight of 206 days. Zoomed-in region with a resolution of 1 day in start date offset, at the lowest attained $\Delta V$.

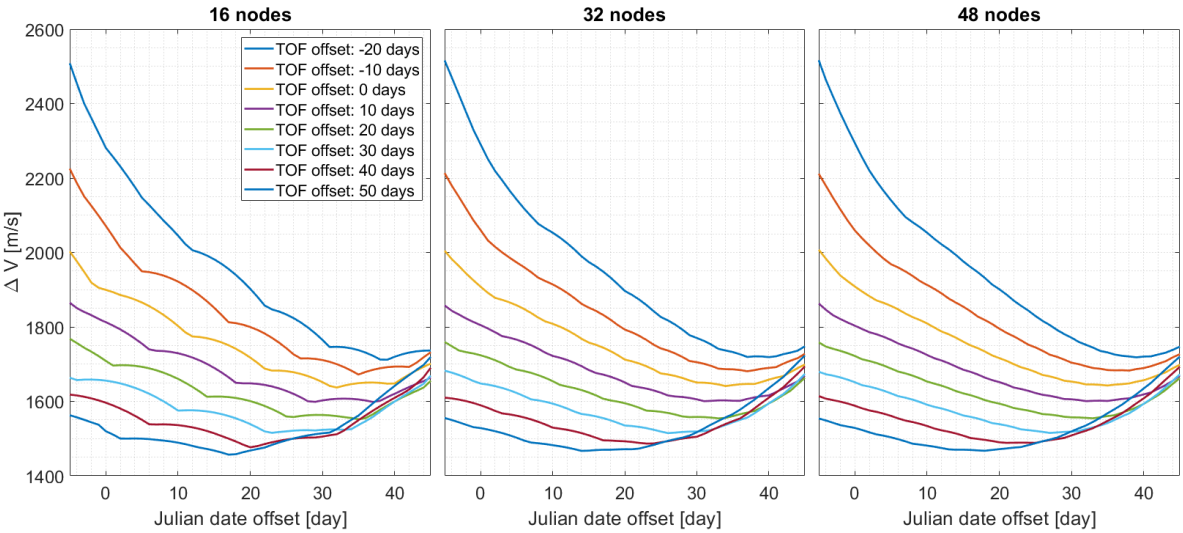



Figure 3.4: Required $\Delta V$ for Exponential Sinusoid trajectories with deviations in time of flight and start date, using a nominal Earth-Mars transfer starting on MJD 4401.5 and a time of flight of 206 days. Same trajectories are computed for different number of nodes for Gaussian quadrature.

# 4

# INVERSE POLYNOMIALS

The inverse polynomials are the second of the two methods to be implemented in the existing TUDAT shape-based method framework. This chapter starts with the relevant theory used for implementation (Section 4.1). Then the implementation itself is presented (Section 4.2). Next, the verification of the implementation is shown (Section 4.3). Finally the validation performed using the original papers is shown (Section 4.4).

## 4.1. THEORY

### SHAPE FUNCTION

Wall and Conway suggested the use of an inverse polynomial for the shape-based approach. This section will be based on their work in [10], unless indicated otherwise. In Equation 4.1 an inverse polynomial is shown; a fifth-order with the black terms and a sixth-order when adding the blue term. The blue term was included to be able to solve for a given transfer time. This higher order allows a Lambert's type of problem formulation. $a$ to $g$ are the coefficients of the polynomial and are used to satisfy the boundary conditions. From now on the blue term will be shown in black.

$$r = \frac{1}{a + b\theta + c\theta^2 + c\theta^2 + d\theta^3 + e\theta^4 + f\theta^5 + g\theta^6} \tag{4.1}$$

### AUXILIARY FUNCTIONS

For this assumption of the shape, the equations of motion are as stated in Equation 2.5, however with this method, the thrust acceleration $F$ is also defined as $T_a$.

Using Equation 4.2, the flight path angle can be determined analytically, and will be used as one of the boundary conditions. With the equations of motion and the assumption of tangential thrust the notation in Equation 4.3 can be made.

$$\tan\gamma = -r \cdot (b + 2c\theta_f + 3d\theta_f^2 + 4e\theta_f^3 + 5f\theta_f^4 + 6g\theta^5 \tag{4.2}$$

$$\dot{\theta}^2 = \frac{\mu}{r^4} \frac{1}{[(1/r) + 2c + 6d\theta + 12e\theta^2 + 20f\theta^3 + 30g\theta^4} \tag{4.3}$$

$$\ddot{\theta} = -\frac{\mu}{2r^4}\left[\frac{4\tan\gamma}{[(1/r) + 2c + 6d\theta + 12e\theta^2 + 20f\theta^3 + 30g\theta^4]} + \frac{6d + 24e\theta + 60f\theta^2 + 120g\theta^3 - (\tan\gamma)/r}{[(1/r) + 2c + 6d\theta + 12e\theta^2 + 20f\theta^3 + 30g\theta^4]^2}\right] \tag{4.4}$$

The total transfer angle $\theta_f$ is the sum of the angle between the position vectors $\Psi$ and the number of full revolutions $N$.

$$\theta_f = \Psi + 2\pi N \tag{4.5}$$

### BOUNDARY CONDITIONS

The boundary conditions are set up similarly as in a Lambert targeting problem (and exponential sinusoids). The initial and final position and flight path angle are constrained. Different from the exponential sinusoids, there are constraints on the velocity, this is done using the expression for the derivative of the polar angle. These boundary conditions are shown in Equation 4.6; the function $f$ represents the right-hand side of Equations 4.2 and 4.3.

$$\begin{array}{rclrcl} r_1 & = & r(0) & r_2 & = & r(0) \\ \tan\gamma_1 & = & f(\theta=0, r=r_1) & \tan\gamma_2 & = & f(\theta=\theta_f, r=r_2) \\ \dot{\theta}_1 & = & f(\theta=0, r=r_1) & \dot{\theta}_2 & = & f(\theta=\theta_f, r=r_2) \end{array} \tag{4.6}$$

In order to compute the coefficients for the shape function, the boundary conditions are used. Equations 4.7 and 4.8 are used to solve for the coefficients of the fifth-order polynomial. For the sixth-order polynomial the coefficient $d$ is used to satisfy the time of flight requirement. As the radius shape equation is different, the coefficients $e, f, g$ need to be computed with a different set of equations as seen in Equation 4.9.

$$a = \frac{1}{r_1}, \quad b = -\frac{\tan\gamma_1}{r_1}, \quad c = \frac{1}{2r_1}\left(\frac{\mu}{r_1^3\dot{\theta}_1^2} - 1\right) \tag{4.7}$$

$$\begin{bmatrix} \theta_f^3 & \theta_f^4 & \theta_f^5 \\ 3\theta_f^2 & 4\theta_f^3 & 5\theta_f^4 \\ 6\theta_f & 12\theta_f^2 & 20\theta_f^3 \end{bmatrix} \begin{bmatrix} d \\ e \\ f \end{bmatrix} = \begin{bmatrix} \frac{1}{r_2} - (a + b\theta_f + c\theta_f^2) \\ -\frac{\tan\gamma_2}{r_2} - (b + 2c\theta_f) \\ \frac{\mu}{r_2^4\dot{\theta}_2^2} - (\frac{1}{r_2+2c}) \end{bmatrix} \tag{4.8}$$

$$\begin{bmatrix} e \\ f \\ g \end{bmatrix} = \frac{1}{2\theta_f^6} \begin{bmatrix} 30\theta_f^2 & -10\theta_f^3 & \theta_f^4 \\ -48\theta_f & 18\theta_f^2 & -2\theta_f^3 \\ 20 & -8\theta_f & \theta_f^2 \end{bmatrix} \begin{bmatrix} \frac{1}{r_2} - (a + b\theta_f + c\theta_f^2 + d\theta_f^3) \\ -\frac{\tan\gamma_2}{r_2} - (b + 2c\theta_f + 3d\theta_f^2) \\ \frac{\mu}{r_2^4\dot{\theta}_2^2} - (\frac{1}{r_2} + 2c + 6d\theta_f) \end{bmatrix} \tag{4.9}$$

For $d$ to be optimised for time of flight, boundaries on possible values for $d$ must be found. These boundaries can be set using the condition that $\dot{\theta}$ should be larger than or equal to zero, otherwise the shape methods' equations are invalid. Using Equation 4.3 and the application, the inequality as shown in Equation 4.10 can be derived. The equation is valid when the denominator is positive, if negative the sign changes to a larger than or equal to sign. It can be seen that the only changing 'variable' in this formula is the azimuth angle $\theta$, thus for every azimuth angle between zero and the final azimuth angle a boundary for $d$ can be found. By applying the formula to a series of azimuth angles, bounds can be found for which $\dot{\theta} \geqslant 0$ is valid over the whole trajectory.

$$\frac{-\left(a + b\theta + c(\theta^2+2)\right) - \left(\frac{1}{2\theta_f^6} \begin{bmatrix} \theta^4 + 12\theta^2 \\ \theta^5 + 20\theta^3 \\ \theta^6 + 30\theta^4 \end{bmatrix}^T \begin{bmatrix} 30\theta_f^2 & -10\theta_f^3 & \theta_f^4 \\ -48\theta_f & 18\theta_f^2 & -2\theta_f^3 \\ 20 & -8\theta_f & \theta_f^2 \end{bmatrix} \begin{bmatrix} \frac{1}{r_2} - (a + b\theta_f + c\theta_f^2) \\ -\frac{\tan\gamma_2}{r_2} - (b + 2c\theta_f) \\ \frac{\mu}{r_2^4\dot{\theta}_2^2} - (\frac{1}{r_2} + 2c) \end{bmatrix}\right)}{\left(\theta^3 + 6\theta + \frac{1}{2\theta_f^6} \begin{bmatrix} \theta^4 + 12\theta^2 \\ \theta^5 + 20\theta^3 \\ \theta^6 + 30\theta^4 \end{bmatrix}^T \begin{bmatrix} 30\theta_f^2 & -10\theta_f^3 & \theta_f^4 \\ -48\theta_f & 18\theta_f^2 & -2\theta_f^3 \\ 20 & -8\theta_f & \theta_f^2 \end{bmatrix} \begin{bmatrix} -\theta_f^3 \\ -3\theta_f^2 \\ -6\theta_f \end{bmatrix}\right)} \leqslant d \tag{4.10}$$

### TIME OF FLIGHT

The constraint not yet calculated is the time of flight and is shown in Equation 4.11; using the relation $\mathrm{d}t = \frac{1}{\dot{\theta}}\mathrm{d}\theta$, this expression is straightforward. Note that the function $r(\theta)$ is used, as shown in Equation 4.1.

$$TOF = \int_0^{t_f} \mathrm{d}t = \int_0^{t_f} \sqrt{\frac{r(\theta)^4}{\mu}\left[\frac{1}{r(\theta)} + 2c + 6d\theta + 12e\theta^2 + 20f\theta^3 + 30g\theta^4\right]}\mathrm{d}\theta \tag{4.11}$$

### 3D SHAPE FUNCTION

The inverse polynomials represent 2D shapes, however Wall has proposed a solution to give approximate solutions for three-dimensional problems [11]. In Equation 4.12 the shape function for the third coordinate z is given, with the shape coefficients $a_z, b_z, c_z$ and $d_z$. $q$ is the order of the function and may be any real number. The shape coefficients can be computed using Equations 4.13 and 4.14.

$$z(\theta) = a_z + b_z\theta + c_z\theta^{q-1} + d_z\theta^{q-1} \tag{4.12}$$

$$a_z = z_1, \quad b_z = \dot{z}/\dot{\theta}_1 \tag{4.13}$$

$$\begin{bmatrix} c_z \\ d_z \end{bmatrix} = \frac{1}{\theta_f^q}\begin{bmatrix} q\theta_f & -\theta_f^2 \\ -(q-1) & \theta_f \end{bmatrix}\begin{bmatrix} z_2 - a_z - b_z\theta_f \\ \left(\frac{\dot{z}_2}{\dot{\theta}_2}\right) - b_z \end{bmatrix} \tag{4.14}$$

### REQUIRED THRUST

In order to follow the prescribed trajectory, a certain thrust needs to be achieved; the expression to calculate this is analytical, and shown in Equations 4.15 and 4.16 for the in-plane and out-of-plane thrust acceleration respectively. The total thrust acceleration is shown in Equation 4.17, as the vectors are perpendicular. It is derived using the equations of motion and the known expressions for the derivatives of the polar angle.

$$T_{a_{in}} = -\frac{\mu}{2r^3\cos\gamma}\cdot\frac{6d + 24e\theta + 60f\theta^2 + 120g\theta^3 - (\tan\gamma)/r}{[(1/r) + 2c + 6d\theta + 12e\theta^2 + 20f\theta^3 + 30g\theta^4]^2} \tag{4.15}$$

$$T_{a_z} = \frac{\mu}{r^3}z - [c_z(q-1)(q-2)\theta^{q-3} + d_z q(q-1)\theta^{q-2}]\dot{\theta}^2 + [b_z + c_z(q-1)\theta^{q-2} + d_z q\theta^{q-1}]\ddot{\theta} \tag{4.16}$$

$$T_a = \sqrt{T_{a_{in}}^2 + T_{a_z}^2} \tag{4.17}$$

The required $\Delta V$ can be computed using Equation 4.18 and was derived again using the equation for the thrust level as well as the Tsiolkovsky equation (Equation 2.6).

$$\Delta V = \int_0^{t_f} T_a \mathrm{d}t = \int_0^{\theta_f} \frac{T_a(\theta)}{\dot{\theta}(\theta)}\mathrm{d}\theta \tag{4.18}$$

## 4.2. IMPLEMENTATION

For the implementation of the inverse polynomials the same can be said as for the exponential sinusoids. Due to the existing TUDAT framework, most of the implementation consisted of redefining inputs, outputs and rewriting the contents of the functions with the theory as given in the section before. In terms of input and outputs, the same are used as with exponential sinusoids, thus Table 3.1 and 3.2 are still valid. The exception is that for IP no winding parameter is required, thus not a possible input in TUDAT. The change in number of nodes for Gaussian quadrature was also done for IP, from 16 to 64 nodes.

## 4.3. VERIFICATION

For the verification of the inverse polynomial implementation, it was expected to meet the boundary conditions, including meeting the required time of flight. While the verification was done on various kinds of trajectories, the results for verification shown in this section use the Earth-Mars trajectory with start date MJD 61895, $TOF = 878$ days and $N_{rev} = 1$.

The boundary conditions (BC) consist of the position, velocity and time of flight. The first two are verified with the difference in value as shown for an example trajectory in Table 4.1. The radius, angular velocity and flight path angle represent the BC in the 2D plane. It is seen that there is no error (<1e-17) for the initial conditions and a rounding error for the final conditions. A similar conclusion can be made for the 3D conditions, z-coordinate and velocity in z, and thus the overall position is closely met as well.

Table 4.1: Comparison of initial and final boundary conditions for trajectory from Earth to Mars, start date of MJD 61895, $TOF = 878$ days, $N_{rev} = 1$.

| | **Radius** | | | **Angular Velocity** | | | **Flight Path Angle** | |
|---|---|---|---|---|---|---|---|---|
| | **Δ [AU]** | **Δ [m]** | **Δ [%]** | **Δ [rad/yr]** | **Δ [rad/s]** | **Δ [%]** | **Δ [rad]** | **Δ [%]** |
| initial | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| final | 7.22E-14 | 1.08E-02 | 4.40E-14 | -1.86E-13 | -5.90E-21 | -6.49E-14 | 2.15E-14 | 4.14E-13 |
| | **z-coordinate** | | | **z-velocity** | | | **Position** | |
| | **Δ [AU]** | **Δ [m]** | **Δ [%]** | **Δ [AU/yr]** | **Δ [m/s]** | **Δ [%]** | **Δ [m]** | **Δ [%]** |
| initial | 0.00E+00 | 0.00E+00 | 0.00E+00 | 5.42E-20 | 2.57E-16 | 1.60E-16 | 2.16E-05 | 1.43E-16 |
| final | 4.86E-17 | 7.27E-06 | 9.46E-16 | -2.91E-15 | -1.38E-11 | -6.47E-14 | 0.010792 | 4.39E-14 |

The coefficient $d$ is used to account for the required time of flight, with the inequality equation Equation 4.10 defining the boundaries. To verify the time of flight, a grid of values for $d$ is computed, it is then expected that outside these boundaries no solution is found (NaN). The computed boundaries for the verification trajectory are [-0.0578513, 100]. A grid from -2.0 to 10.0 and resolution of 0.0001 for the value of $d$ is used and the computed TOFs are shown in Figure 4.1. It is seen that the valid results are indeed between the computed boundaries.

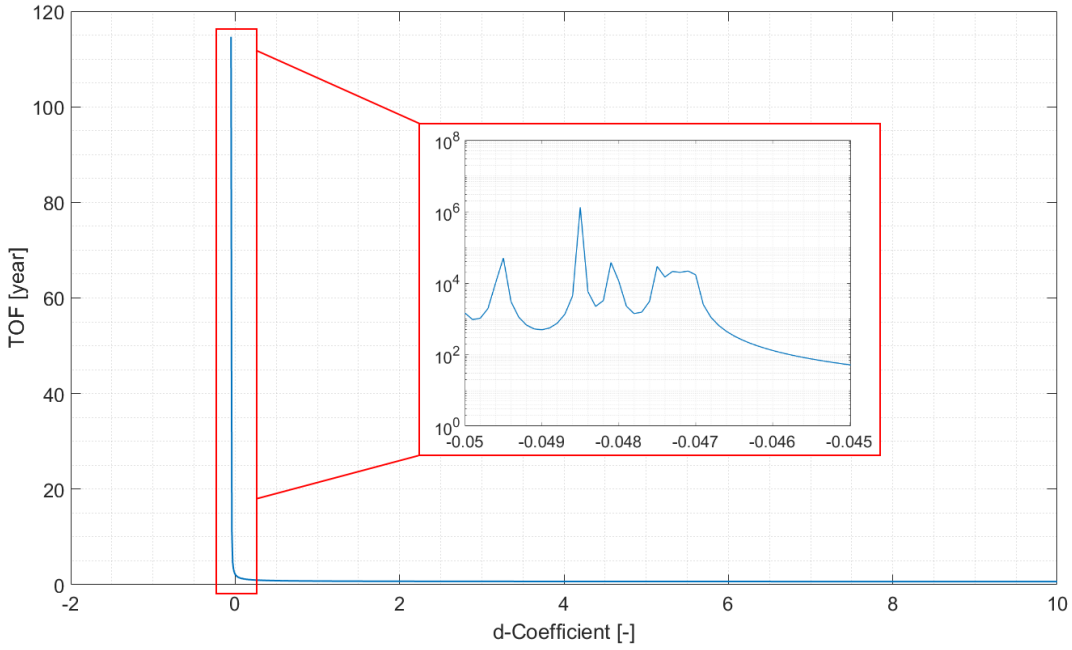



Figure 4.1: Time of flight for given coefficient $d$, zoom-in of area of interest. Earth-Mars trajectory with start date: MJD 61895 (05-04-2028), time of flight: 878 days.

This kind of investigation was done for multiple trajectories and it found they all have the shape as shown in the figure in common, they increase smoothly and exponential-like towards the lower boundary and almost constant otherwise. A zoom-in near the lower boundary is shown with a logarithmic y-scale, it can be seen that the time of flights can become very high and that there is some kind of oscillation occurring. Normally for the desired trajectories such high TOFs are not practical. The oscillation is thus not really a problem as the area of interest is well outside the domain where the TOF becomes excessively high. These long trajectories usually require very high thrust as they shoot out of the solar system and then thus require high $\Delta V$. As they have a very high TOF and $\Delta V$, it is not expected they will become an issue during the case studies, however it should be known that the phenomenon's exists.

## 4.4. VALIDATION

Validation of the inverse polynomials is done using the test case as in Wall's paper [11]. The test case is an Earth-Mars transfer, that is optimised for $\Delta V$. The paper uses the IP to find an initial guess for a direct numerical optimisation method. As such the ranges for input data are provided and results are given for the IP and the direct method. However the data provided is insufficient to reproduce the IP part of the test case, what can be found is the start date by looking at the relative position of the planets, the number of full revolutions and the time of flight, although the latter is specified for the direct method only.

In Figure 4.2 the trajectory as found in the paper and the computed IP trajectory are shown. It can be seen that the start date and time of flight are set-up such that they match. The more irregular lines represent the reference data. It can be seen that the Mars orbit does not exactly align with the orbit defined in TUDAT. The trajectory overall aligns, however there are differences visible likely due to the different Mars orbit and the fact that the paper does not specify whether the trajectory is the IP trajectory or the direct optimised trajectory.

In order to examine whether the computed IP transfer is located close to a $\Delta V$ minimum, use is made of the results shown in Figure 4.3. The start date and time of flight is varied as the shown trajectory in the paper could be the direct optimised one. It can be seen that the minimal $\Delta V$ results are not far from the optimal conditions as specified in the paper.

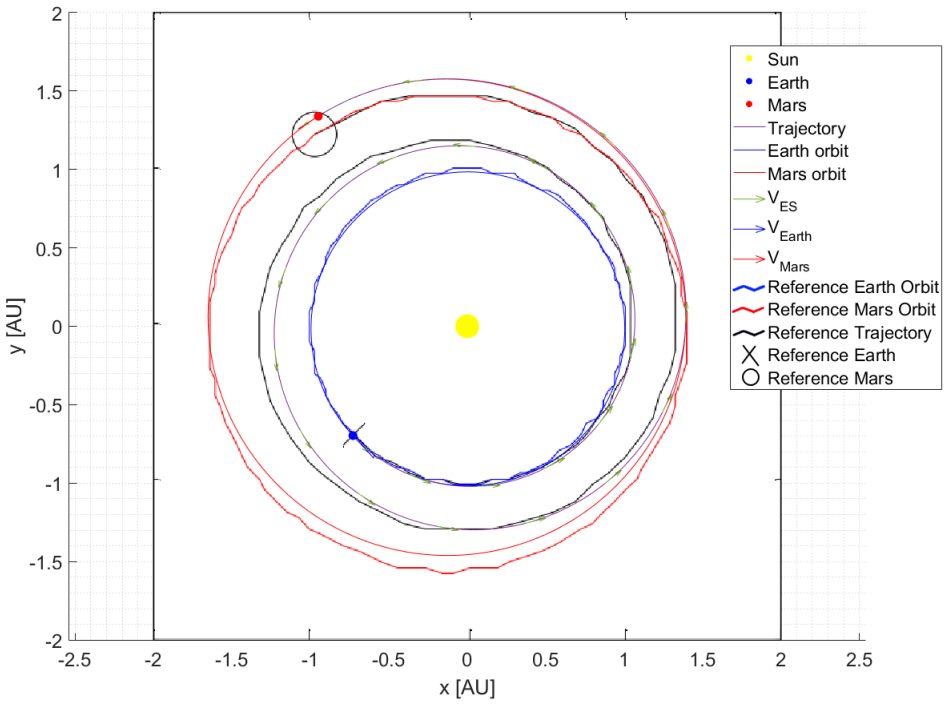



Figure 4.2: Comparison between Earth-Mars trajectory obtained with IP implementation and reference trajectory in [11, Fig. 4]. Start date: MJD 61895 (05-04-2028), time of flight: 878 days. Please note that the wiggling of the reference lines were introduced from the source.

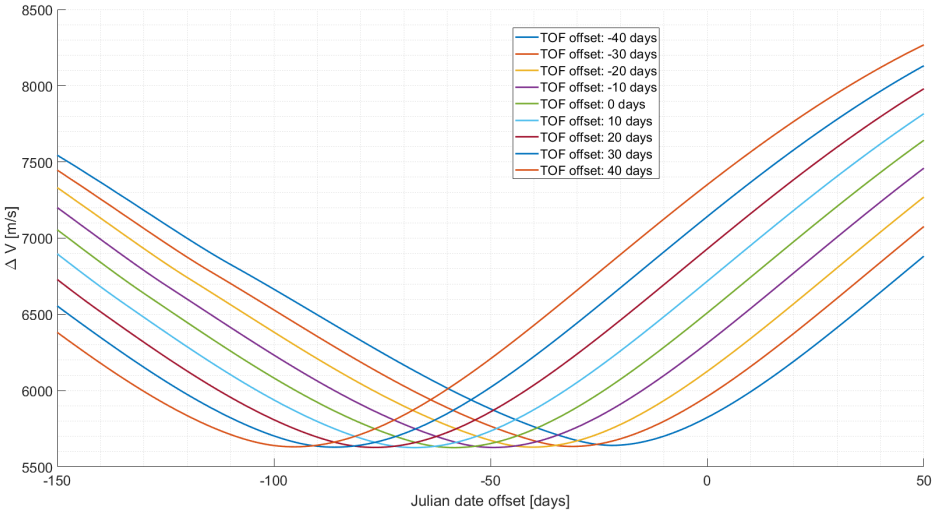



Figure 4.3: $\Delta V$ for Earth-Mars IP trajectories for various initial start dates and time of flights. Nominal start date: MJD 61895 (05-04-2028), nominal time of flight: 878 days.

# 5

# SPHERICAL AND HODOGRAPHIC SHAPING

This chapter combines spherical shaping and hodographic shaping and is briefer compared to the exponential sinusoids and the inverse polynomials chapters as they are already implemented in TUDAT. For spherical shaping an adjustment was made in the existing C++ code which will be shown here as well. Both will have their main shape functions explained and key parameter equations shown. This chapter contains the background information on spherical shaping in Section 5.1 and hodographic shaping in Section 5.2.

## 5.1. SPHERICAL SHAPING

For spherical shaping (SS), the main shape function and its elements will be given first. Secondly the equations for the performance parameters will be provided. Finally an adjustment made to the current implementation of SS in TUDAT will be explained.

### 5.1.1. DESCRIPTION AND SHAPE

A third type of shaping method that will be used is called spherical shaping made by Novak and Vasile [14]. It uses spherical coordinates $r, \theta, \varphi$ which are used in three main functions as seen in Equation 5.1: radial shape function $R(\theta)$, elevation shape function $\Phi(\theta)$, and the time shape function $T(\theta)$. $a_0, ..., a_6$ and $b_0, ...b_3$ represent the coefficients for the radial and elevation function respectively, $D$ represents an intermediate function and $\mu$ represents the standard gravitational parameter. The formulas for the derivatives of the shape functions as well as the computation of the coefficients is shown in Appendix B, as they are quite extensive.

$$\begin{cases} R(\theta) = \frac{1}{a_0+a_1\theta+a_2\theta^2+(a_3+a_4\theta)\cos\theta+(a_5+a_6\theta)\sin\theta} \\ \Phi(\theta) = (b_0+b_1\theta)\cos\theta+(b_2+b_3\theta)\sin\theta \\ T' = \sqrt{\frac{DR^2}{\mu}} \\ D = -r'' + 2\frac{r'^2}{r} + r'\Phi'\frac{\Phi''-\sin\Phi\cos\Phi}{\Phi'^2+\cos^2\Phi} + r(\Phi'^2+\cos^2\Phi) \end{cases} \tag{5.1}$$

Roegiers implemented the spherical shape generation into TUDAT [19], Chambe then re-implemented it into the official public TUDAT version.

### 5.1.2. PERFORMANCE PARAMETERS

#### TIME OF FLIGHT

The time of flight can be computed by integrating the time function T' from the initial to final azimuthal angle as shown in Equation 5.2. Roegiers proposed using a Runge-Kutta integrator as TUDAT has this functionality already, a time-step for $d\theta = \frac{2\pi}{20}$ is proposed resulting in an error of around 14 minutes. This step might be interesting to see if other options for integrator and step are possible or even better.

$$TOF = \int_{\theta_i}^{\theta_f + 2\pi n_r} T' \mathrm{d}\theta \tag{5.2}$$

There are two methods for satisfying the time of flight requirement. Either use a Newton loop for the remaining free coefficient or a method "augmenting the original time evolution". Roegiers investigated both options and concluded that due to a limited amount of documentation on the second option, the Newton loop is used, however this loop can fail to converge in some cases. In TUDAT this was implemented by Chambe using a bisection algorithm to satisfy the time of flight requirement, which need the user to define the bounds manually.

#### CONTROL ACCELERATION

The control acceleration can be computed in the radial ($\hat{e}_r$), normal ($\hat{e}_n$), out-of-plane ($\hat{e}_h$) and tangential ($\hat{e}$) acceleration components with Equation 5.3. The not yet calculated variables such as $a$, $\dot{\theta}$ and $\ddot{\theta}$ can be found with Equation 5.4.

$$\mathbf{u} = \begin{bmatrix} u_t \\ u_n \\ u_h \end{bmatrix} = \begin{bmatrix} \frac{\mu}{r^2}\mathbf{e_r}\cdot\mathbf{e_r} + \ddot{\theta}\mathbf{v}\cdot\hat{e}_t + \dot{\theta}^2\mathbf{a}\cdot\hat{e}_t \\ \frac{\mu}{r^2}\hat{e}_r\cdot\hat{e}_n + \dot{\theta}^2\mathbf{a}\cdot\hat{e}_n \\ \dot{\theta}^2\mathbf{a}\cdot\hat{e}_h \end{bmatrix}, \quad \begin{cases} \dot{\theta} = \frac{1}{T'} \\ \ddot{\theta} = -\frac{T''}{T'^3} \end{cases} \tag{5.3}$$

$$\tilde{v} = \begin{bmatrix} \tilde{v}_r \\ \tilde{v}_\theta \\ \tilde{v}_\varphi \end{bmatrix} = \begin{bmatrix} R' \\ R\cos\Phi \\ R\Phi' \end{bmatrix}, \quad \mathbf{a} = \begin{bmatrix} \tilde{a}_r \\ \tilde{a}_\theta \\ \tilde{a}_\varphi \end{bmatrix} = \begin{bmatrix} R'' - R(\Phi'^2 + \cos^2\Phi) \\ 2R'\cos\Phi - 2R\Phi'\sin\Phi \\ 2R'\Phi' + R(\Phi'' + \sin\Phi\cos\Phi) \end{bmatrix} \tag{5.4}$$

#### REQUIRED $\Delta V$

The amount of $\Delta V$ or velocity change can be computed by integrating the control acceleration from the initial azimuthal to the final azimuthal angle as shown in Equation 5.5.

$$\Delta V = \int_{\theta_i}^{\theta_f + 2\pi n_r} |\mathbf{u}| T' \mathrm{d}\theta \tag{5.5}$$

### 5.1.3. ADJUSTMENT TO SPHERICAL SHAPING IN TUDAT

In Chambe's version the bounds for the free parameters are required to be given by the user. When the shaping method is used with inputs in a grid-search method, a manual input of these bounds is infeasible. Similar as with the inverse polynomials, the bounds will be derived such that they can be computed automatically.

For the time function to yield non-imaginary numbers and thus the trajectory to be valid, the parameter $D$ needs to be positive. Equation 5.6 shows how this parameter is computed.

$$D = -R'' + 2\frac{R'^2}{R} + r'\Phi'\frac{\Phi'' - \sin\Phi\cos\Phi}{\Phi'^2 + \cos^2\Phi} + r(\Phi'^2 + \cos^2\Phi) \tag{5.6}$$

The full derivation is extensive and is given in Appendix B, here the only the essential equation will be given.

The value for the free coefficient $a_2$ for which the parameter $D$ inequality is satisfied is shown in Equation 5.7. Note that all the 'C' variables are comprised of the shape functions and their derivatives, the coefficient matrices and the boundary conditions. It is seen that the azimuth angle $\theta$ is also part of the equation, this means that for each azimuth angle, there is a limit for $a_2$. Therefore the limits need to be evaluated over the full transfer.

$$a_2 < \frac{C_2 C_F + C_A - C_C}{C_2 C_G - C_2\theta^2 + C_B - 2 - C_D + C_E} \tag{5.7}$$

The inequality sign direction depends on the sign of the radius equation (1) (Equation 5.1), the denominator$_1$ of an in-between equation (Equation B.14), with here only the denominator shown (2) (Equation 5.8) and finally the denominator$_2$ in the final inequality (3) (Equation 5.7).

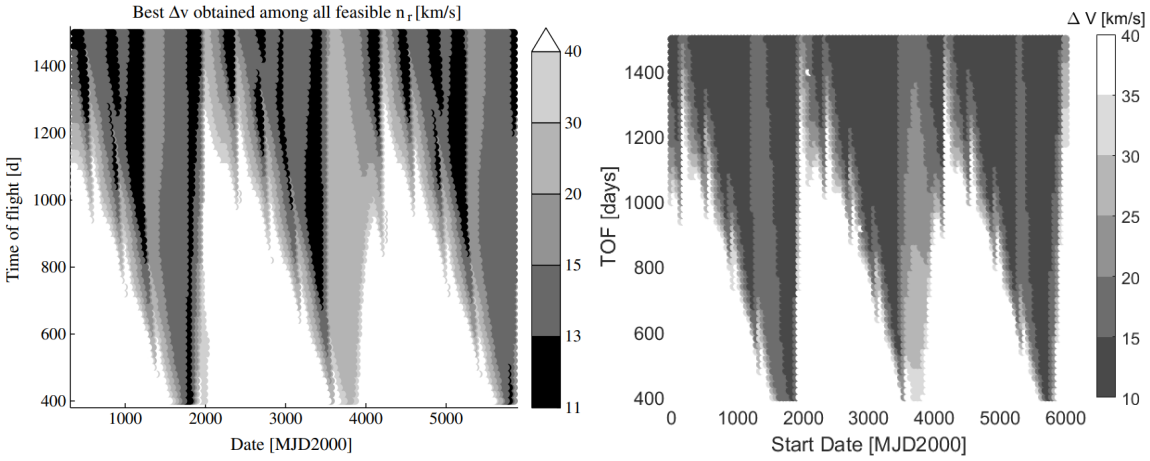
$$C_F - C_G a_2 + a_2 \theta^2 \tag{5.8}$$

For the following combinations of equation values as shown in Equation 5.9, the condition of $a_2$ is as given in Equation 5.7, i.e. 'smaller than'; if not, the sign is reversed, thus it becomes 'larger than'.

$$\begin{cases} (\text{denominator}_1 < 0 \text{ and denominator}_2 > 0 \text{ and } R > 0) \\ (\text{denominator}_1 > 0 \text{ and denominator}_2 < 0 \text{ and } R > 0) \\ (\text{denominator}_1 > 0 \text{ and denominator}_2 > 0 \text{ and } R < 0) \\ (\text{denominator}_1 < 0 \text{ and denominator}_2 < 0 \text{ and } R < 0) \end{cases} \tag{5.9}$$

With all these limits imposed on $a_2$, the algorithm loops over from the initial to the final azimuth angle, where for each angle it computes the boundary $a_2$ and whether it is an upper or lower boundary. This is a similar 'patch' as was made for inverse polynomials, but now with different inequalities. With this addition it is no longer necessary to manually provide the bisection boundaries and allows for its use in grid searches.

In order to verify its function, unit tests were performed which confirmed its working. Another verification that was performed is by redoing a grid search as in literature, to verify the implementation and addition together. As the position of Tempel-1 is used within this method, its correct implementation is inherently verified as well. Figure 5.1a shows the reference data: the required $\Delta V$ for a given start date and time of flight when computing trajectories with spherical shaping. Figure 5.1b gives results for the same computation, but done in TUDAT, with the adjusted spherical shaping method; note the slightly different colour scale. It is seen that the two figures and thus data points are as good as equal, proving the correct implementation by Chambe and the correct functionality of the automatic boundary computation for the free coefficient.




(a) Reference figure, taken from [14, Fig.6a].

(b) Computed with Spherical Shaping in TUDAT with same inputs as reference data.

Figure 5.1: Start date vs time of flight and the resulting best $\Delta V$ obtained among all feasible number of revolutions (0,1 and 2).

## 5.2. HODOGRAPHIC SHAPING

For hodographic shaping, the main shape function and its elements will be given first. Secondly the performance parameter's equations will be provided.

### 5.2.1. DESCRIPTION AND SHAPE

Gondelach and Noomen came up with a shaping method based on velocities rather than position [5, 28, 41]. The position can be retrieved by integrating an analytic velocity profile such that boundary conditions can be satisfied. Both a time-driven model (analytical integration) and a polar-angle-driven model (numerical integration) are possible for the velocity. Note that this method also allows for three-dimensional trajectories.

The velocity profile $V(t)$ is as shown in Equation 5.10, with $c_i$ the coefficients and $v_i$ the base functions. The base functions can be a constant, power function, sine, cosine or a power times a trigonometric function. The number of base functions depends on the boundary conditions; if the number of base functions is equal to the number of boundary conditions, base functions can be added or they can be added for even more for more degrees of freedom. The boundary conditions that were considered are initial and final position and velocity, and the time of flight.

$$V(t) = \sum_{i=1}^{n} c_i v_i(t) \tag{5.10}$$

It was found that the time-driven model performed best for trajectories to the outer planets, and the polar-driven works best for missions to the inner planets. A similar extension can probably be made for orbit transfers around Earth. Depending on how many degrees of freedom are used the computation time is low (1.6 ms) or higher (1-2 s) per trajectory. The method implemented in TUDAT by Chambe is the time-driven model, and as such further equations will be for that version.

The three velocities radial $V_r$, transverse/polar angle $V_\theta$ and axial $V_z$ are shape functions and are dependent on time. The radial distance, polar angle and axial distance can be retrieved by integration of the velocity function as seen in Equation 5.11. While both the radial and axial distance can be computed analytically, the polar angle is retrieved numerically.

$$\begin{cases} r(t) = r_0 + \int_0^t V_r \mathrm{d}t \\ z(t) = z_0 + \int_0^t V_z \mathrm{d}t \\ \theta(t) = \theta_0 + \int_0^t \frac{V_\theta}{r} \mathrm{d}t \end{cases} \tag{5.11}$$

#### CHARACTERISTICS

This method is different as it shapes the velocity profiles instead of the position profiles. This method can satisfy multiple boundary conditions as well as solve three-dimensional problems. A large advantage is its relative simplicity, while still being relatively versatile. A main thing to consider is that the default base functions have to be determined already, but possibly there exist other combinations of functions which yield better results.

### 5.2.2. PERFORMANCE PARAMETERS

#### THRUST

The thrust can be expressed analytically, by assuming the new parameters as in Equation 5.12 and entering them into the EOM, the thrust can be expressed as seen in Equation 5.13. All parameters in this representation can be found analytically.

$$\begin{cases} \ddot{r} \equiv \dot{V}_r, \quad \ddot{z} \equiv \dot{V}_z \\ \ddot{\theta} = \frac{d}{dt}\dot{\theta} = \frac{d}{dt}\left(\frac{V_\theta}{r}\right) = \frac{\dot{V}_\theta}{r} - \frac{v_\theta V_r}{r} \end{cases} \tag{5.12}$$

$$\begin{cases} f_r = \dot{V}_r - \frac{V_\theta^2}{r} + \frac{\mu}{s^3} r \\ f_\theta = \dot{V}_\theta + \frac{V_\theta V_r}{r} \\ f_z = \dot{V}_z + \frac{\mu}{s^3} z \end{cases} \quad \rightarrow \quad f = \sqrt{f_r^2 + f_\theta^2 + f_z^2} \tag{5.13}$$

### REQUIRED $\Delta V$

This expression for the thrust needs to be integrated numerically to come up with a value for the required $\Delta V$ as seen in Equation 5.14.

$$\Delta V = \int_0^{t_f} f \, dt \tag{5.14}$$

### USED VELOCITY BASE FUNCTIONS

Up to now the velocity base functions have not yet been discussed. There exist several base functions which are presented in Table 5.1. These must be differentiable and integrable analytically and their summation should be able to satisfy the boundary conditions.

Table 5.1: Proposed base functions, their derivatives and integrals, $u$ is either $t$ or $\theta$ [5, Table 2].

| **Base function** | $v(u)$ |
|---|---|
| Constant | 1 |
| Power | $u^n$ |
| Sine | $\sin(nu)$ |
| Cosine | $\cos(nu)$ |
| Power times sine | $u^n \sin(nu)$ |
| Power times cosine | $u^n \sin(nu)$ |

The independent variables $(t, \theta)$ should be normalised, such that the magnitude of the functions is similar and as a consequence the coefficients as well. Periodic functions for a scaled function results in a low number of revolutions, thus special scaling should be applied. At the basic level, the velocity function needs three base functions in order to satisfy the boundary conditions, then an additional function can be used for satisfying a certain thrust acceleration or for more degrees of freedom. While any function can be used, as already highlighted before, some functions can accommodate the boundary conditions more easily.

In TUDAT hodographic shaping was already implemented, as such the default settings were used, in Table 5.2 these are presented; the abbreviations correspond to the functions as defined in Table 5.3.

Table 5.2: Preferred base function for hodographic shaping [5, Table 11].

| **Method** | **DoF** | **Radial** | **Transverse** | **Axial** |
|---|---|---|---|---|
| Time driven | 0 | CPCos | CPCos | CosN5P3CosN5P3SinN5 |

Table 5.3: Abbreviations for the velocity base functions [5, Table 3].

| **Base function** | **Abbreviation** | **Base function** | **Abbreviation** |
|---|---|---|---|
| 1 | C | $\cos(N \cdot 2\pi t)$ | CosN |
| $t$ | Pow | $t \sin(2\pi t)$ | Psin |
| $t^2$ | Pow2 | $t^2 \sin(2\pi t)$ | P2Cos |
| $t^5$ | Pow5 | $t^X \cos(2\pi t)$ | PXCos |
| $\sin(2\pi t)$ | Sin | $t^3 \sin(1.5 \cdot 2\pi t)$ | P3Sin15 |
| $\sin(0.5 \cdot 2\pi t)$ | Sin05 | $t^6 \cos[(N + 0.5) \cdot 2\pi t)]$ | P6CosN5 |
| $\cos(2\pi t)$ | Cos | $t^X \cos(0.5 \cdot 2\pi t)$ | PXCos05 |

# 6

# Interplanetary Transfer: Earth-Tempel-1

The first main transfer that will be investigated is the Earth-Tempel-1 transfer. First the transfer case will be explained, with its description, inputs and outputs (Section 6.1). Then the four shape-based methods will be applied to the problem in a grid-search manner and discussed individually; the results will be compared and general conclusions will be drawn (Section 6.2). Finally the four methods will be applied and free coefficients optimised to find the absolute minima, the results will then be compared and the best results overall will be shown (Section 6.3).

## 6.1. Transfer Case Set-up

To analyse the performance of the shape-based methods, a transfer case needs to be defined with an outline, inputs and outputs. This is done in the aforementioned order in the following subsections.

### 6.1.1. Outline

All four methods will be applied to the Earth-Tempel-1 transfer. This interplanetary transfer is a circular-to-elliptical orbit rendez-vous transfer. Only the interplanetary transfer is assumed, manoeuvres in their spheres of influence are not accounted for. The case was selected for the target body orbit being elliptical, and as three of the methods are implemented for rendez-vous conditions.

The shape-based methods will first be analysed using a grid search for a list of inputs as described in Subsection 6.1.2, with the outputs as specified in Subsection 6.1.3. Conclusions can be made and behaviour reported per method, as well as the establishment of areas of interest with respect to the used inputs. The areas of interest are expected to be regions with desirable results for the performance parameters, for example low values for $\Delta V$. Given the areas of interest optimisers are applied for finding the inputs resulting in absolute optimal performance parameters, as a grid search only provides the best performance for the resolution of used inputs.

### 6.1.2. Inputs

For the grid search there are three main inputs which each shape-based method uses. First, the start date will decide the position and velocity of the departure body, Earth. Second, the required time of flight of the trajectory will decide the position and velocity of the arrival body Tempel-1. Third, the number of revolutions of the trajectory decides the number of full 360 degrees revolutions the spacecraft makes around the central body.

The ranges used for the various inputs are given in Table 6.1, they are selected for the following reasons. The start date is chosen to be during the writing of this thesis (2020) and to be more than once the synodic period (446 days) of Earth and Tempel-1. As the orbit of Tempel-1 is elliptic, and the synodic period is mostly useful for two circular orbits, the start date range was set to be five years (2024), this to ensure all possible relative positions were examined. With a resolution of 40 days a year is split into 10 points for a decent spread. By not using 36 days, the same position of Earth is not repeated. The time of flight was chosen to be at maximum 10 times the Hohmann transfer time of 890 days as it is tied to the number of revolutions. The resolution of 100 days ensures a good spread; a lower step size is not needed for the grid-search and for limiting the computation time. The number of revolutions was chosen to be similar to those used in literature and to limit the time of flight.

Table 6.1: Input definition and resolution used for grid search of the Earth-Tempel-1 transfer.

| **Input Name** | **Values** | **Unit** | **Resolution** | **Unit** |
|---|---|---|---|---|
| Start Date | [7305, ..., 9105] | [MJD2000] | 40 | [days] |
| Time of Flight | [100, . . . , 8900] | [days] | 100 | [days] |
| Number of Revolutions | [0, . . . , 2] | [-] | 1 | [-] |

### 6.1.3. OUTPUTS

The performance parameters are the required $\Delta V$ and thrust for the computed trajectories. The former is a single value per transfer, while the thrust is a continuous function. For each computed trajectory, its state is saved as well in 100 steps, such that peculiar results can be investigated further and illustrations can be made.

For the grid search the $\Delta V$ and the peak thrust acceleration (maximum thrust acceleration of the function throughout the trajectory) will be used as performance parameters, to analyse the search space and shape-based method performance. For the optimisation in the areas of interest the $\Delta V$ will be optimised for, while the peak thrust will simply follow from it, and will be further investigated if it shows high results.

## 6.2. RESULTS GRID SEARCH

In the following sub-sections the grid search has been applied using all four shape-based methods as described before. Each sub-section starts with a four-panel plot showing the results per number of revolutions and the compilation of the best for all revolutions. These are then discussed individually and compared with each other. Then the best results are looked at in more detail with respect to $\Delta V$ resolution. Finally a typical trajectory is shown together with its thrust profile to highlight the general characteristics. This section concludes with a sub-section which combines the results for all shape-based methods per number of revolution and a compilation of all the best results. These are then discussed individually and compared with each other.

### 6.2.1. RESULTS EXPONENTIAL SINUSOIDS

In Figure 6.1 the results for the $\Delta V$s for the various start dates, time of flights and number of revolutions are provided.

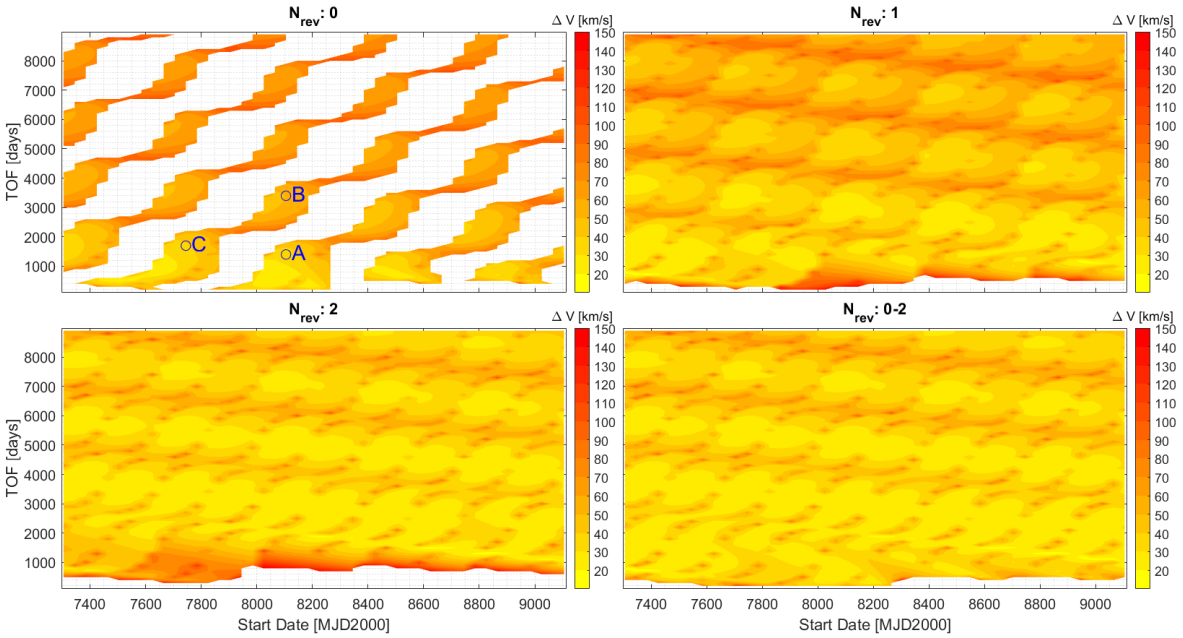



Figure 6.1: $\Delta V$ for Earth-Tempel-1 Rendez-Vous for 0 to 2 revolutions using Exponential Sinusoids Shaping. Three trajectories of interest marked with A, B and C.

First of all it is seen that not all grid points have results (white space), as both no valid solutions (no trajectory possible) or infeasible ones ($\Delta V$>150 km/s) have been removed. In literature the $\Delta V$ scale usually ends at 40 km/s, however here it was set to 150 km/s to be able to observe possible patterns and be able to investigate the full design space.

A kind of repeating wave-like pattern is visible in all graphs; this is the result of similar relative positions resulting in similar required $\Delta V$s. Whereby certain combinations of start date and time of flight (TOF) result in these similar relative positions. This can be confirmed by observing that the periodicity in start date is close to the orbital period of Earth (365 days), and the periodicity in TOF is close to the orbital period of Tempel-1 (2016 days). To maintain a similar $\Delta V$ there is a slight decrease in TOF for increasing start date which is 365 days. The periodicities and relative position can be confirmed by inspecting the trajectories with similar $\Delta V$ as indicated in the top left plot with A, B and C. The trajectories themselves are shown in Figure 6.2. The periodicity of 2016 days in TOF is seen between trajectories A and B, the periodicity of 365 days in start date and TOF is seen between trajectories A and C. All three trajectories have a very similar relative position.

For the results with zero revolutions around the central body (Figure 6.1), the lowest attained $\Delta V$s are located between a time of flight of 200 and 1000 days. It can be seen that for increasing number of revolutions, the TOFs that result in the lowest $\Delta V$s, increase. This makes sense considering that for the spacecraft to achieve the same TOF with more revolutions, the spacecraft needs to travel faster thus requiring more propellant. Optimally the spacecraft would spend as little time and with as little force thrusting, thus optimally the trajectory should be very similar to the solution of the two-body problem (coasting). The results for $N_{rev} = 0$ show that the TOFs in the low $\Delta V$ regions are similar to a Hohmann transfer time of 890 days in which the trajectory is from Earth to the apocenter of Tempel-1. It is seen that as the TOF is increased, the general $\Delta V$ values increase, as the spacecraft are required to spend more $\Delta V$ to slow down to adhere to an unnatural transfer orbit. This phenomenon can be seen in Figure 6.2 with trajectory B.

When the results for all full number of revolutions are grouped and the inputs with lowest $\Delta V$s are kept as shown in the bottom right plot (Figure 6.1), the patterns become pronounced and the lowest regions mostly from the number of revolutions of 2 are shown.

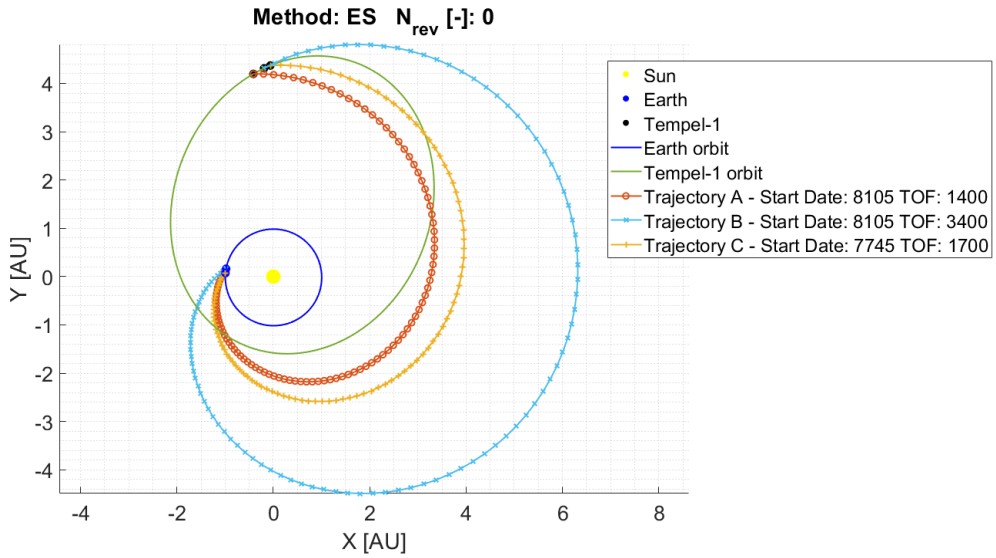



Figure 6.2: Trajectories for Earth-Tempel-1 rendez-vous, using the inputs given in the figure title and legend and exponential sinusoids shaping. The 'Start Date' and 'TOF' are expressed in MJD2000 and in days respectively.

Originally the exponential sinusoid method is only for two-dimensional problems and only considers matching the initial and final positions as well as the provided TOF. As seen in the previous chapter in order to standardise it to the other methods the calculation of the required $\Delta V$ is the sum of the $\Delta V$ required for the transfer and the $\Delta V$ to match the velocity at the initial and target body in three dimensions. This included $\Delta V$ might cause it to stand out incorrectly, compared to the other method's patterns.

Figure 6.3 shows the same plot as seen in the bottom left plot of Figure 6.1, but enlarged and a more useful range for the $\Delta V$ (0-40 km/s) that could be practically used. The same conclusions can be made as before, however it is seen that the most lowest range of 10-20 km/s only has a few results in the bottom left and bottom right valleys.

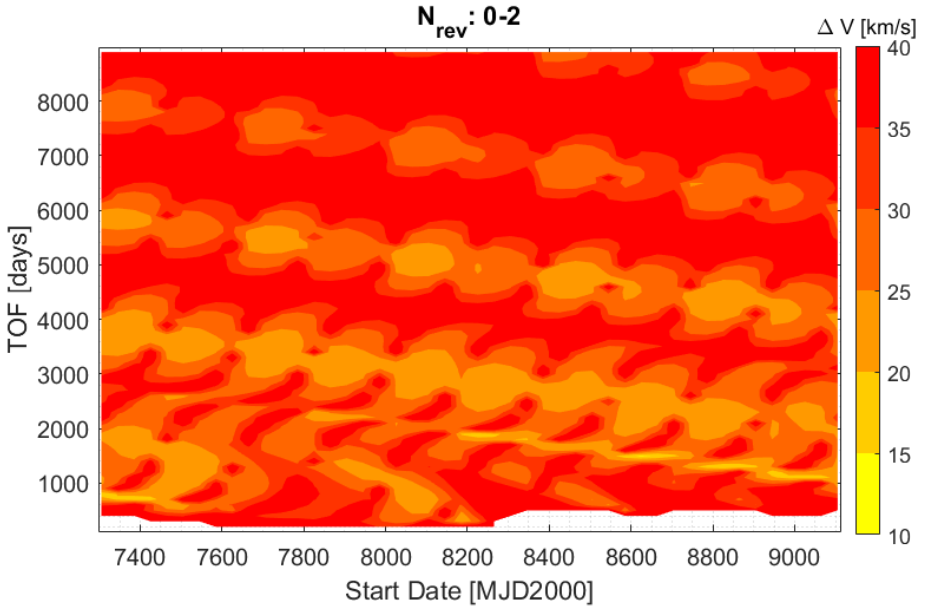



Figure 6.3: Minimum $\Delta V$ for Earth-Tempel-1 Rendez-Vous for 0 to 2 revolutions using Exponential Sinusoids Shaping.

Figure 6.4 shows the top and side view of an example trajectory, the used inputs are given in the figure title and lies in the low $\Delta V$ region. As discussed before the trajectory does not align with the trajectory resulting a velocity mismatch especially in z. This shows the limited usefulness of exposins for 3D applications. The thrust profile required for such a trajectory is shown in Figure 6.5; note that this does not include the required thrust for velocity matching. It can be seen that the required thrust levels (0.4 mm/s$^2$) are similar yet slightly higher than those already achieved in low-thrust missions.

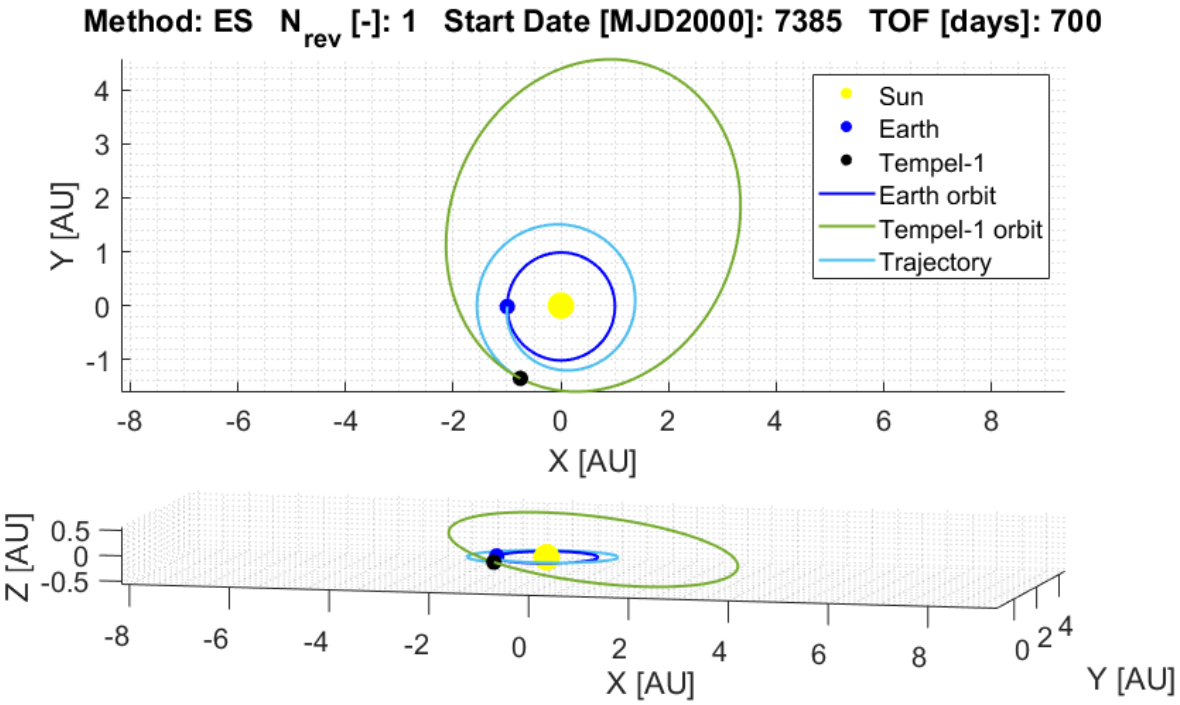



Figure 6.4: Trajectory for Earth-Tempel-1 rendez-vous, using the inputs given in the figure title and exponential sinusoids shaping.

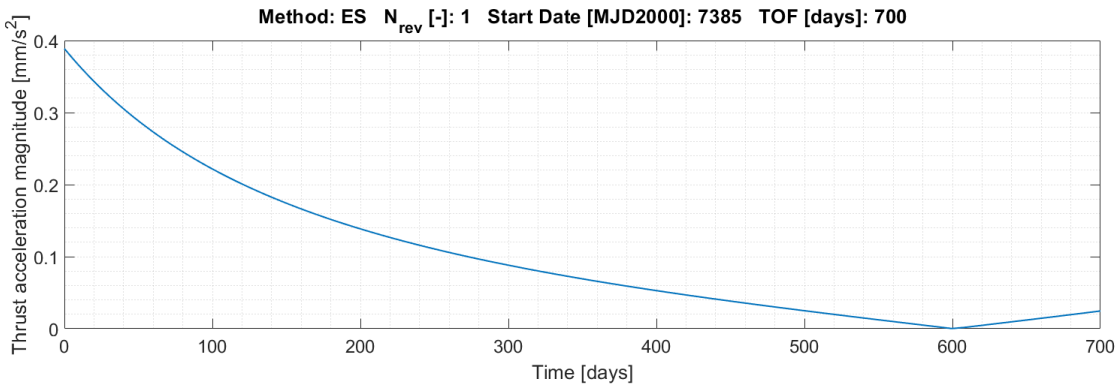



Figure 6.5: Thrust profile required for the trajectory as shown in Figure 6.4 and the inputs as defined in the figure title.

### 6.2.2. RESULTS INVERSE POLYNOMIALS

In Figure 6.6 the results for the $\Delta V$s for the various start dates, time of flights and number of revolutions are provided.

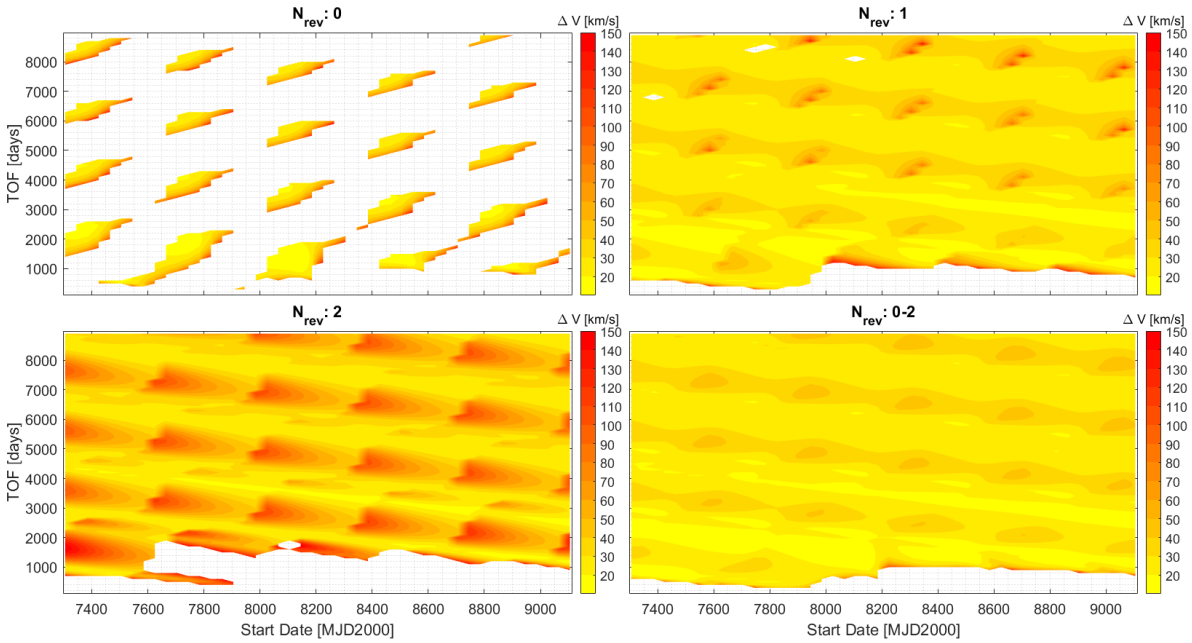



Figure 6.6: $\Delta V$ for Earth-Tempel-1 Rendez-Vous for 0 to 2 revolutions using Inverse Polynomial Shaping.

First of all it is seen that not all grid points have results (white space), as both no valid solutions (no trajectory possible) or infeasible ones ($\Delta V$>150 km/s) have been removed. The repeating pattern with valleys and ridges as seen with the results for the exponential sinusoid shaping are seen here as well. This was expected as the observations about the relative position of the bodies and TOFs are still applicable. The phenomena of optimal results near the TOF similar to the Hohmann flight times can be observed here as well. The smaller number of available results for the zero revolution case can be observed here as well. It is seen that the increase of number of revolutions leads to an increase of solutions for which a low $\Delta V$ is present.

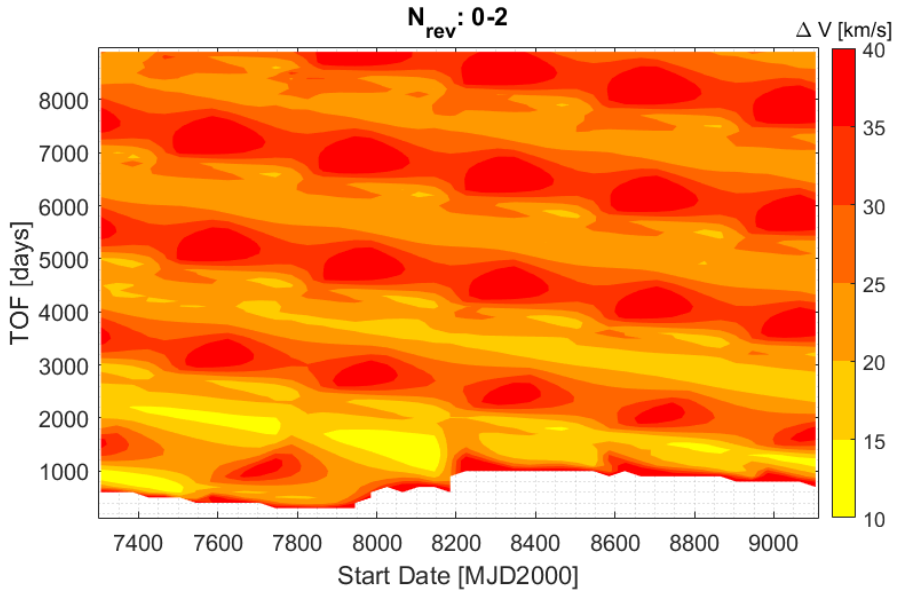



Figure 6.7: Minimum $\Delta V$ for Earth-Tempel-1 Rendez-Vous for 0 to 2 revolutions using Inverse Polynomial Shaping.

If the results are combined for all number of revolutions, it becomes clear that lowest regions in $\Delta V$ are mostly taken from the $N_{rev} = 1$ results. The required $\Delta V$s in the optimal valleys (<20 km/s) are lower than those found with the exponential sinusoids (<40 km/s) and the optimal valleys themselves are larger; this would indicate an overall better performance for the inverse polynomials, for the used transfer problem.

Figure 6.7 shows the plot combining the minimum $\Delta V$ for all revolutions, but enlarged and with a more useful range for $\Delta V$. It can be seen the lowest region of 10-15 km/s is mostly concentrated for low TOF.

Figure 6.8 shows the top and side view of an example trajectory, the used inputs are given above the figure and lies in the low $\Delta V$ region. It can be seen that in the early phase of the trajectory, it spins the most outward, while the plane change occurs more slowly over time and increases more rapidly at the end. It makes sense that this trajectory is low in $\Delta V$ cost as the optimal velocity changes are in-plane with higher velocities, close to the central body and out-of-plane with lower velocities, further from the central body. This is reinforced by the thrust profile and the inclination of the transfer shown in Figure 6.9: larger accelerations at the start and lower accelerations over time afterwards for the plane change and an increase at the end. This profile resembles the use of 'impulsive' shots as with a Hohmann transfer. Compared to flown missions, the thrust level here of 2 mm/s$^2$ is very high, reducing the practical usefulness of the optimal result.

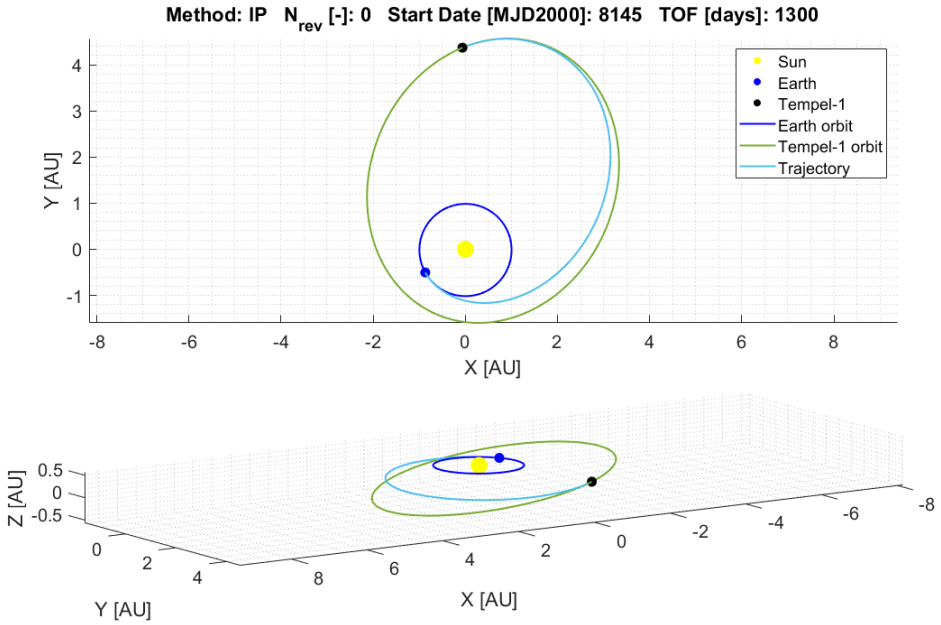



Figure 6.8: Trajectory for Earth-Tempel-1 rendez-vous, using the inputs given in the figure title and inverse polynomials shaping.

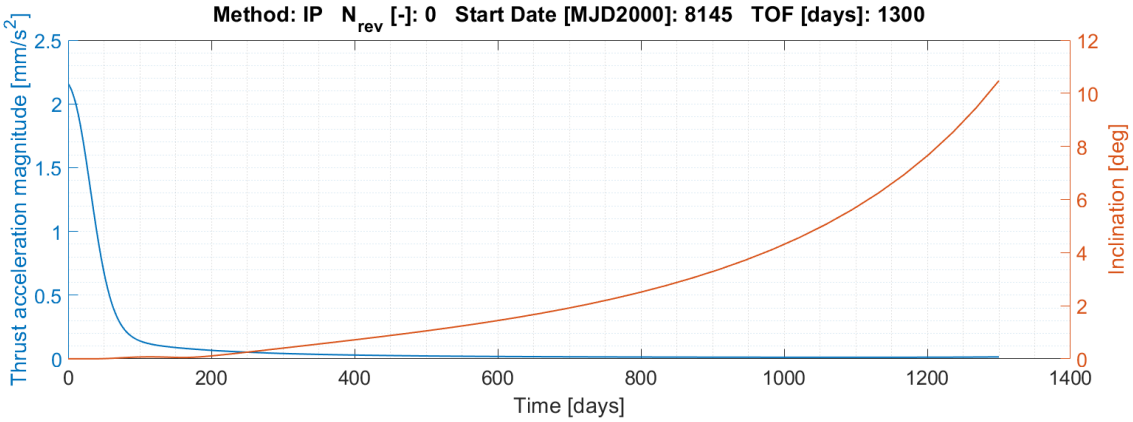



Figure 6.9: Thrust profile required for the transfer trajectory (Left y-axis, blue). Inclination of the transfer trajectory over time (Right y-axis, red). The transfer trajectory is as shown in Figure 6.8 and the inputs as defined in the figure title.

### 6.2.3. RESULTS SPHERICAL SHAPING

In Figure 6.10 the results for the $\Delta V$s for the various start dates, time of flights and number of revolutions are provided.

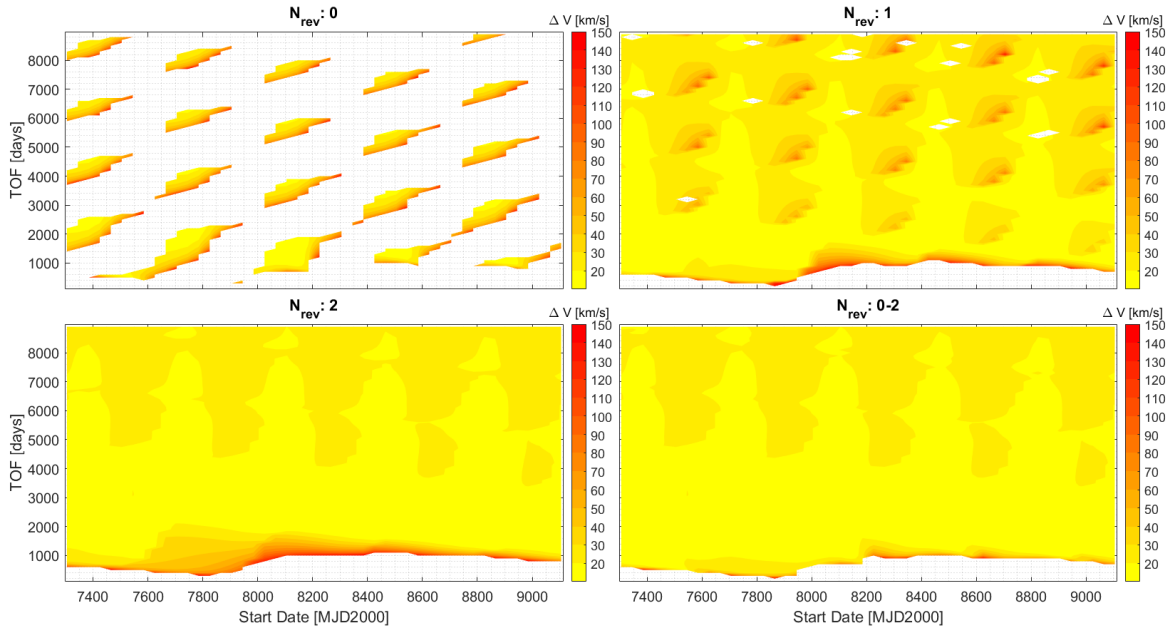



Figure 6.10: $\Delta V$ for Earth-Tempel-1 Rendez-Vous for 0 to 2 revolutions using Spherical Shaping.

For $N_{rev} = 0$, the general wave-like pattern can be observed again. The phenomena where the best results (smallest $\Delta V$) are near the Hohmann time of flights can be observed. The related effect of increasing number of revolutions and increasing time of flight can be observed as well. For $N_{rev} = 1,2$ and the used $\Delta V$ levels, the pattern is more difficult to notice. The repeating pattern in TOF and start date can be seen in the top region, with the higher time of flights.

When all data is combined, it would seem that the data mostly comes from $N_{rev} = 1,2$. However the low $\Delta V$ regions come from the zero revolution data and the gaps are filled with the higher-revolution plots. When compared to the previously seen methods, spherical shaping overall has lower $\Delta V$ trajectories, suggesting it has a better performance for a given geometry of the transfer problem.

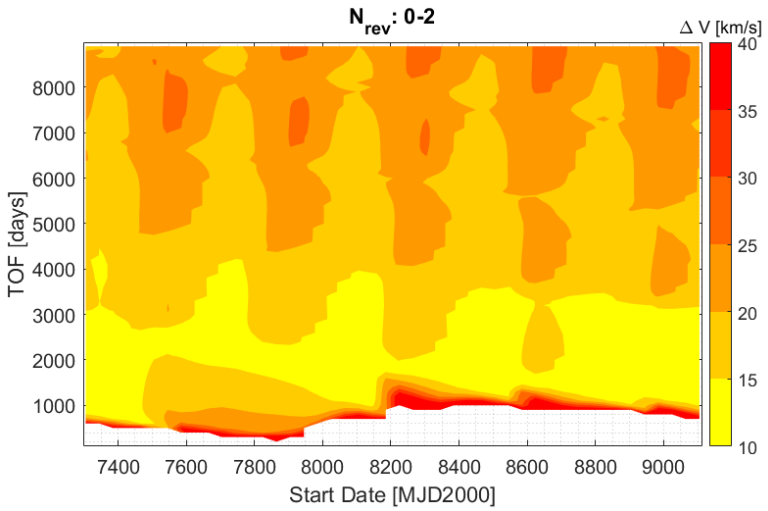



Figure 6.11: Minimum $\Delta V$ for Earth-Tempel-1 Rendez-Vous for 0 to 2 revolutions using Spherical Shaping.

Figure 6.11 shows the plot combining the minimum $\Delta V$ for all revolutions, but enlarged and with a more useful range for $\Delta V$. Overall the $\Delta V$s are low; it can be seen that the lowest region of 10-15 km/s is larger when compared to the previously seen methods. This region also seems to be mostly influenced by the start time and thus the relative positions of Earth and Tempel-1.

Figure 6.12 shows the top and side view of an example trajectory, the used inputs are given above the figure and lie in the low $\Delta V$ region. The same phenomena as with the trajectory shown in Figure 6.8 can be observed here; first two large in-plane changes and then the planar change over time.

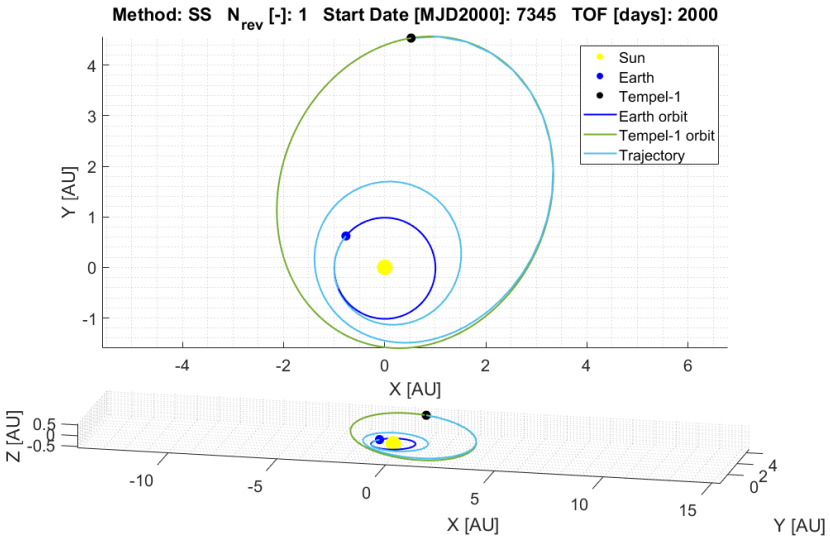



Figure 6.12: Trajectory for Earth-Tempel-1 rendez-vous, using the inputs given in the figure title and spherical shaping.

The thrust acceleration, inclination, radius and velocity of the trajectory over time are shown in Figure 6.13. The first thrust peak is reflected in a change in inclination and an increase in orbit radius. The second thrust peak causes the second large change in orbit radius as well the increase of inclination. It is seen the inclination increases gradually and more evenly compared to inverse polynomials, with a more sudden increase during the thrust peaks. The increases of the orbit radius at closer distance to the Sun as well as at higher velocities reinforces the idea of when it is optimal to perform certain manoeuvres as discussed with the inverse polynomials. Compared to the inverse polynomials, the spherical shaping method combines both in-plane and out-of-plane manoeuvres during the peaks; the combination could be the reason for higher $\Delta V$s, yet lower overall thrust level. Comparing the peaks with the achievable thrust levels (0.37 mm/s$^2$), it can be seen that this thrust profile would be feasible.

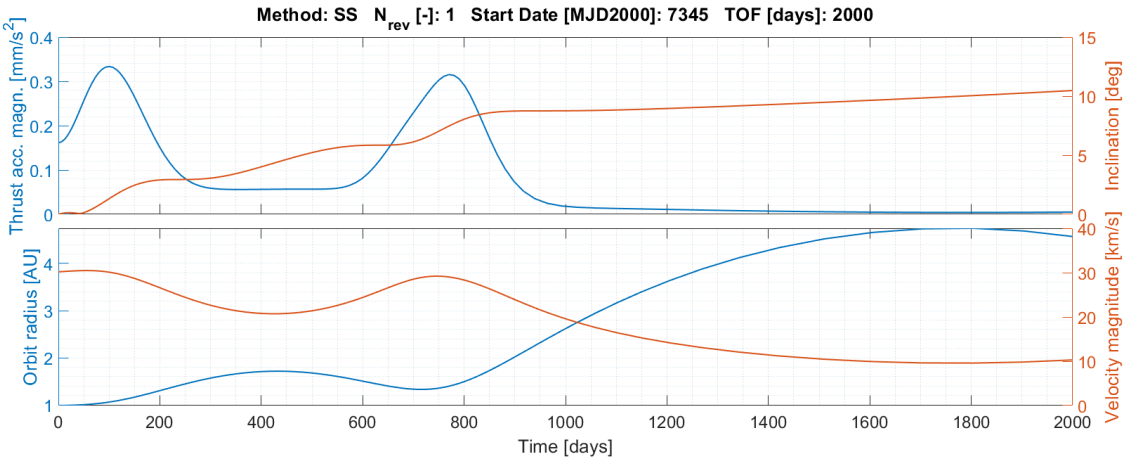



Figure 6.13: Thrust profile required for the transfer trajectory (Top left y-axis, blue). Inclination of the transfer trajectory over time (Top right y-axis, red). Orbital radius of the transfer trajectory over time (Bottom left y-axis, blue). Velocity magnitude of the transfer trajectory over time (Bottom right y-axis, blue). The transfer trajectory is as shown in Figure 6.12 and the inputs as defined in the figure title.

### 6.2.4. RESULTS HODOGRAPHIC SHAPING

As seen in Figure 6.14, hodographic shaping yields results for all given inputs, however not all are feasible. The latter can be seen already by the extremely high required $\Delta V$s (>150 km/s), however they are infeasible due to their close passing of the Sun (< solar radius) or their complete reversal of flight direction around the central body. These infeasible trajectories can be filtered out by their high $\Delta V$ cost and their very high peak accelerations. After such filtering the results can be seen in Figure 6.14.

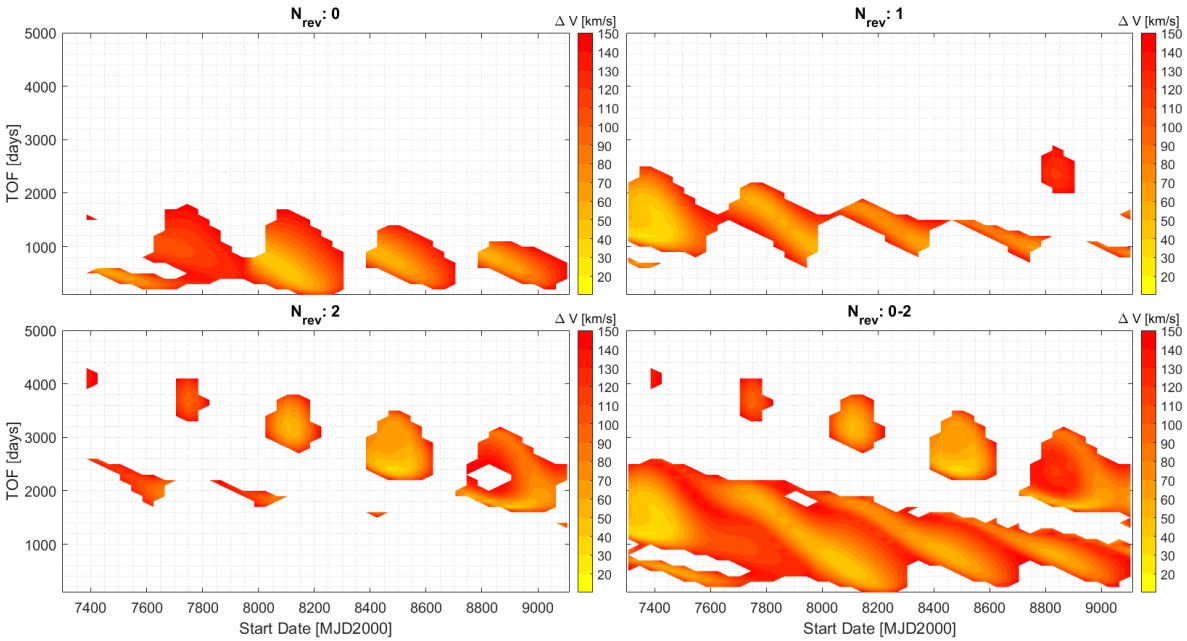



Figure 6.14: $\Delta V$ for Earth-Tempel-1 Rendez-Vous for 0 to 2 revolutions using Hodographic Shaping.

The remaining results show that there are valleys for the $\Delta V$ near the Hohmann flight-times as was seen with the other shape-based methods. However the found $\Delta V$ values for the inputs are quite high (>40 km/s).

It is seen that many results are simply infeasible, so while the hodographic shaping method can compute a fitting trajectory they can be quite infeasible, which can be attributed to the nature of the method. First of all its shape function is the velocity function, and second it is a superposition of varying base functions. The advantage of this method is its flexibility in fitting results and allowing for many different combinations for its base functions. However in this case the default TUDAT base functions have been used, without any tuning, and the results appear less attractive.

When the results for all number of revolutions are combined, it can be seen that most of the optimal results are from the plot with $N_{rev} = 1$. Also the repeating wave pattern can be observed.

Figure 6.15 show the plot combining the minimum $\Delta V$ for all revolutions, but enlarged and with a more useful range for $\Delta V$ and time of flight. Overall the $\Delta V$s are high, likely attributable to the non-optimal base shape functions. Only a small region with 25-30 km/s can be seen for low start date.

Figure 6.16 shows the top and side view of an example trajectory, the used inputs are given in the figure title and lie in the low $\Delta V$ region. It can be seen that the hodographic trajectory spirals far outward in-plane, at the middle and end it performs its out-of-plane manoeuvre. Its pronounced trajectory changes can be seen in the thrust profile in Figure 6.17. It is seen that the thrust levels are relatively acceptable, and more spread over the full flight time compared to the other methods that had only peaks. The peak at the end, is reflected in the rapid change of inclination as seen in the trajectory. Both the consistent thrust acceleration level and peak would explain its high $\Delta V$.

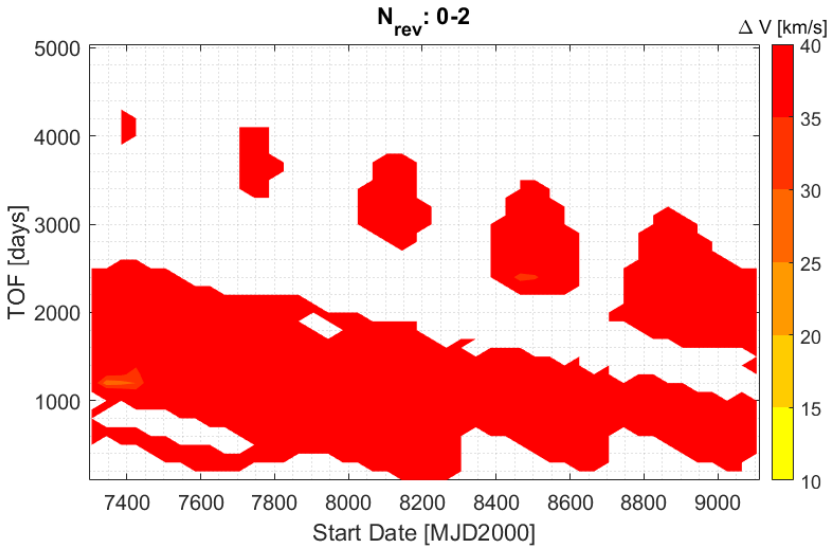



Figure 6.15: Minimum $\Delta V$ for Earth-Tempel-1 Rendez-Vous for 0 to 2 revolutions using Hodographic Shaping.

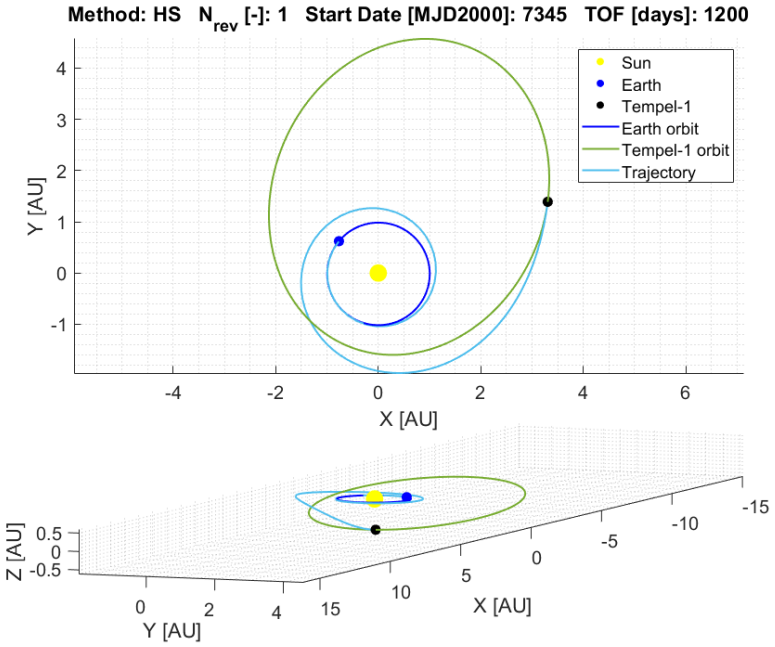



Figure 6.16: Trajectory for Earth-Tempel-1 rendez-vous, using the inputs given in the figure title and hodographic shaping.

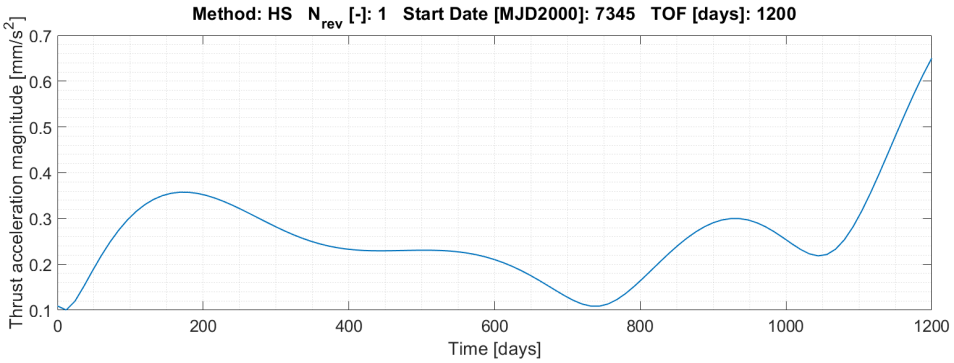



Figure 6.17: Thrust profile required for the trajectory as shown in Figure 6.16 and the inputs as defined in the figure title.

### 6.2.5. GENERAL RESULTS OF THE GRID SEARCH

The results have been inspected per method and per number of revolutions. In this section the results will be compared to each other and general conclusions will be made.

#### GENERAL RESULTS PER NUMBER OF REVOLUTIONS

In Figure 6.18 the results of the $\Delta V$ are shown per number of revolutions (three plots) and per method (four colours). Figure 6.19 show the results in a similar way, however now only the method with the lowest $\Delta V$ for a given input is shown, using the four colours. Both plots give a clearer picture on the overall performance of the four shape-based methods with respect to each other.

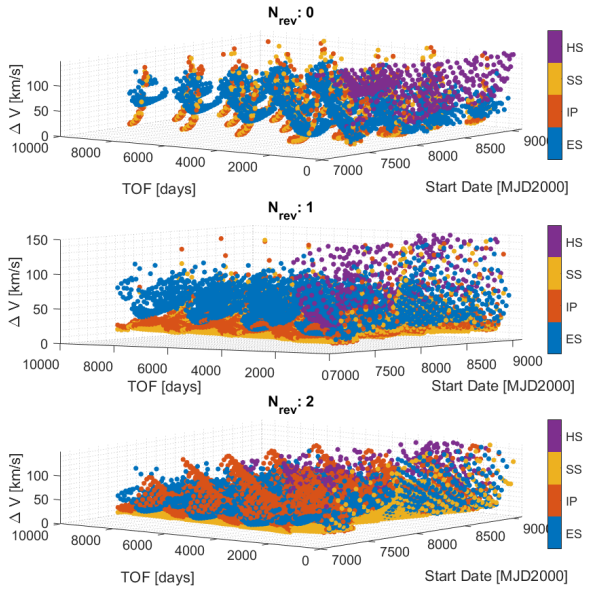



Figure 6.18: All $\Delta V$ results for given inputs of start date, time of flight and $N_{rev}$, with its method indicated.

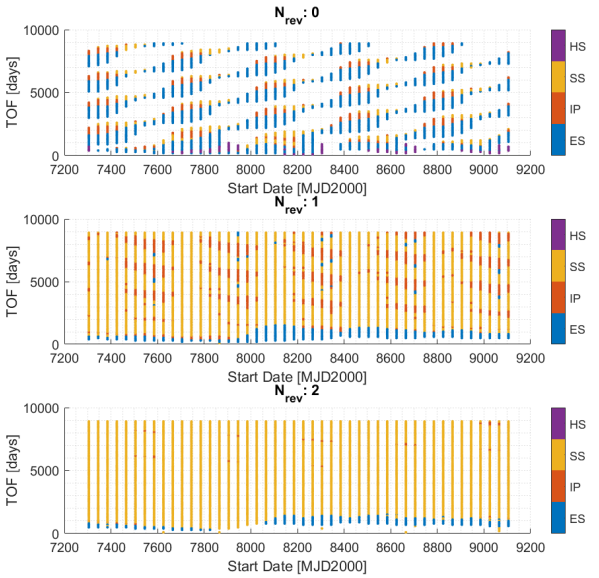



Figure 6.19: The lowest $\Delta V$ for given inputs of start date, time of flight and $N_{rev}$, with its method indicated.

For $N_{rev} = 0$ there are less data points shown, indicating some difficulty for all methods to provide solutions with feasible $\Delta V$s for all inputs, although this can be attributed to the aforementioned phenomena regarding optimum TOF. It is seen that the exponential sinusoids' results are quite spread, the inverse polynomials' results are grouped in patches giving some very low results. Usually at these patches the results for the spherical shaping method can be found with similarly low results. The hodographic shaping results are very high and are only low enough for lower TOFs.

For $N_{rev} = 1$ there are more shapes feasible, although a similar observation can be made as before. The lowest results are mostly from spherical shaping, and to a lesser extent inverse polynomials. Exponential sinusoids fill the gaps with higher $\Delta V$s and is found for low TOFs, the hodographic shaping is never low enough to appear in the plot.

For $N_{rev} = 2$ the results are dominated by spherical shaping. with some values for inverse polynomials and exponential sinusoids at the lower TOFs.

### GENERAL RESULTS FOR ALL NUMBER OF REVOLUTIONS

The results for all number of revolutions and methods are combined resulting in the plots of Figure 6.20. The plot for minimum $\Delta V$ mostly contains the results obtained with the spherical shaping method. The aforementioned patterns can be seen again, like the fact that the optimal $N_{rev}$ increases with increasing TOF, as more revolutions require more TOF.

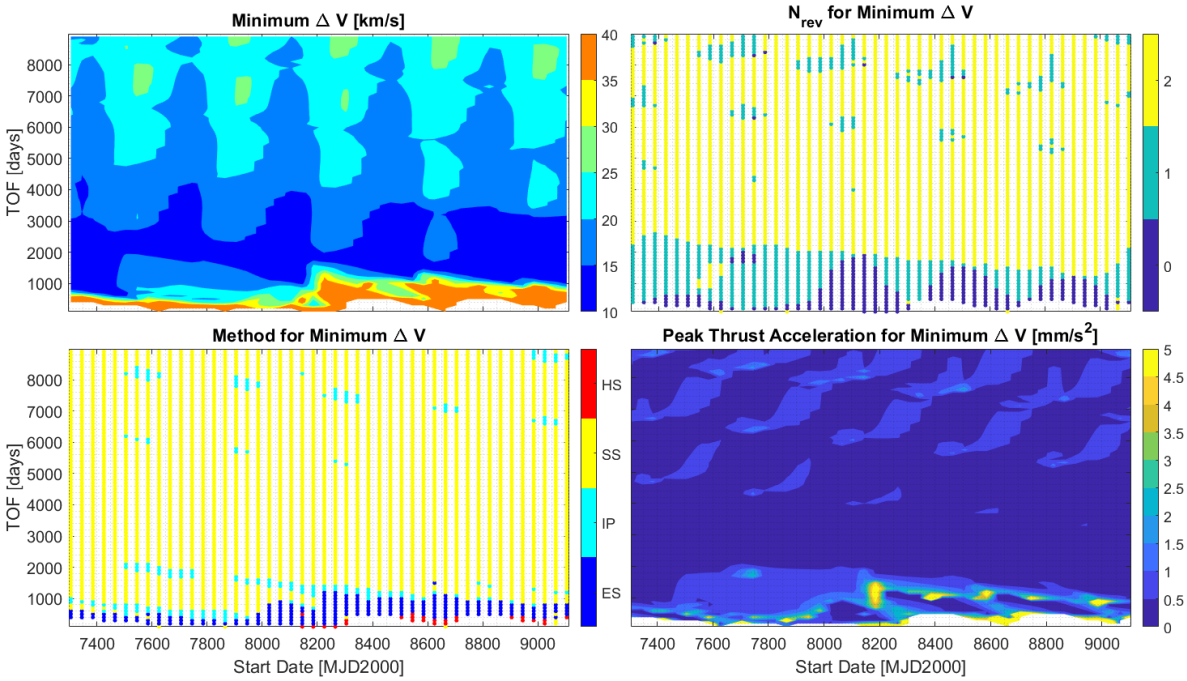



Figure 6.20: The lowest $\Delta V$ for given inputs of start date, time of flight with the associated $N_{rev}$, method and peak acceleration.

It is seen that the region of interest (low $\Delta V$s) is located between 500 and 3000 days for TOF and while the overall pattern is dominated by spherical shaping, a mix of the other methods is visible. The influence of the start date, thus relative planet positions, clearly has a large influence.

The accompanying thrust levels for the low $\Delta V$ region are between 0.2 and 2 mm/s$^2$. In the literature study it was found that the current low-thrust spacecraft deliver accelerations that vary from 0.037 to 0.37 mm/s$^2$, which means only a part of the low $\Delta V$ region has a low enough thrust to be considered feasible. Using the plot of the peak acceleration thrust, the dark blue regions (0-0.5 mm/s$^2$) are used to determine which inputs yield practical trajectories in combination with the low $\Delta V$ regions. The start dates from MJD2000 7300 to 9100 trajectories are promising ($\Delta V$ of 10-15 km/s), although there are more trajectories for start dates beginning at MJD2000 8300. The overall best trajectories (low $\Delta V$ and low TOF) seem to be located near start dates of MJD2000 7300 and 8800-9100 and a TOF between 1000 and 2000 days. These trajectories use spherical shaping for their trajectory shape.

The poor performance of exponential sinusoids could be explained by the fact that it has a free parameter. Exponential sinusoids use a winding parameter, which was unoptimised for all runs.

The poor results for the hodographic shaping method are likely due to the used default TUDAT base functions combination, which of course dominates its behaviour. When looking at literature [28, Fig. 9.3] better $\Delta V$ results between 13-40 km/s are found for similar inputs. This really highlights the importance of finding the proper base functions.

## 6.3. RESULTS GLOBAL OPTIMISATION

After a grid search has been performed a more specialised method of finding the absolute minimum in de search space can be done. The global minimum will be located using global optimisers which find the optimum per method and per given number of revolutions. This allows for the optimisation of the winding parameter as used by the exponential sinusoids as well.

The used inputs are the same as the ones used for the grid search. The start date range is kept the same, however the number of revolutions is now up to 3. Following this new number of revolutions and the knowledge that too high and low TOFs do not yield valuable results, the input TOF now ranges from 500 to 6000 days. An additional input is the winding parameter for the exposins which ranges from 0.001 to 0.999. By including this parameter, better results are expected for the exposins compared to the grid search, where it was kept constant.

For optimisation, various seeds and algorithms have been used. Five seeds have been used for the random number generator; for optimisation, the DE algorithm, the simple GA and the PSO have been used. Table 6.2 contains the $\Delta V$s and accompanying peak accelerations from the various optimisations.

First of all it is seen that the overall results for the same inputs, but different seeds and optimiser are in general agreement. There are both small and large variations between seeds, algorithms and shape-based methods. There are some larger variations present in the inputs for the exponential sinusoids and inverse polynomials; it was seen that their grid searches resulted in many valleys with low $\Delta V$s. Therefore it could be expected that due to different initial sampling and optimisation method, different valleys are found and as a consequence the optimiser gets trapped in a sub-optimal one.

In terms of the actual results, it is seen that the inverse polynomials and spherical shaping yield the best $\Delta V$ results for all number of revolutions. The exposins produce slightly worse results but still low, likely as now the winding parameter is optimised for as well. The hodographic shaping results have the highest required $\Delta V$s, as seen with the grid search likely attributable to the use of poor base functions for the velocity shape function.

Interestingly the required $\Delta V$ increases with increased number of revolutions, while it would have been expected that a longer travel time would result in lower propellant costs. However when looking at the peak accelerations it is seen that the reverse is true, which too can be attributed to shorter travel times requiring more thrust. The latter seems to be true for all used shape-based methods except for the exposins, which is logical considering its use of high-thrust burns to meet the initial and final velocity boundary conditions neglected here.

Comparing thrust acceleration of past missions (0.037 to 0.37 mm/s$^2$) with the ones found in the optimisation, It is seen that the trajectories with at least one or more full revolutions are feasible, however for spherical shaping and hodographic shaping they are relatively high (0.6-0.9 mm/s$^2$) for one full revolution, which could be considered infeasible.

The best solution (least amount of $\Delta V$) for each number of revolutions has been determined and the accompanying optimiser, shape-based method, start date, time of flight and peak acceleration is given in Table 6.3. It is seen that the DE algorithm seems to yield the best results, not surprisingly the best results come from the inverse polynomials and spherical shaping method. An outlier seems to be the best result for 3 revolutions coming from the exponential sinusoids; this could be explained by the neglect of high $\Delta V$ burns for matching boundary velocities.

The top view of the trajectories of the four best results is shown in Figure 6.21; it can be seen that all trajectories are similar to the ones seen before. Figure 6.22 shows the required thrust acceleration and resulting inclination for the optimal trajectories. For the inverse polynomial trajectories it is seen that usually there is a peak at the start for raising the orbit followed by an increase near the end to match the required inclination. For $N_{rev} = 1$, the second peak combines the radius increase and inclination change due to the position of Tempel-1. For $N_{rev} = 2$ with spherical shaping, it is seen that there are two peaks: each peak increases the orbit radius and they coincide with a closer proximity to the central body, thus more optimal for executing the manoeuvre. The inclination is increased gradually over time, with an increase near the end, and two increases with the peaks. As the exponential sinusoid method used extra high burns at the start and end this behaviour is not noticed.

Table 6.2: Minimum $\Delta V$ in km/s from a global optimisation of an Earth-Tempel-1 Rendez-Vous low-thrust transfer for various start dates, time of flight inputs per number of revolutions per shape-based method and seed for random generator. DE = Differential Evolution, SGA = Simple Genetic Algorithm, PSO = Particle Swarm Optimisation.

| | | | $\Delta V$ [km/s] | | | | | Peak Acceleration [mm/s$^2$] | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Algo.** | **Shape** | $N_{rev}$ | **123** | **456** | **789** | **111** | **222** | **123** | **456** | **789** | **111** | **222** |
| **DE** | **ES** | 0 | 15.925 | 14.794 | 15.050 | 17.384 | 14.990 | 0.172 | 0.155 | 0.159 | 0.253 | 0.166 |
| | | 1 | 15.463 | 16.352 | 15.936 | 14.284 | 14.203 | 0.183 | 0.159 | 0.170 | 0.181 | 0.161 |
| | | 2 | 13.096 | 13.099 | 13.570 | 13.963 | 13.530 | 0.069 | 0.070 | 0.062 | 0.067 | 0.078 |
| | | 3 | 13.247 | 13.138 | 13.180 | 13.092 | 13.114 | 0.049 | 0.057 | 0.061 | 0.058 | 0.057 |
| | **IP** | 0 | 10.508 | 10.509 | 10.508 | 10.509 | 10.512 | 2.692 | 2.680 | 2.690 | 2.684 | 2.661 |
| | | 1 | 10.872 | 10.872 | 10.871 | 10.871 | 10.872 | 0.318 | 0.340 | 0.337 | 0.332 | 0.346 |
| | | 2 | 13.383 | 13.267 | 13.260 | 13.253 | 13.279 | 0.323 | 0.316 | 0.316 | 0.314 | 0.314 |
| | | 3 | 17.805 | 17.816 | 17.326 | 17.738 | 17.039 | 0.242 | 0.247 | 0.435 | 0.253 | 0.450 |
| | **SS** | 0 | 11.714 | 11.714 | 11.714 | 11.714 | 11.714 | 1.634 | 1.634 | 1.635 | 1.629 | 1.630 |
| | | 1 | 12.258 | 12.274 | 12.309 | 12.288 | 12.293 | 0.826 | 0.826 | 0.841 | 0.878 | 0.884 |
| | | 2 | 13.041 | 13.039 | 13.023 | 13.039 | 13.021 | 0.285 | 0.285 | 0.319 | 0.280 | 0.315 |
| | | 3 | 13.199 | 13.199 | 13.199 | 13.199 | 13.199 | 0.193 | 0.193 | 0.192 | 0.193 | 0.192 |
| | **HS** | 0 | 40.128 | 40.128 | 40.128 | 40.128 | 40.128 | 1.325 | 1.325 | 1.325 | 1.325 | 1.326 |
| | | 1 | 26.226 | 26.226 | 26.226 | 26.226 | 26.226 | 0.672 | 0.672 | 0.672 | 0.672 | 0.672 |
| | | 2 | 30.001 | 30.001 | 30.001 | 30.001 | 30.001 | 0.330 | 0.330 | 0.330 | 0.330 | 0.330 |
| | | 3 | 29.463 | 29.804 | 29.427 | 29.492 | 29.460 | 0.222 | 0.218 | 0.221 | 0.219 | 0.221 |
| **SGA** | **ES** | 0 | 17.509 | 17.462 | 17.408 | 17.385 | 14.811 | 0.243 | 0.259 | 0.250 | 0.253 | 0.155 |
| | | 1 | 19.305 | 20.398 | 16.862 | 14.471 | 14.836 | 0.297 | 0.261 | 0.185 | 0.181 | 0.132 |
| | | 2 | 13.830 | 21.029 | 14.464 | 20.573 | 15.350 | 0.084 | 0.129 | 0.120 | 0.155 | 0.117 |
| | | 3 | 16.955 | 20.592 | 20.822 | 16.557 | 13.673 | 0.130 | 0.114 | 0.107 | 0.095 | 0.069 |
| | **IP** | 0 | 11.340 | 10.577 | 10.623 | 10.662 | 10.909 | 0.613 | 2.521 | 2.253 | 2.248 | 2.199 |
| | | 1 | 11.837 | 10.911 | 12.478 | 11.239 | 11.992 | 0.598 | 0.554 | 1.044 | 0.797 | 0.601 |
| | | 2 | 17.645 | 17.143 | 13.807 | 13.258 | 16.202 | 0.240 | 0.205 | 0.379 | 0.316 | 0.363 |
| | | 3 | 18.988 | 19.401 | 19.148 | 19.909 | 18.427 | 0.176 | 0.186 | 0.112 | 0.220 | 0.151 |
| | **SS** | 0 | 11.715 | 11.750 | 11.772 | 11.715 | 11.748 | 1.606 | 1.664 | 1.623 | 1.602 | 1.609 |
| | | 1 | 12.427 | 12.565 | 12.782 | 12.782 | 12.782 | 0.663 | 0.719 | 0.387 | 0.389 | 0.386 |
| | | 2 | 13.092 | 13.097 | 13.039 | 13.096 | 13.097 | 0.321 | 0.254 | 0.282 | 0.256 | 0.255 |
| | | 3 | 13.209 | 13.203 | 13.202 | 13.263 | 13.260 | 0.194 | 0.204 | 0.189 | 0.208 | 0.209 |
| | **HS** | 0 | 42.760 | 40.623 | 40.457 | 40.182 | 40.647 | 1.131 | 1.221 | 1.229 | 1.284 | 1.230 |
| | | 1 | 27.792 | 26.567 | 26.318 | 26.309 | 28.215 | 0.709 | 0.652 | 0.662 | 0.662 | 0.622 |
| | | 2 | 34.600 | 36.885 | 30.272 | 46.620 | 37.599 | 0.804 | 0.828 | 0.325 | 0.476 | 0.405 |
| | | 3 | 31.232 | 39.552 | 39.135 | 30.258 | 34.025 | 0.227 | 0.270 | 0.258 | 0.226 | 0.349 |
| **PSO** | **ES** | 0 | 16.297 | 15.382 | 16.975 | 17.010 | 15.570 | 0.161 | 0.176 | 0.155 | 0.141 | 0.166 |
| | | 1 | 14.758 | 14.614 | 13.860 | 14.274 | 14.047 | 0.161 | 0.162 | 0.166 | 0.171 | 0.167 |
| | | 2 | 14.580 | 14.418 | 16.491 | 13.388 | 13.530 | 0.097 | 0.124 | 0.161 | 0.068 | 0.062 |
| | | 3 | 17.845 | 14.167 | 14.044 | 15.029 | 13.715 | 0.058 | 0.046 | 0.055 | 0.043 | 0.073 |
| | **IP** | 0 | 10.836 | 10.822 | 10.785 | 10.709 | 10.617 | 1.549 | 2.390 | 2.271 | 1.960 | 2.270 |
| | | 1 | 10.872 | 10.872 | 10.872 | 10.876 | 10.878 | 0.319 | 0.319 | 0.320 | 0.329 | 0.322 |
| | | 2 | 14.022 | 13.570 | 14.798 | 14.487 | 13.530 | 0.326 | 0.314 | 0.343 | 0.328 | 0.318 |
| | | 3 | 18.432 | 18.492 | 18.518 | 18.256 | 18.253 | 0.147 | 0.275 | 0.175 | 0.155 | 0.468 |
| | **SS** | 0 | 11.732 | 11.749 | 11.748 | 11.746 | 11.742 | 1.477 | 1.467 | 1.649 | 1.405 | 1.634 |
| | | 1 | 12.427 | 12.446 | 12.370 | 12.353 | 12.397 | 0.663 | 0.799 | 0.872 | 0.777 | 0.672 |
| | | 2 | 13.044 | 13.043 | 13.040 | 13.062 | 13.055 | 0.288 | 0.286 | 0.277 | 0.286 | 0.281 |
| | | 3 | 13.207 | 13.201 | 13.230 | 13.204 | 13.202 | 0.187 | 0.198 | 0.199 | 0.191 | 0.202 |
| | **HS** | 0 | 40.369 | 40.196 | 40.134 | 40.241 | 40.136 | 1.400 | 1.285 | 1.311 | 1.400 | 1.344 |
| | | 1 | 26.242 | 26.472 | 26.446 | 26.258 | 26.562 | 0.676 | 0.666 | 0.670 | 0.668 | 0.658 |
| | | 2 | 30.306 | 31.115 | 30.576 | 30.099 | 30.597 | 0.333 | 0.312 | 0.337 | 0.325 | 0.320 |
| | | 3 | 29.609 | 29.523 | 30.795 | 32.081 | 34.215 | 0.222 | 0.219 | 0.216 | 0.227 | 0.350 |

Table 6.3: Minimum $\Delta V$ in km/s from a global optimisation of an Earth-Tempel-1 Rendez-Vous low-thrust transfer with accompanying start date, time of flight, peak acceleration, optimisation algorithm and shape-based method.

| Nrev [-] | Algo. | Shape | Seed | Start Date [MJD200] | TOF [days] | $\Delta V$ [km/s] | Peak Accel. [mm/s$^2$] |
|---|---|---|---|---|---|---|---|
| 0 | DE | IP | 123 | 8154.1 | 1300.4 | 10.508 | 2.692 |
| 1 | DE | IP | 789 | 7314.2 | 917.8 | 10.871 | 0.337 |
| 2 | DE | SS | 222 | 7332.4 | 2482.7 | 13.021 | 0.315 |
| 3 | DE | ES | 111 | 8526.0 | 1610.5 | 13.092 | 0.058 |

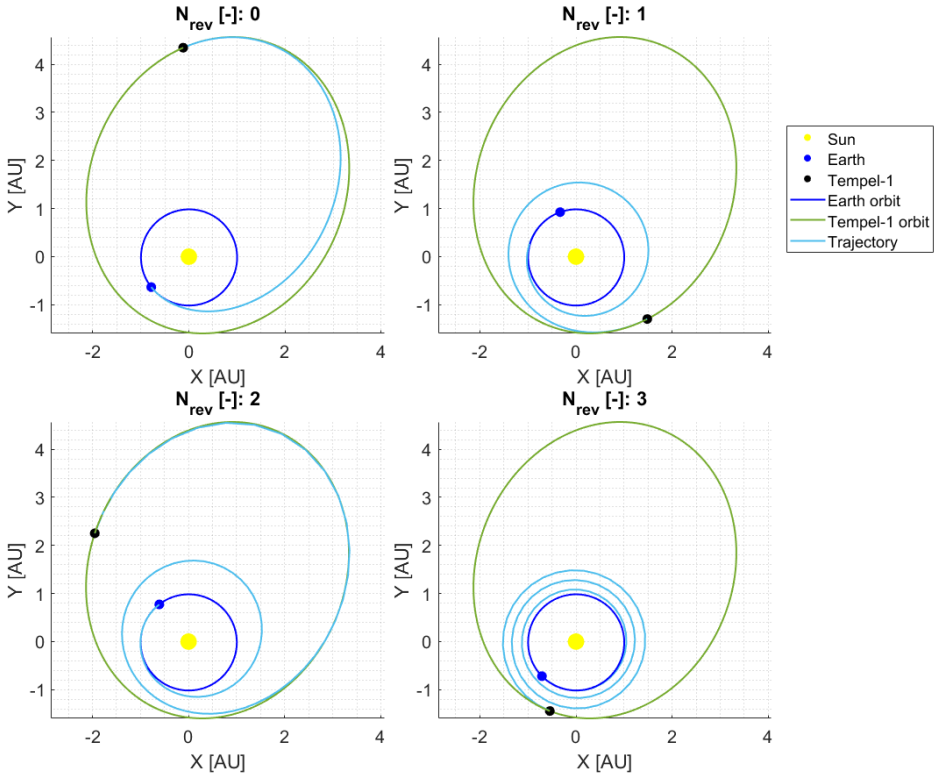



Figure 6.21: Trajectories of the optimum transfers found with global optimisation, with the initial conditions as described in Table 6.3.

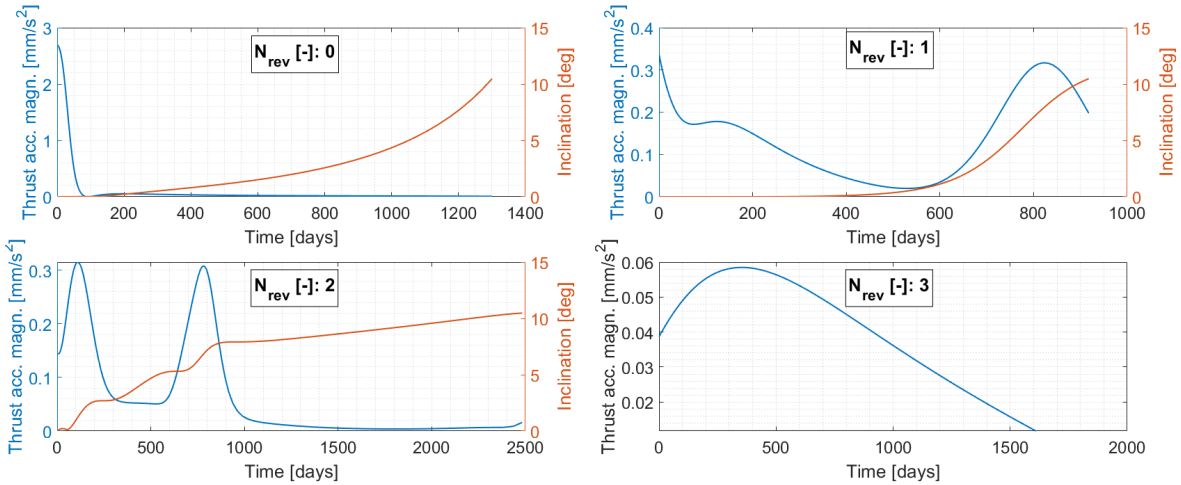



Figure 6.22: Thrust profiles and inclination over time of the optimum transfers found with global optimisation, with the initial conditions as described in Table 6.3.

# 7

# Interplanetary Transfer: Earth Tempel-1 With Coasting

In the previous chapter the Earth-Tempel-1 transfer consisted of a trajectory with a single arc with (expected) continuous thrusting. In this chapter a coasting phase will be added between two shape-based method arcs. In the first section the transfer case is defined, with its outline, inputs and outputs (Section 7.1). The next section discusses the optimisation strategy, with the establishment of the analysis groups, the order of the optimisation runs, the effects of the optimiser settings and finally the methodology for the actual optimisation (Section 7.2). The next section presents the results where the single-objective optimisation of spherical shaping with coasting is used (Section 7.3). Then the same type of analysis is repeated, but now with inverse polynomials (Section 7.4). After single-objective optimisation, the same problem is analysed using multi-objective optimisation (Section 7.5). The next analysis that is performed is the combination of IP and SS with coasting (Section 7.6). The last analysis group is the multi-objective optimisation again with IP, but now rebalanced fitness values (Section 7.7). The following section presents the comparison of all previous results with both each other and the previous chapter (Section 7.8). Finally general conclusions from all the presented results are made (Section 7.9).

## 7.1. Transfer Case Set-up

To analyse the performance of the shape-based methods by the inclusion of coasting, a transfer case needs to be defined with its powered and coasting phases, inputs and outputs. This is done in the aforementioned order in the following subsections.

### 7.1.1. Outline

Figure 7.1 shows the transfer case with its various phases. There are three main phases, each with its related variables; the actual independent variables are given in bold. Each phase consists of a departure point, arrival point and Time Of Flight (TOF) as is required for the shape-based methods. The number of full revolutions for each powered arc is set to zero, meaning the shape is allowed to cover less than a full revolution around the central body.

First there is the powered phase from the departure body (**start date**) to an intermediate state (**coasting state 1**) with a certain time of flight (**TOF1**). Secondly there is a coasting phase from the intermediate state with its time of flight (**TOF2/C**) and endpoint decided by the central body gravity (coasting state 2). Thirdly the last phase is a powered flight from coasting point 2 to the arrival body (TOF) with a given time of flight (**TOF3**). As seen there are now 10 independent variables, note that coasting state 1 includes the 6 elements for defining its state.

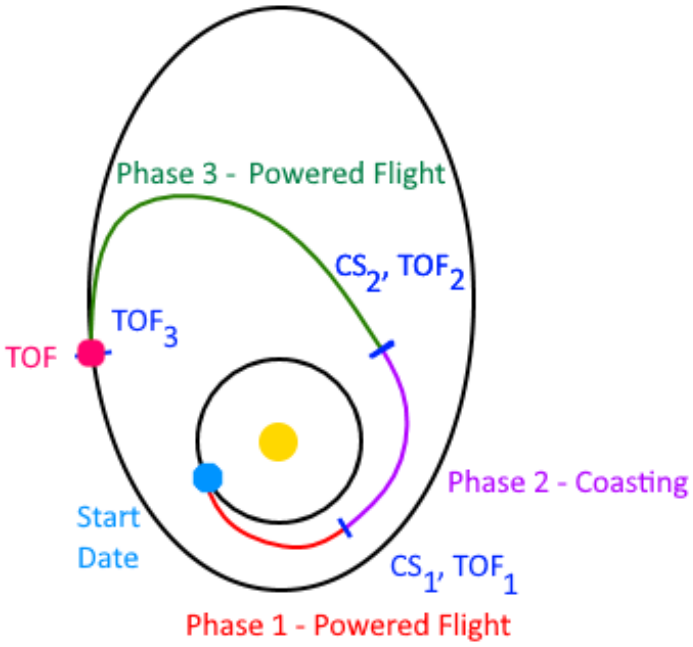



Figure 7.1: Earth-Tempel-1 transfer with 2 powered arcs and a coasting arc.

### 7.1.2. INPUTS AND OUTPUTS

Table 7.1 shows the inputs (independent variables) and the ranges that have been used for the first optimisation runs. In the next sections more information will be provided on the actual values used for the inputs.

The ranges for the start date are the same as those used for the study done in the previous chapter i.e. without coasting. The ranges for the TOF of the arcs have been chosen to be between 100 and 1000 days, as this leads to a total TOF of 300 to 3000 days. This range covers the same area where the best results with the non-coasting cases were found, as well as a reasonable time frame of 9 years at most. The remaining 6 elements form the first coasting state; its elements were chosen such that they allow a wide range of positions (1-5 AU from the Sun) and velocities. It is expected that after a preliminary optimisation, better and more appropriate bounds could be found.

The outputs are the same as those used in the non-coasting transfer (Chapter 6). They are the performance parameters $\Delta V$, peak thrust, the trajectories and their required thrust profiles.

Table 7.1: Input definition and ranges used for the optimisation of the Earth-Tempel-1 transfer including coasting.

| **Inputs** | **Bounds [lower, upper]** | **Units** |
|---|---|---|
| Start date | [7300, 9100 ] | MJD2000 |
| TOF1 | [200, 1000 ] | days |
| TOF2 | [200, 1000 ] | days |
| TOF3 | [200, 1000 ] | days |
| Radius | [0.9, 5 ] | AU |
| Azimuth | [0, 360 ] | deg |
| Elevation | [-28.65 28.65 ] | deg |
| $V_r$ | [-30.0, 30.0] | [km/s] |
| $V_\theta$ | [ 5.0, 40.0 ] | [km/s] |
| $V_\varphi$ | [-20.0, 20.0] | [km/s] |

## 7.2. OPTIMISATION STRATEGY

This section presents the definition of the analysis groups, the order of the optimisation runs and which inputs they use. Then the effects of the optimiser settings found with experimentation are shown and how they should be defined or tuned. Finally the methodology for the actual general and detailed optimisation is presented per analysis group.

### 7.2.1. OPTIMISER DEFINITIONS

In order to evaluate the performance of the shape-based methods combined with coasting, several groups of analysis have been performed. In this section these groups are elaborated, the specifications for the optimisations that have been performed are given as well as which inputs for the transfers themselves have been used.

This transfer has significantly more independent variables (ten) versus the three required for the non-coasting method. A grid search was considered, however with this many variables, the more difficult manner of visualisation and the relatively high CPU-time per run, it was deemed infeasible. Instead a global optimiser has been applied to the problem with large ranges for the variables, such that a more focused 'grid search' can be emulated. Areas of interest were found and were used in the next step. This more focused search for global optima is performed by applying the same global optimiser but to these new areas of interest, thus smaller ranges for the input variables.

Five groups of analyses have been performed: two with only $\Delta V$ to be optimised, two where both the peak thrust and $\Delta V$ are optimised and the fifth where both performance parameters were normalised for a balanced multi-objective optimisation. Thus both a single and a multi-objective optimisation problem is analysed, while for the former, peak thrust is not taken into the optimisation; it is used during the analysis of its results. The shape-based methods used in these problems are Inverse Polynomials (IP) and spherical shaping (SS), as from the previous chapter it was found they yielded the best results. The exact configuration of the optimised parameters and methods are summarised in Table 7.2, whereby the first column represents the analysis group, the second one the arcs of the transfer and the third one the optimised performance parameters. The run number column represents the associated optimisation runs. These run numbers will be used to identify which inputs have been used, the configuration for the optimiser and why this run was performed. It would seem that group E is the same as group C, however group E has a rebalanced cost function, to decrease the potential dominance of $\Delta V$.

Table 7.2: Main groups of optimisations, with the used shape-based method optimisation parameters and corresponding run numbers.

| Analysis Group | Method | Objective | Run Nr |
|---|---|---|---|
| A | SS-C-SS | $\Delta V$ | O1-O8, O20-O30 |
| B | IP-C-IP | $\Delta V$ | O8-O19, O31-O38 |
| C | IP-C-IP | $\Delta V$, peakAcc | MO1-MO14, MO21-23 |
| D | IP-C-SS | $\Delta V$, peakAcc | MO14-MO20 |
| E | IP-C-IP | $\Delta V$, peakAcc | MO24-MO30 |

Table 7.3 shows the settings and link to inputs for analysis group A. The first column provides information on which problem formulation is used; SS-C-SS means spherical shaping is used for both powered phases. The second column indicates the goal of this optimisation, with GO for general optimisation and DO detailed optimisation. The third column defines which set of input ranges have been used, for example IID1 (Input Identification 1) refers to the corresponding table of inputs in Section C.1, Appendix C. The column "Name", means the name of the optimisation run, where O stands for single-objective optimisation. Each run consists of optimising a population (nPop) and evolving them for a certain number of generations (nGen) using a certain optimisation algorithm (algo). The column "Seed", refers to the seed number in TUDAT that has been used to generate quasi-random numbers. The column "Ntraj", is simply how many trajectories have been evaluated and runtime is how long it took. This table together with the input ranges tables, allows for the reproduction of the optimisation, it will also be explained why these specific inputs were used.

Table 7.3: All optimisation runs, with their settings for analysis group A. GO = General Optimisation, DO = Detailed Optimisation.

| Method | Goal | Input ID | Name | nPop | nGen | algo | seed | Ntraj | runtime [h:mm:ss] |
|---|---|---|---|---|---|---|---|---|---|
| SS-C-SS | nPop/nGen | 1 | runO1 | 500 | 200 | de | 123 | 100000 | - |
| SS-C-SS | nPop/nGen | 1 | runO2 | 100 | 100 | de | 123 | 10000 | - |
| SS-C-SS | algo | 1 | runO3 | 100 | 100 | pso | 123 | 10000 | - |
| SS-C-SS | nPop/nGen | 1 | runO4 | 500 | 300 | de | 123 | 150000 | - |
| SS-C-SS | GO | 2 | runO5 | 100 | 100 | de | 123 | 10000 | - |
| SS-C-SS | nPop/nGen | 2 | runO6 | 100 | 1000 | de | 123 | 100000 | 0:45:00 |
| SS-C-SS | runtime | 2 | runO7 | 100 | 500 | de | 123 | 50000 | 0:17:53 |
| SS-C-SS | GO | 12 | runO20 | 100 | 1000 | de | 123 | 100000 | 0:33:57 |
| SS-C-SS | DO | 13 | runO21 | 100 | 750 | de | 123 | 75000 | 0:31:12 |
| SS-C-SS | DO | 14 | runO22 | 100 | 750 | de | 123 | 75000 | 0:30:30 |
| SS-C-SS | DO | 15 | runO23 | 100 | 750 | de | 123 | 75000 | 0:39:51 |
| SS-C-SS | DO | 16 | runO24 | 100 | 750 | de | 123 | 75000 | 0:35:47 |
| SS-C-SS | DO | 17 | runO25 | 100 | 750 | de | 123 | 75000 | 0:36:52 |
| SS-C-SS | DO | 18 | runO26 | 100 | 750 | de | 123 | 75000 | 0:48:43 |
| SS-C-SS | DO | 19 | runO27 | 100 | 750 | de | 123 | 75000 | 0:46:54 |
| SS-C-SS | DO | 20 | runO28 | 100 | 750 | de | 123 | 75000 | 0:46:29 |
| SS-C-SS | DO | 21 | runO29 | 100 | 750 | de | 123 | 75000 | 0:43:48 |
| SS-C-SS | DO | 22 | runO30 | 100 | 750 | de | 123 | 75000 | 0:39:32 |

RunO1 to runO6 were set up to experiment with the optimiser settings. They were used to determine the boundaries of the input values as well as the best nPop and nGen settings and their combination and their influence on runtime and interaction with seed number. RunO7 was used to verify the implementation of a code optimisation for runtime. If the requirements of the first arc are impossible to be met by the shape-based method, the subsequent arcs are skipped and marked for failure, thus reducing overall runtime. RunO20 was the final optimisation for a general overview of the full design space and subsequent runs of runO21 to runO30 were to narrow down on several local optima. The conclusions for the optimiser settings are discussed in Subsection 7.2.2 and the results for the actual optimisation in Section 7.3.

Table 7.4 is the optimiser input table for group B, which uses inverse polynomials as the shape-based method, with all columns as explained before. RunO8 to runO12 are to explore the design space for the input ranges, with runO11 to look at the effect of a different algorithm. The run numbers are chronologically, it is thus seen that the exploration runs of group A are followed by those of group B. This was done for computation time requirements and for a more robust analysis of the optimisers. It is seen that a larger Npop has been used as computation time was sufficiently low enough for IP. RunO13 is the final general analysis of the design space, after which the more detailed optimisation starts with runs O14-O19 and O33-36. With observations made from the group A analyses it was found that group B still had the first half of the start date design space which was unexplored in detail, thus resulting in a later analysis. It is seen that the overall runtimes are much lower compared to those of group A, which is one the main reasons IP is used again in groups C and E. Conclusions on the optimiser inputs can be found in Subsection 7.2.2 and the results from the actual optimisations in Section 7.4.

Table 7.5 is the optimiser input table for group C, which uses inverse polynomials as the shape-based method and multi-objective optimisation for the $\Delta V$ and peak acceleration. These runs were made after group A and B were completed and used the lessons learned, to be discussed in Subsection 7.2.2. With runMO1 it was already seen that multi-objective optimisers were much faster in converging. Runs MO1-MO3 were used to study the effects of seed number, which were larger than seen with single-objective optimisation. Runs MO5 to MO8 were done to experiment again with the population size and number of generations, as multi-objective optimisation has different properties. All these runs were not only used to explore the optimiser settings, but to explore the design space as well, with runMO9 being the final general overview run. run MO10 to MO14 were the more detailed analyses. Runs MO21-MO23 were performed to test different seeds in combination with the more detailed optimisation. Conclusions on the optimiser inputs can be found in Subsection 7.2.2 and the results from the actual optimisations in Section 7.5.

Table 7.4: All optimisation runs, with their settings for analysis group B. GO = General Optimisation, DO = Detailed Optimisation.

| Method | Goal | Input ID | Name | nPop | nGen | algo | seed | Ntraj | runtime [h:mm:ss] |
|---|---|---|---|---|---|---|---|---|---|
| IP-C-IP | GO | 2 | runO8 | 100 | 1000 | de | 123 | 100000 | 0:06:03 |
| IP-C-IP | GO | 1 | runO9 | 100 | 1000 | de | 123 | 100000 | 0:04:39 |
| IP-C-IP | GO | 3 | runO10 | 100 | 1000 | de | 123 | 100000 | 0:05:27 |
| IP-C-IP | algo | 3 | runO11 | 100 | 1000 | pso | 123 | 100000 | 0:04:39 |
| IP-C-IP | GO | 4 | runO12 | 100 | 1000 | de | 123 | 100000 | 0:06:28 |
| IP-C-IP | GO | 5 | runO13 | 100 | 1000 | de | 123 | 100000 | 0:06:11 |
| IP-C-IP | DO | 6 | runO14 | 100 | 1000 | de | 123 | 100000 | 0:09:09 |
| IP-C-IP | DO | 7 | runO15 | 100 | 1000 | de | 123 | 100000 | 0:07:46 |
| IP-C-IP | DO | 8 | runO16 | 100 | 1000 | de | 123 | 100000 | 0:06:14 |
| IP-C-IP | DO | 9 | runO17 | 100 | 1000 | de | 123 | 100000 | 0:08:58 |
| IP-C-IP | DO | 10 | runO18 | 100 | 1000 | de | 123 | 100000 | 0:08:13 |
| IP-C-IP | DO | 11 | runO19 | 100 | 1000 | de | 123 | 100000 | 0:03:23 |
| IP-C-IP | DO | 23 | runO31 | 100 | 1000 | de | 789 | 100000 | 0:06:29 |
| IP-C-IP | DO | 24 | runO32 | 100 | 1000 | de | 789 | 100000 | 0:09:15 |
| IP-C-IP | DO | 25 | runO33 | 100 | 1000 | de | 789 | 100000 | 0:09:20 |
| IP-C-IP | DO | 26 | runO34 | 100 | 1000 | de | 789 | 100000 | 0:15:56 |
| IP-C-IP | DO | 27 | runO35 | 100 | 1000 | de | 789 | 100000 | 0:06:00 |
| IP-C-IP | DO | 28 | runO36 | 100 | 1000 | de | 789 | 100000 | 0:07:18 |
| IP-C-IP | DO | 29 | runO37 | 100 | 1000 | de | 789 | 100000 | 0:09:35 |
| IP-C-IP | DO | 30 | runO38 | 100 | 1000 | de | 789 | 100000 | 0:15:09 |

Table 7.5: All optimisation runs, with their settings for analysis group C. GO = General Optimisation, DO = Detailed Optimisation.

| Method | Goal | Input ID | Name | nPop | nGen | algo | seed | Ntraj | runtime [h:mm:ss] |
|---|---|---|---|---|---|---|---|---|---|
| IP-C-IP | seed | 1 | runMO1 | 100 | 1000 | msoed | 123 | 100000 | 0:04:01 |
| IP-C-IP | seed | 1 | runMO2 | 100 | 1000 | msoed | 456 | 100000 | 0:07:44 |
| IP-C-IP | seed | 1 | runMO3 | 100 | 1000 | msoed | 789 | 100000 | 0:06:18 |
| IP-C-IP | nPop/nGen | 1 | runMO4 | 1000 | 100 | msoed | 123 | 100000 | 0:04:28 |
| IP-C-IP | nPop/nGen | 1 | runMO5 | 500 | 200 | msoed | 123 | 100000 | 0:05:53 |
| IP-C-IP | nPop/nGen | 1 | runMO6 | 300 | 300 | msoed | 123 | 90000 | 0:06:00 |
| IP-C-IP | nPop/nGen | 1 | runMO7 | 100 | 1000 | msoed | 123 | 100000 | 0:12:38 |
| IP-C-IP | nPop/nGen | 1 | runMO8 | 100 | 1000 | msoed | 123 | 100000 | 0:14:19 |
| IP-C-IP | GO | 1 | runMO9 | 1000 | 200 | msoed | 123 | 200000 | 0:24:32 |
| IP-C-IP | DO | 31 | runMO10 | 1000 | 200 | msoed | 123 | 200000 | 0:12:29 |
| IP-C-IP | DO | 32 | runMO11 | 1000 | 200 | msoed | 123 | 200000 | 0:25:02 |
| IP-C-IP | DO | 33 | runMO12 | 1000 | 200 | msoed | 123 | 200000 | 0:15:06 |
| IP-C-IP | DO | 34 | runMO13 | 1000 | 200 | msoed | 123 | 200000 | 0:16:06 |
| IP-C-IP | DO | 35 | runMO14 | 1000 | 200 | msoed | 123 | 200000 | 0:12:31 |
| IP-C-IP | seed | 1 | runMO21 | 1000 | 100 | msoed | 456 | 100000 | - |
| IP-C-IP | seed | 1 | runMO22 | 1000 | 100 | msoed | 789 | 100000 | - |
| IP-C-IP | seed | 1 | runMO23 | 1000 | 100 | msoed | 456 | 100000 | - |

Table 7.6 is the multi-objective optimiser input table for group D, which uses inverse polynomials for the first powered phase and spherical shaping for the last phase. Due to the code-optimisation (the second and third arcs are skipped in case of failure of the first arc), it is more advantageous to use the method with the smallest computation time first, which is IP. With the effects of optimiser inputs fully explored, run MO15 was used for an overview of the design space and runs MO16-MO20 for detailed analysis. Results from the actual optimisations can be found in Section 7.6.

Table 7.7 is the optimiser input table for group E, which uses inverse polynomials as the shape-based method and multi-objective optimisation for the $\Delta V$ and peak acceleration. The differences with group C and D are that the optimiser fitness values have been rebalanced: the $\Delta V$ is computed in km/s (from m/s) and peak acceleration in mm/s$^2$ (from m/s$^2$). This acts as a normalisation of the parameters as the optimiser assumes both are equally important; before the value for $\Delta V$ was in the thousands and for the peak acceleration below one, the former thus dominating the optimisation. Runs MO21 and MO22 were performed to test the change and to see whether the number of generations needed to be adjusted, which it turned out could be kept at 200. MO23-MO27 are the runs for the detailed analysis and use the same input ranges as for the other MO runs, however the coasting TOF lower bound was changed to 10 days. The results for the actual optimisations are shown in Section 7.7.

Table 7.6: All optimisation runs, with their settings for analysis group D. GO = General Optimisation, DO = Detailed Optimisation.

| Method | Goal | Input ID | Name | nPop | nGen | algo | seed | Ntraj | runtime [h:mm:ss] |
|---|---|---|---|---|---|---|---|---|---|
| IP-C-SS | GO | 31 | runMO15 | 1000 | 100 | msoed | 123 | 100000 | 0:21:15 |
| IP-C-SS | DO | 31 | runMO16 | 1000 | 100 | msoed | 123 | 100000 | 0:27:29 |
| IP-C-SS | DO | 32 | runMO17 | 1000 | 100 | msoed | 123 | 100000 | 0:31:15 |
| IP-C-SS | DO | 33 | runMO18 | 1000 | 100 | msoed | 123 | 100000 | 0:34:00 |
| IP-C-SS | DO | 34 | runMO19 | 1000 | 100 | msoed | 123 | 100000 | 0:26:01 |
| IP-C-SS | DO | 35 | runMO20 | 1000 | 100 | msoed | 123 | 100000 | 0:27:24 |

Table 7.7: All optimisation runs, with their settings for analysis group E. GO = General Optimisation, DO = Detailed Optimisation.

| Method | Goal | Input ID | Name | nPop | nGen | algo | seed | Ntraj | runtime [h:mm:ss] |
|---|---|---|---|---|---|---|---|---|---|
| IP-C-IP | GO | 37 | runMO24 | 1000 | 100 | msoed | 123 | 100000 | 0:04:20 |
| IP-C-IP | GO | 37 | runMO25 | 1000 | 300 | msoed | 123 | 300000 | 0:11:09 |
| IP-C-IP | DO | 32 | runMO26 | 1000 | 200 | msoed | 123 | 200000 | 0:07:09 |
| IP-C-IP | DO | 33 | runMO27 | 1000 | 200 | msoed | 123 | 200000 | 0:17:28 |
| IP-C-IP | DO | 34 | runMO28 | 1000 | 200 | msoed | 123 | 200000 | 0:14:01 |
| IP-C-IP | DO | 35 | runMO29 | 1000 | 200 | msoed | 123 | 200000 | 0:07:16 |
| IP-C-IP | DO | 36 | runMO30 | 1000 | 200 | msoed | 123 | 200000 | 0:07:44 |

### 7.2.2. EFFECT OF PROBLEM OPTIMISER SETTINGS

During the simulation and optimisation of the trajectories, the effect of several of the optimisation parameters could be established. These are summarised here and are found to be valid for the various analyses groups. The found optimal numbers are given at the end of this subsection.

The effect of 'Npop' is that smaller population sizes mean a smaller variety in different combination of the input variables. Thus with smaller 'Npop', there is more chance that areas of interest are missed. The effect of 'Ngen' is that with increasing number of generations, the optimiser is better allowed to converge on an optimum. More generations mean that the optimiser narrows its search space and has more time to find an optimum.

The effect of seed number was investigated as well as with the previous chapter, where it was found it could have an effect on the found optima. With smaller populations, the optimiser relies on chance that there are points in the actual areas of interest. Thus with a sufficient large enough population the influence of seed number (or quasi-random numbers) can be reduced if not avoided. This mostly matters for the early optimisation runs, where the design space is largest. For the detailed optimisation, this matters less. The latter was confirmed with runs O21-O23 where various seeds were used for performing the same optimisation as runs MO10-MO12 respectively.

In terms of runtime it was found that spherical shaping computations take considerably longer than inverse polynomial ones (up to 5-10 times). Therefore finding a good nPop and nGen was not only about a good spread and convergence, but also keeping within reasonable runtimes.

The effect of optimisers was already looked at in the previous chapter where DE, PSO and SGA were used. It was found that DE had better overall and more consistent results, and as such it is used for these optimisations. In the pursuit of curiosity, PSO was used again to determine its performance in a more complex problem. It was found that PSO produces worse results and has possibly inadequate implementation of box constraints on the inputs. MSOED is the multi-objective optimiser and includes the use of DE, and is recommended to use in TUDAT.

For groups A and B it was found that a population of 100 members and 1000 generations worked well because there were many manual changes to the design space between the optimisation runs, however for group A, it was determined that due to the long computation times 750 generations was still sufficient and not too small. For groups C, D and E it was found that the reverse was best: a population size of 1000 and at least 100 generations. Multi-objective optimisation has as trait that it proved to converge very quickly, but without a sufficient spread in population number not necessarily to the best one, therefore the switch in numbers. For groups C and E the number of generations was increased to 200 as the lower runtime allowed for it.

### 7.2.3. GENERAL AND DETAILED OPTIMISATION METHODOLOGY

With the optimiser inputs and optimisation runs explained, a final yet crucial aspect is the inputs for the optimisation problem and the methodology of performing the general and detailed optimisation and finding the optimal trajectories. For all groups a similar strategy was used; these are shown in terms of single or multi-objective optimisation and are elaborated by example. The full results for both the general and the detailed optimisations are not presented here, but will be in Sections 7.3 to 7.9.

#### OPTIMISATION STRATEGY FOR GROUPS A AND B

With general optimisation a large range for the inputs was used, as shown in Table 7.1. An optimisation run was then performed after which the results were investigated by plotting the $\Delta V$ versus their inputs. The input values of the trajectories with lower $\Delta V$ combined with the ones of the last couple of generations are inspected. If these results converge, a smaller range for the input values can be established, which will be used for further optimisation runs.

Starting with runO4, Figure 7.2 shows the computed $\Delta V$s for given start date, with the generation indicated in colour. It is seen that there are five regions of results with lower $\Delta V$s in combination with being part of the last few generations (yellow colour). As these seem to converge, five areas of interest (low $\Delta V$) are identified with red lines with which new input ranges can be established for a more detailed optimisation.

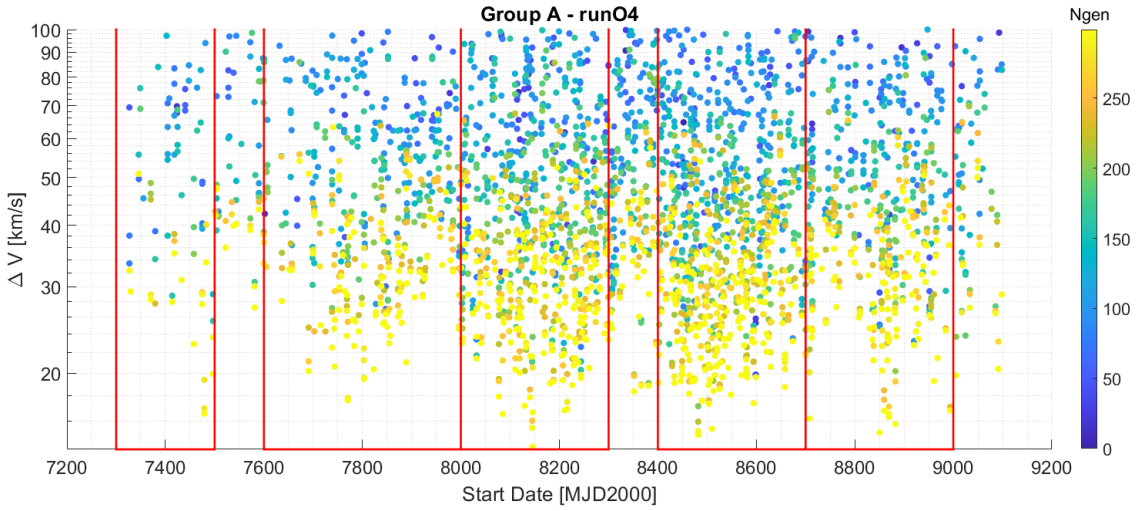



Figure 7.2: $\Delta V$s computed for given start date for the Earth-Tempel-1 transfer with generations indicated with colour for the given optimisation run. Areas of interest with respect to start date indicated with red.

Figure 7.3 shows the computed $\Delta V$s for two spherical elements $V_\varphi$ (out-of-plane velocity) and elevation (out-of-plane position) of the start of the coasting phase, with the plot not showing any result above 100 km/s for $\Delta V$ as the lower regions are of interest. Observing the colours of the results (yellow), the later the number of generations the more the results converge to a lower $\Delta V$ within a smaller range for the inputs. The input values of these results are between -7.5 km/s and 5.0 km/s for $V_\varphi$ and between -11.5 and 17.2 deg for elevation. These will become the bounds of the new inputs ranges as seen in Table 7.8 for Input 12, for the other inputs a similar process is performed. Note that the demarcation in terms of start date will not yet be applied. With these new ranges for the inputs the optimiser is executed for runO20. After viewing its results versus their inputs and their ranges in figures as seen before, new limits on the inputs are established for runO21 as defined with Input 13 in Table 7.8. Note that this time the limits for the start date are finally applied as well. The process repeats by inspecting its results versus the inputs and establishing the final input ranges for runO26 as seen in Input 18 in Table 7.8. From this the final optima can be found. These steps are performed for all five areas of interest (Figure 7.2) with respect to the start date, while here only one such area is given as example.

It can be seen that for each subsequent set, the ranges for the inputs narrow, such that the optimum can be better found. In Appendix C the remainder of the input tables can be found; they are to be linked to the optimiser runs using Table 7.3 and similarly for IP with Table 7.4.

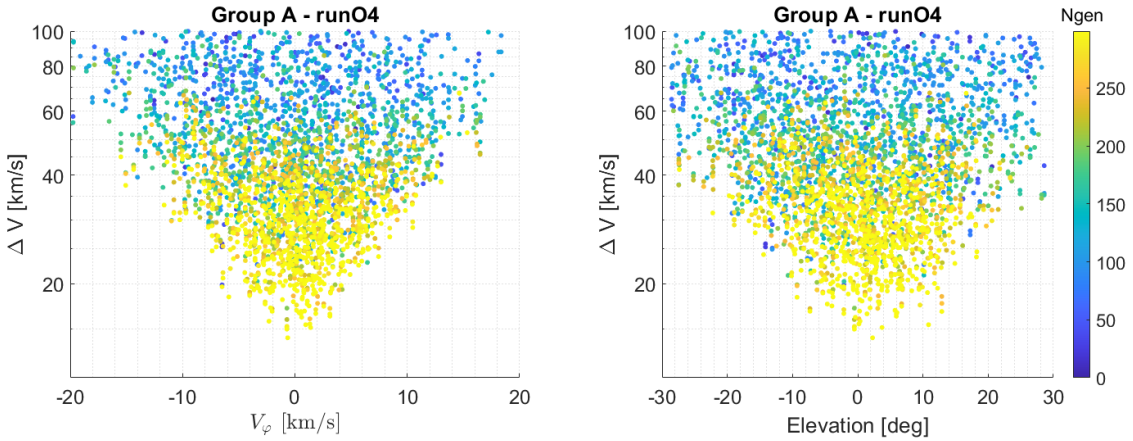



Figure 7.3: $\Delta V$s computed for given $V_\varphi$ (Left). $\Delta V$s computed for given elevation (Right). Results for the Earth-Tempel-1 transfer with generations indicated with colour for the given optimisation run.

Table 7.8: Input definition and ranges used for the optimisation of the Earth-Tempel-1 transfer including coasting, tables compiled from Appendix C. LB = Lower Bound, UB = Upper Bound.

| **IID** | **Input 1** | | **Input 12** | | **Input 13** | | **Input 18** | | |
|---|---|---|---|---|---|---|---|---|---|
| **Inputs** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **Units** |
| Start date | 7300 | 9100 | 7300 | 9100 | 7300 | 7500 | 7300 | 7420 | MJD2000 |
| TOF1 | 200 | 1000 | 200 | 1000 | 300 | 1000 | 300 | 650 | days |
| TOF2/C | 200 | 1000 | 100 | 1000 | 100 | 1000 | 25 | 300 | days |
| TOF3 | 200 | 1000 | 200 | 1000 | 400 | 1000 | 700 | 1100 | days |
| Radius | 0.9 | 5 | 1 | 5 | 1 | 2.5 | 1.2 | 1.8 | AU |
| Azimuth | 0 | 360 | 0 | 360 | 0 | 360 | 25 | 175 | deg |
| Elevation | -28.65 | 28.65 | -11.46 | 17.19 | -11.46 | 17.19 | -2.86 | 6.88 | deg |
| $V_r$ | -30.00 | 30.00 | -15.00 | 5.00 | -15.00 | 5.00 | -7.50 | 0.00 | km/s |
| $V_\theta$ | 5.00 | 40.00 | 17.50 | 27.50 | 17.50 | 27.50 | 19.00 | 27.50 | km/s |
| $V_\phi$ | -20.00 | 20.00 | -7.50 | 5.00 | -7.50 | 5.00 | -0.20 | 2.20 | km/s |

## OPTIMISATION STRATEGY FOR GROUPS C, D AND E

For the multi-objective optimisations, the process is simpler: it was found during the experimentation with the optimiser set-up that the multi-objective form converges much faster. Therefore only one general run needs to be performed with very wide ranges for the inputs as defined in Input 31 and 37 in Table 7.10. After this, only the start date has been split into five ranges in which the optimiser is executed. These start date ranges are equal for groups C, D and E and are defined in Table 7.9. The other input ranges are still as defined with Input 31 for groups C and D, however for group E it was found that the TOF2 was optimising in the direction of the lower bound, and as such the lower bound was reduced to 10 days as seen with Input 37.

Table 7.9: Input definition and ranges used for the optimisation of the Earth-Tempel-1 transfer including coasting, Inputs31-37. LB = Lower Bound, UB = Upper Bound.

| | | **Start Date [MJD2000]** | |
|---|---|---|---|
| **IID** | **Run Number** | **LB** | **UB** |
| 31 | runMO1-6 | 7300 | 9100 |
| 32 | runMO7,10,16,21-23,26 | 7300 | 7600 |
| 33 | runMO8,11,17,27 | 7600 | 8000 |
| 34 | runMO9,12,18,28 | 8000 | 8400 |
| 35 | runMO13,19,29 | 8400 | 8800 |
| 36 | runMO14,20,30 | 8800 | 9100 |
| 37 | runMO24-25 | 7300 | 9100 |

Table 7.10: Input definition and ranges used for the optimisation of the Earth-Tempel-1 transfer including coasting, Inputs 31,37. LB = Lower Bound, UB = Upper Bound.

| **IID** | **Input 31** | | **Input 37** | | |
|---|---|---|---|---|---|
| **Inputs** | **LB** | **UB** | **LB** | **UB** | **Units** |
| Start date | 7300 | 9100 | 7300 | 9100 | MJD2000 |
| TOF1 | 100 | 1000 | 100 | 1000 | days |
| TOF2 | 100 | 1000 | 10 | 1000 | days |
| TOF3 | 100 | 1000 | 100 | 1000 | days |
| Radius | 0.9 | 5 | 0.9 | 5 | AU |
| Azimuth | 0 | 360 | 0 | 360 | deg |
| Elevation | -28.65 | 28.65 | -28.65 | 28.65 | deg |
| $V_r$ | -30.00 | 30.00 | -30.00 | 30.00 | km/s |
| $V_\theta$ | 5.00 | 40.00 | 5.00 | 40.00 | km/s |
| $V_\varphi$ | -20.00 | 20.00 | -20.00 | 20.00 | km/s |

## 7.3. RESULTS SS-C-SS-SOO TRANSFER

This section presents the results for analysis group A, with the transfer consisting of two spherical shaping arcs and the analysis uses a single-objective optimiser.

RunO4 and runO20 are representative for the general optimisation. RunO4's start date vs $\Delta V$ figure has already been presented in Figure 7.2, where five areas of interest were identified. RunO20's results are shown in Figure 7.4. It is seen the optimiser converged on several local optima, where after a start date of 8000 there are three clear clusters of results. However before 8000 MJD2000 it was already seen there are potential results, and it is expected that the optimiser got stuck on the other local minima and discarded the early start dates.

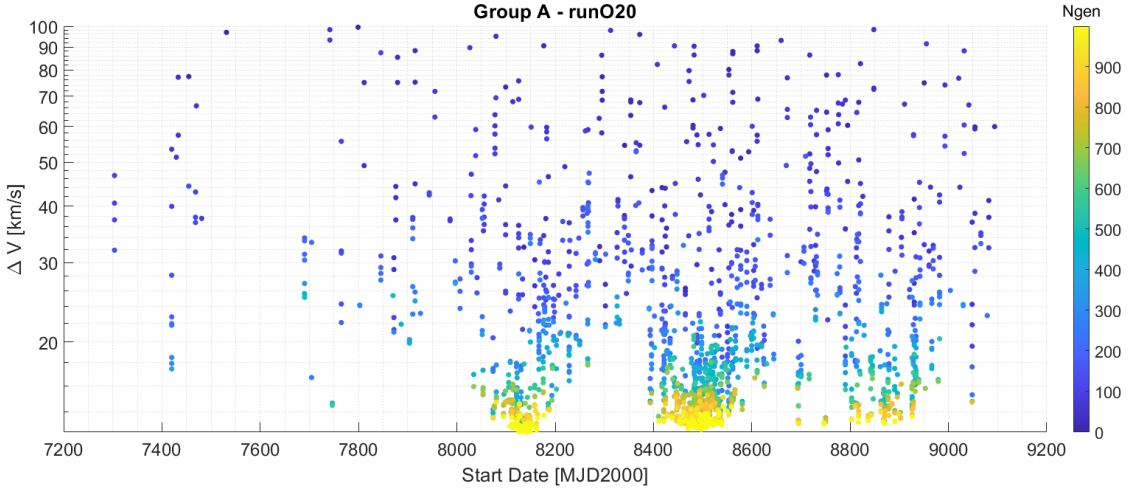



Figure 7.4: General Optimisation - $\Delta V$ plot for the Earth-Tempel-1 trajectories computed with spherical shaping over the indicated generations (Ngen), for the given optimisation run.

The results of the detailed optimisation of runO26 to O30 are compiled into one plot and shown in Figure 7.5. The plot is zoomed in onto the lower region for $\Delta V$ to view the differences more clearly in results found and the effect of later generations.

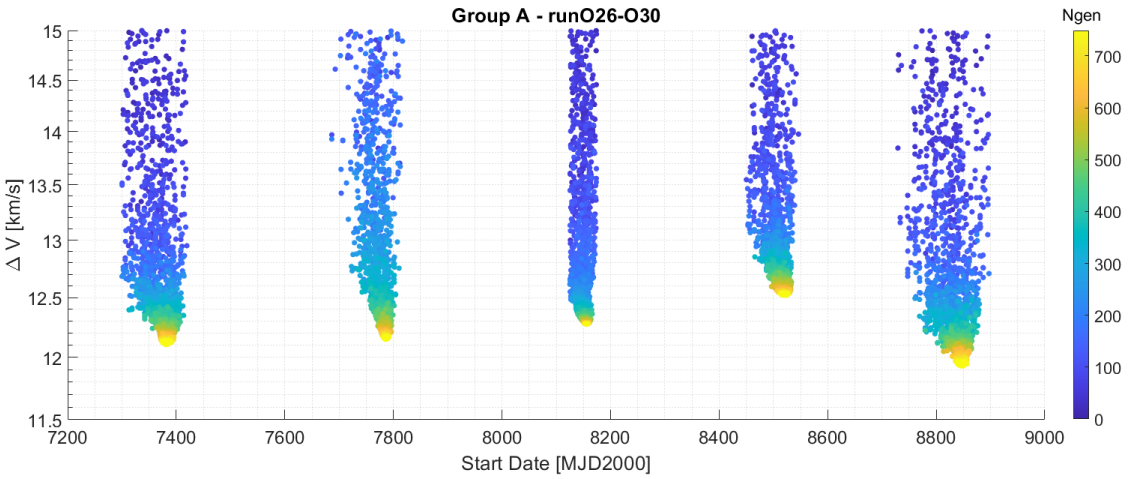



Figure 7.5: Detailed Optimisation - $\Delta V$ plot for the Earth-Tempel-1 trajectories computed with spherical shaping over the indicated generations (Ngen), for the given optimisation run.

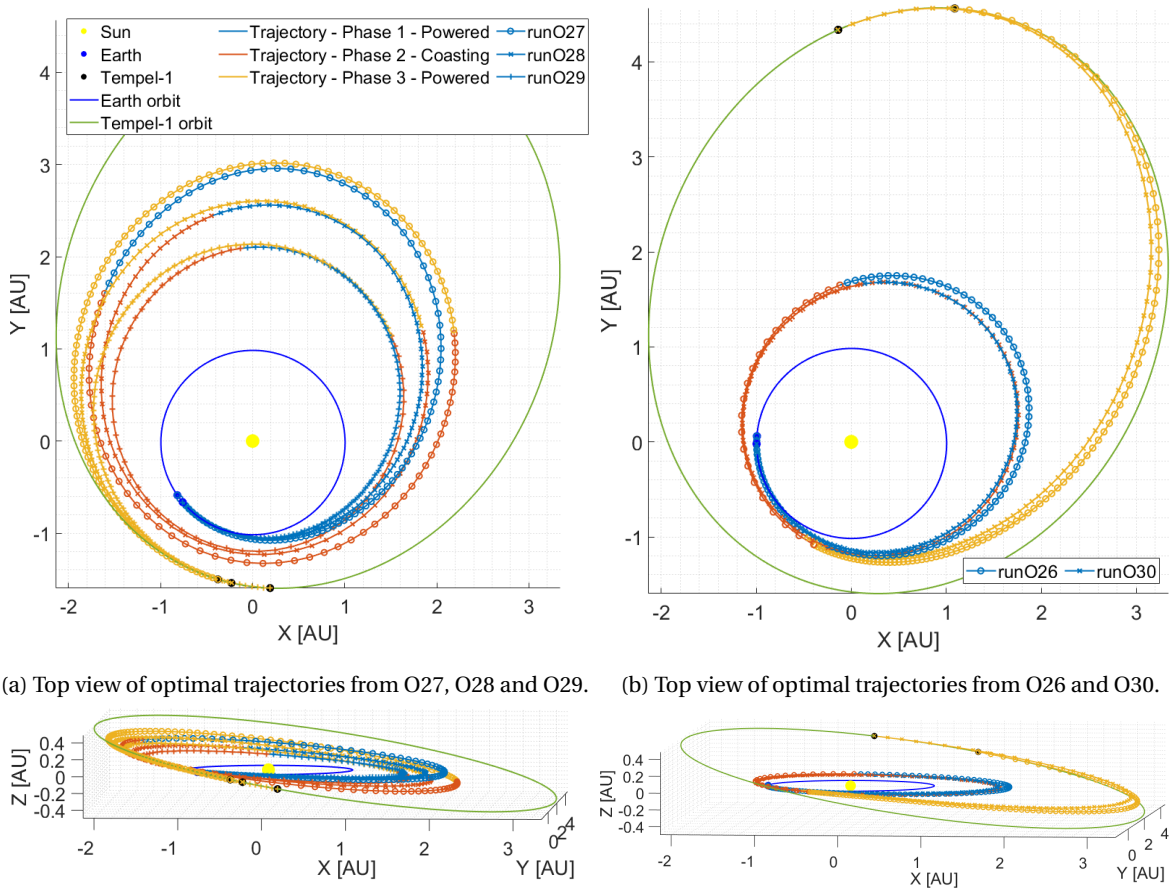
The results show the five main areas of interest with respect to start date and show that each converges on a tight cluster. Similar observations were made for the other input variables. While the time of flight converges as well, there is some more variance in the separate time of flights for the phases.

It can be seen that after 500 generations the results seem to properly converge in the local minima, and all minimum seem to have similar results of around 11.9 to 12.5 km/s. The actual minimum per cluster as well as their inputs are shown in Table 7.11. Combining the inputs as shown in the tables with the trajectories as shown in Figure 7.6, commonalities in the inputs can be observed.

Table 7.11: Group A - Inputs for each of the trajectories for the local minima with respect to $\Delta V$, specified per optimisation run.

| Run Nr | Start Date [MJD2000] | TOF1 [days] | TOF2 [days] | TOF3 [days] | TOF [days] | $r$ [AU] | $\theta$ [deg] | $\phi$ [deg] | $V_r$ [km/s] | $V_{theta}$ [km/s] | $V_{phi}$ [km/s] | $\Delta V$ [km/s] | $a_{peak}$ [mm/s$^2$] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| O26 | 7380.5 | 555.8 | 210.8 | 1098.5 | 1865.1 | 1.65 | 94.139 | 1.215 | -6.272 | 21.425 | 0.720 | 12.105 | 1.224 |
| O27 | 7785.5 | 904.6 | 543.1 | 913.9 | 2361.6 | 2.28 | 135.104 | 6.573 | -7.478 | 17.620 | 0.847 | 12.150 | 1.157 |
| O28 | 8156.3 | 605.5 | 636.4 | 758.1 | 1999.9 | 2.49 | 100.297 | 3.005 | -3.012 | 15.391 | 1.343 | 12.275 | 1.186 |
| O29 | 8521.7 | 455.4 | 580.2 | 624.6 | 1660.2 | 2.09 | 93.704 | 2.056 | -1.283 | 17.507 | 1.388 | 12.518 | 0.973 |
| O30 | 8846.2 | 494.0 | 867.2 | 1288.3 | 2649.5 | 1.65 | 87.793 | 0.768 | -5.151 | 21.078 | 0.604 | 11.929 | 1.317 |

It can be observed that while the three trajectories, from runO27,O18 and O19, have the same overall shape, they differ in the duration for the three phases of flight. Looking at the thrust profiles as shown in Figure 7.7, it can be observed that the three trajectories have a similar profile and reinforce the variation in phase duration. It seems that at the start and end of the trajectories the thruster is needed most, while the middle part of the trajectory is needed to get into position for arriving at Temple-1.




(a) Top view of optimal trajectories from O27, O28 and O29.

(b) Top view of optimal trajectories from O26 and O30.

(c) Side view of optimal trajectories from O27, O28 and O29.

(d) Side view of optimal trajectories from O26 and O30.

Figure 7.6: The optimal trajectory with respect to $\Delta V$ per optimisation run for group A, with the inputs as defined in Table 7.11.

The three trajectories have two main thrust peaks which can be connected to trajectory changes seen in Figure 7.6a. The first one at the start is to mainly increase the radius of the trajectory as well as the inclination of the trajectory. The second peak is to further finalise the inclination as well as meet Temple-1 at the required velocity. It could be expected that these trajectories are optimal in terms of $\Delta V$ as normally speed changes are more efficient at high speeds (closer to the central body). In Figures 7.6c and 7.7 it can be seen that the inclination is changed at the start peak and more gradually during the remainder of the first phase, which is to be expected as plane changes are more optimal at lower speeds (further away from the central body). The remainder of the inclination changes are done gradually during the second phase of the trajectory, with a larger increase near the end.

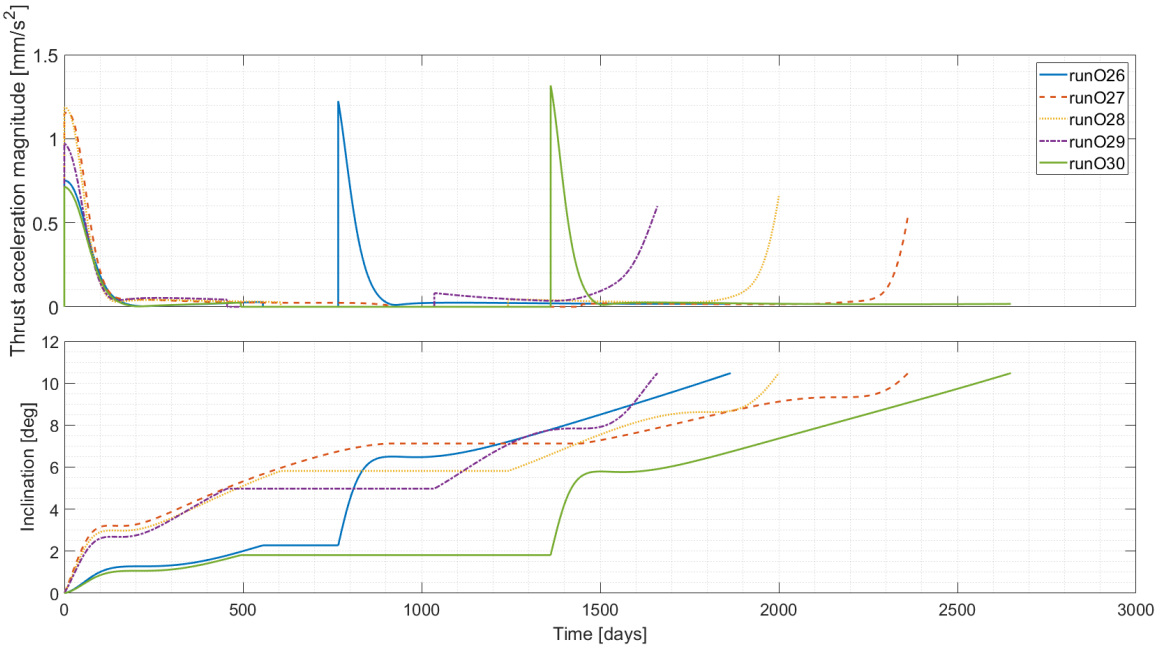



Figure 7.7: The thrust profile (Top). Inclination over time (Bottom). Results for the optimal trajectories with respect to $\Delta V$ per optimisation run for group A, with the inputs as defined in Table 7.11.

Similar observations can be made for the optimal trajectories found in runO26 and runO30. In Figures 7.6b and 7.6d the relative position of Earth and Tempel-1 are very similar for both trajectories, however in this case they do not arrive at the same time (Tempel-1 has made a full revolution around the Sun). As the relative position is the same, the trajectories themselves seem the same as well. A coasting time of 4 times longer can be found; this is seen when looking closer at the trajectory of runO30, it has an extra orbit around the Sun, thus waiting for the better relative position of Tempel-1.

The thrust profiles for these two trajectories have two peaks as well. The first one at the start is to increase the velocity, and allow a revolution around the Sun; note that the peaks themselves are lower than for the other 3 trajectories. The second peak occurs after the coasting phase and combines radius raising as well as a large inclination change. The thrust then is overall low, slowly making the final adjustments for meeting Tempel-1's orbit, with a slight increase near the end. Both peaks make sense from an energy optimisation point of view as discussed before.

All results seem to have similar values in terms of $\Delta V$s and similar to those found with non-coasting phases in the previous chapter, which also in a way validates the current implementation of the problem.

It is observed that runO27-O29 as shown Figure 7.6a have start dates which are spaced apart around a year, which results in the same position relative of Earth at departure. The total time of flight also has a difference of around a year but in opposite direction, which results in similar arrival time at Tempel-1. The three spacecraft would thus depart from Earth a year apart, but would arrive at approximately at the same time and location.

## 7.4. RESULTS IP-C-IP-SOO TRANSFER

This section presents the results for analysis group B, with the transfer consisting of two inverse polynomial shaping arcs and the analysis uses a single-objective optimiser.

Similar to group A, first the results of the general optimisation are shown, with those of runO9 as a representative run shown in Figure 7.8. Five rough clusters of local minima can be observed, and these were further investigated in the detailed optimisation. This figure also shows the effect of having many inputs and the difficulty to converge on the optimal solution.

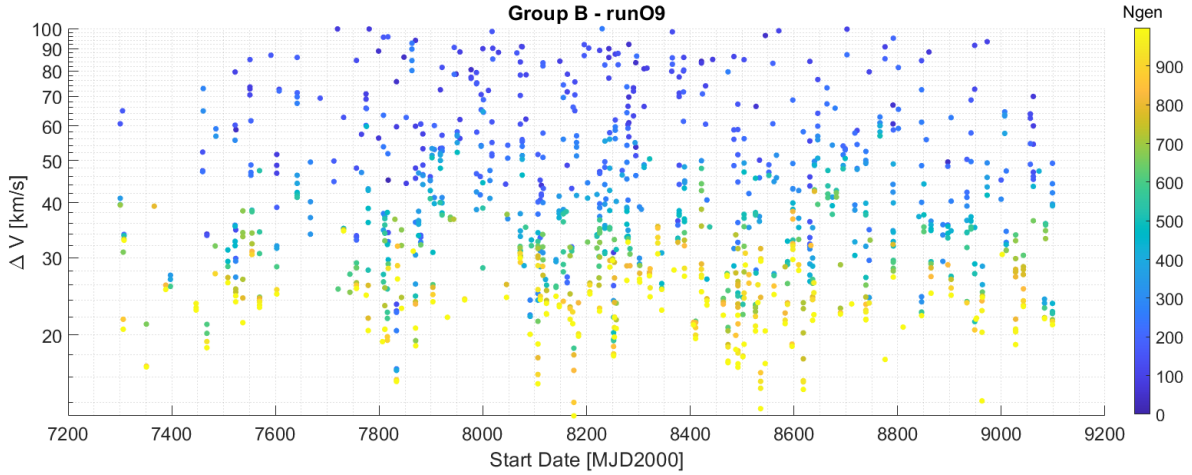



Figure 7.8: General Optimisation - $\Delta V$ plot for the Earth-Tempel-1 trajectories computed with inverse polynomials over the indicated generations (Ngen), for the given optimisation run.

In Figure 7.9 the compilation of runO17-19 and runO36-38 is shown; the plot is zoomed-in with respect to $\Delta V$ to highlight the clustering of later generations. It is seen that after generation 500 the solution has been found. A repetitive spacing can be seen between the 'pillars', which is the same pattern as observed with group A. It is thus likely related to the relative positions of Earth and Tempel-1 and not necessarily to the methods themselves.

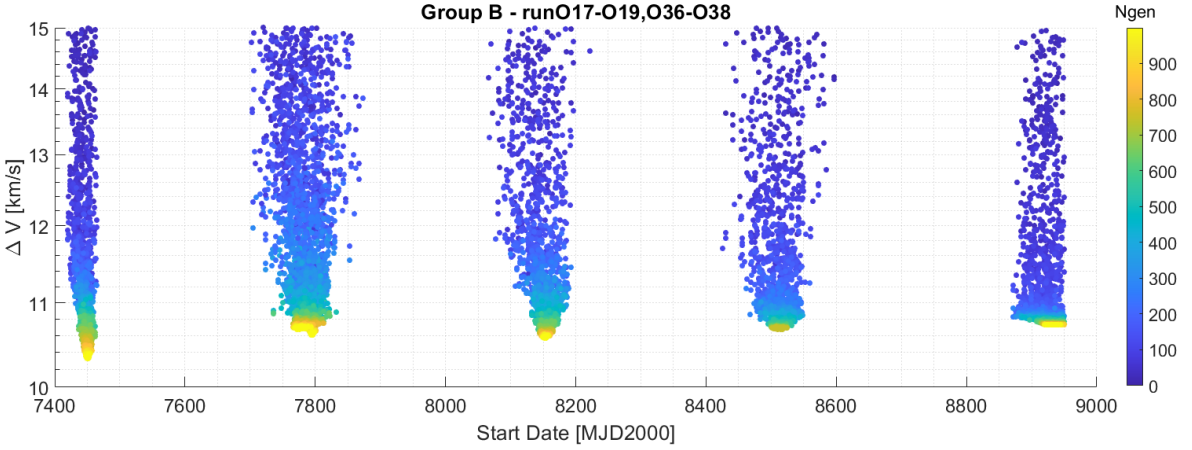



Figure 7.9: Detailed Optimisation - $\Delta V$ plot for the Earth-Tempel-1 trajectories computed with inverse polynomials over the indicated generations (Ngen), for the given optimisation run.

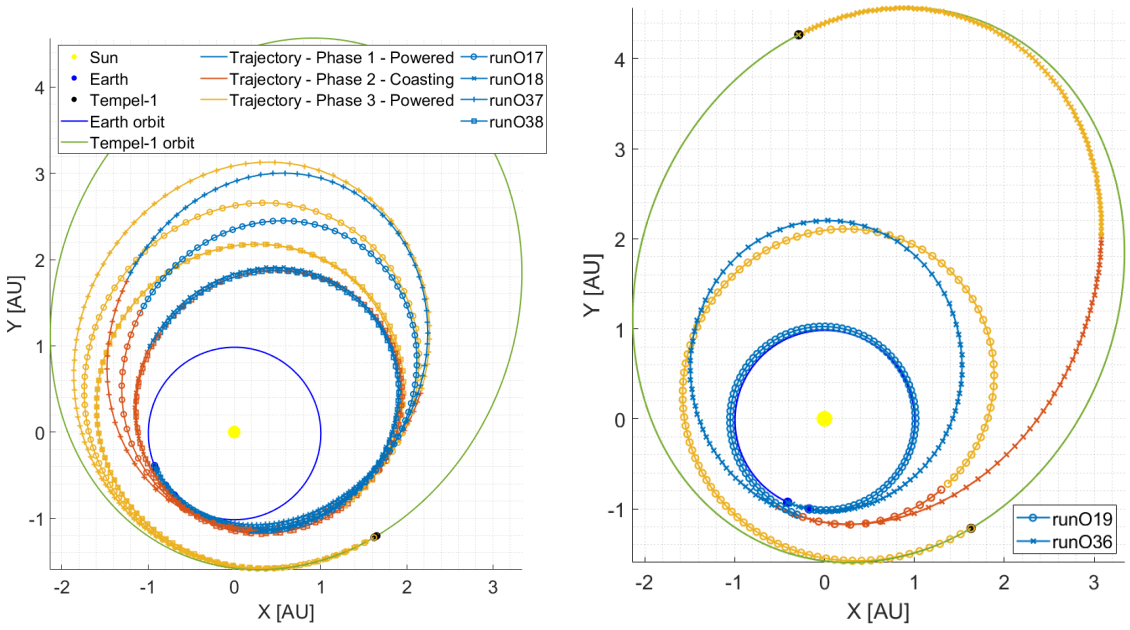
The best results in terms of $\Delta V$ per optimisation run are shown in Table 7.12 together with its required inputs. The right-most pillar contains two runs, meaning two distinct solutions were thought to have been found. Similar as with analysis group A, trajectories were found with commonalities, and will be discussed together.

Table 7.12: Group B - Inputs for each of the trajectories for the local minima with respect to $\Delta V$, specified per optimisation run.

| Run Nr | Start Date [MJD2000] | TOF1 [days] | TOF2 [days] | TOF3 [days] | TOF [days] | $r$ [AU] | $\theta$ [deg] | $\phi$ [deg] | $V_r$ [km/s] | $V_{theta}$ [km/s] | $V_{phi}$ [km/s] | $\Delta V$ [km/s] | $a_{peak}$ [mm/s$^2$] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| O17 | 8152.8 | 774.9 | 310.6 | 1042.8 | 2128.3 | 1.76 | 128.337 | 0.034 | -9.646 | 20.830 | 0.032 | 10.568 | 1.999 |
| O18 | 8505.5 | 641.1 | 221.5 | 915.7 | 1778.3 | 1.37 | 137.395 | 0.040 | -8.599 | 25.619 | 0.028 | 10.666 | 1.693 |
| O19 | 8927.6 | 376.7 | 106.1 | 871.4 | 1354.3 | 1.12 | 257.494 | 0.121 | 3.999 | 31.815 | -0.003 | 10.731 | 1.647 |
| O36 | 7451.8 | 749.1 | 431.5 | 850.8 | 2031.5 | 1.12 | 242.005 | -1.834 | -1.138 | 35.713 | -1.475 | 10.325 | 1.694 |
| O37 | 7796.5 | 923.2 | 414.8 | 1147.9 | 2485.9 | 2.15 | 125.605 | 0.084 | -9.812 | 17.515 | 0.055 | 10.609 | 2.175 |
| O38 | 7772.4 | 628.7 | 963.5 | 916.8 | 2509.0 | 1.39 | 132.211 | 0.034 | -8.481 | 25.356 | 0.025 | 10.674 | 1.666 |

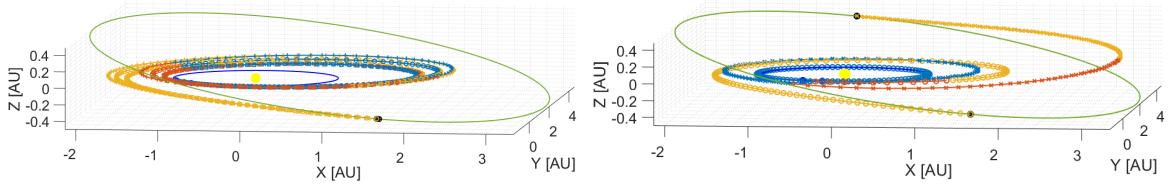
RunO17-18 and runO37-38 all have similar trajectories as seen in Figure 7.10a; these have the same relative position for departure and arrival. They begin at different start dates, however they all share the same arrival date (start date + TOF). This group is directly comparable to runs O27-29 from analysis group A, as such they share most of the remarks made there.

The main difference is the second peak of the thrust profile as seen in Figure 7.11, which is now more of a hill. Using Figure 7.10c, it is seen that this hill corresponds to a smooth, but larger plane change than seen in group A. Note that the plane change already gradually starts before the thrust hill. A smaller but important difference is the magnitude of the peak acceleration, which here is 1.6-2 mm/s$^2$, while it was 1-1.2 mm/s$^2$ before. Optimising for $\Delta V$ seems to make the thrust profiles tend towards higher and shorter peaks.




(a) Top view of optimal trajectories from O17, O18, O37 and O38.

(b) Top view of optimal trajectories from O19 and O36.




(c) Side view of optimal trajectories from O17, O18, O37 and O38.

(d) Side view of optimal trajectories from O19 and O36.

Figure 7.10: The optimal trajectory with respect to $\Delta V$ per optimisation run for group B, with the inputs as defined in Table 7.12.

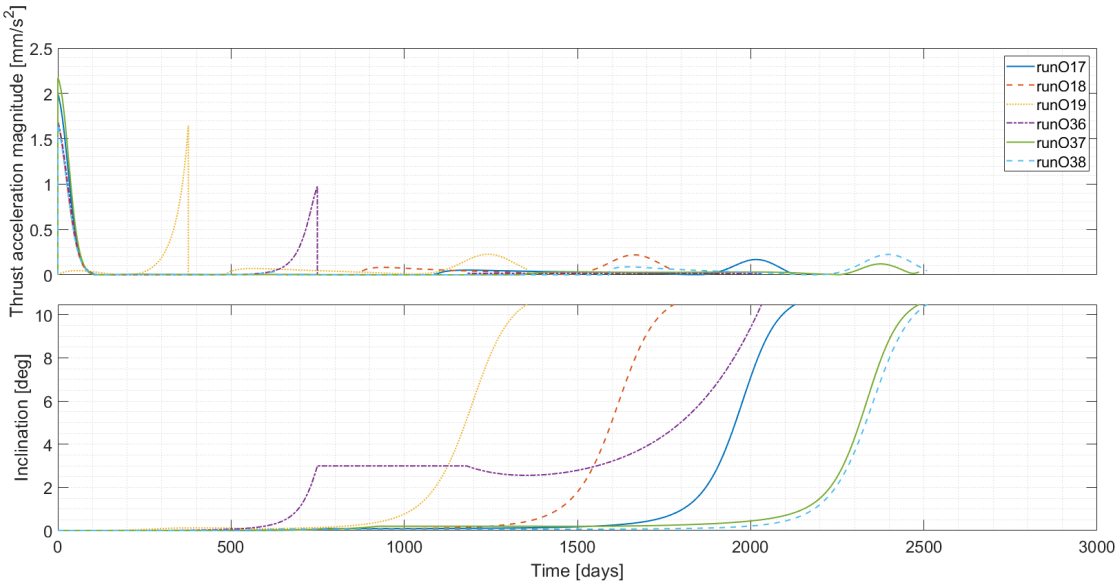



Figure 7.11: The thrust profile (Top). Inclination over time (Bottom). Results for the optimal trajectories with respect to $\Delta V$ per optimisation run for group B, with the inputs as defined in Table 7.12.

During optimisation it was thought that runO37 and runO38 would result in two distinct trajectories. It is now clear that while they have the same start date and time of flight, their trajectories are indeed different. Run O38 seems to overlap O18 in Figure 7.10a, which is only possible because O38 makes an extra revolution around the Sun to allow for a long time of flight. This similarity reinforces the found optima as well. This also confirms the concept that TOFs of the individual phases do not matter that much, however where the thrust peaks occur does.

Figure 7.10b shows the optimal trajectories for runO19 and runO36, these two are not grouped for their common shape, rather their different shape from those already discussed. The trajectory from runO19 is the most similar to the four already discussed trajectories, with the same relative position of the bodies, and the same arrival time at Tempel-1. What makes it distinct is its extra revolution around the Sun and its short coasting phase. Looking at the thrust profile in Figure 7.11, it is seen that the start peak simply occurs later, however its end peak is similar to the other four. This reinforces the idea that its mostly a matter of optimal relative position and that there are two main types of manoeuvres. Viewing the inclination change in the same plot, it is observed that the first peak combines the orbit raising and a first plane change manoeuvre.

RunO36 shares a similar shape and thrust profile as the trajectories from runO26 and runO30, as such the same comments made there apply here. The main difference is changing the inclination most near the end of the trajectory, thus at lower speeds and thus being more efficient. This also explains why the peak thrust is lower than those for the other two trajectories, as the inclination change is done more gradually.

## 7.5. RESULTS IP-C-IP-MOO TRANSFER

This section presents the results for analysis group C, with the transfer consisting of two inverse polynomial arcs and the analysis uses a multi-objective optimiser.

Figure 7.12 shows the required $\Delta V$ for the trajectory for a given start date, and again the results are separated per generation and denoted with colour. It is seen that once again five regions of interest are found, these regions have been used for the detailed optimisation. Compared to groups A and B the multi-objective optimiser is very fast and able to already find optima similar to those found in the detailed search of groups A and B. It seems that the addition of the peak thrust acceleration as fitness value enhances the optimisation. Another reason could be the use of a different optimiser. As explained before the faster convergence allows for limiting the start date input range.

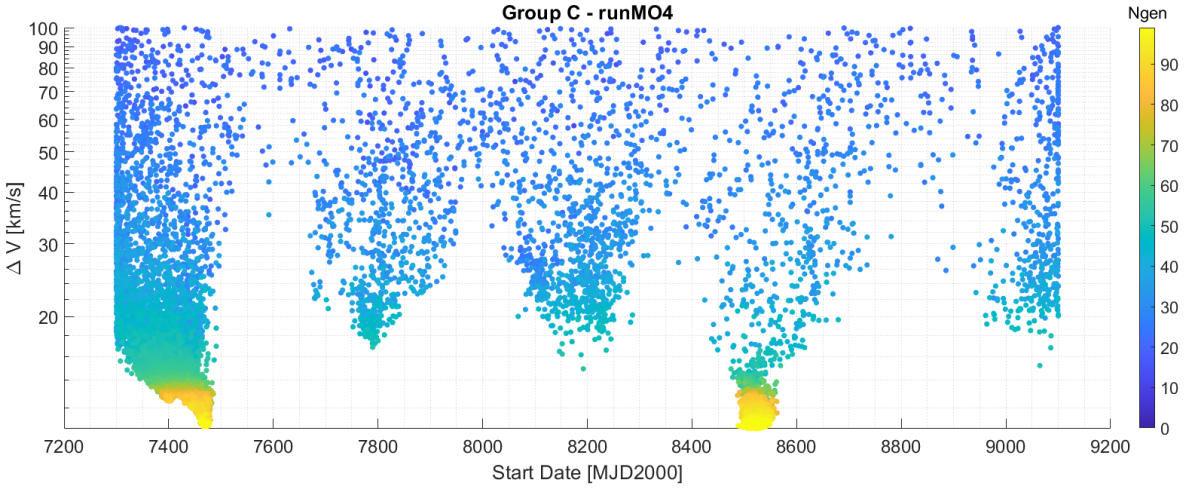



Figure 7.12: General Optimisation - $\Delta V$ plot for the Earth-Tempel-1 trajectories computed with inverse polynomials over the indicated generations (Ngen), for the given optimisation run.

Figure 7.13 shows the peak thrust acceleration reached during the trajectories computed over the whole optimisation. Again five regions of interest with respect to start date are identified, although the optimiser focusses on two after 50 generations. The later generations do not seem to have the lowest achieved peak acceleration thrust. It would seem the peak acceleration at first optimises towards a low value, but after 50 generations it follows the optimisation of the $\Delta V$. This suggests that the optimisation for $\Delta V$ is dominant during the optimisation.

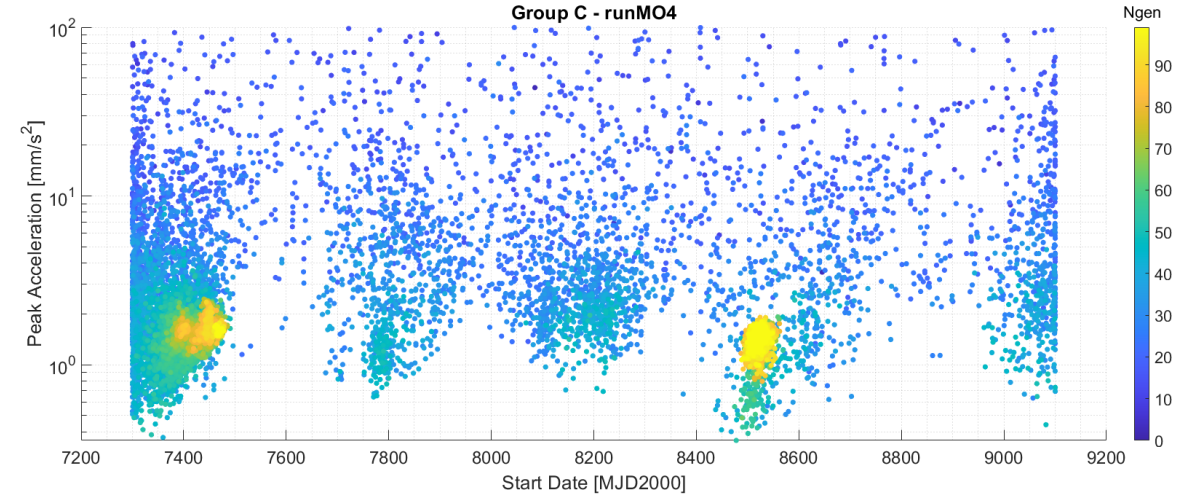



Figure 7.13: General Optimisation - Peak acceleration plot for the Earth-Tempel-1 trajectories computed with inverse polynomials over the indicated generations (Ngen), for the given optimisation run.

Figure 7.14 shows the compilation of computed $\Delta V$ from five different optimisations with their respective ranges for start date. The shown figure is zoomed in on the 10 to 15 km/s range to more clearly observe the found values per generations and final optima. After 100 generations the optimiser is already very close to the final optima $\Delta V$. The found optima are similar in terms of magnitude when compared to group B, which was with inverse polynomials as well.

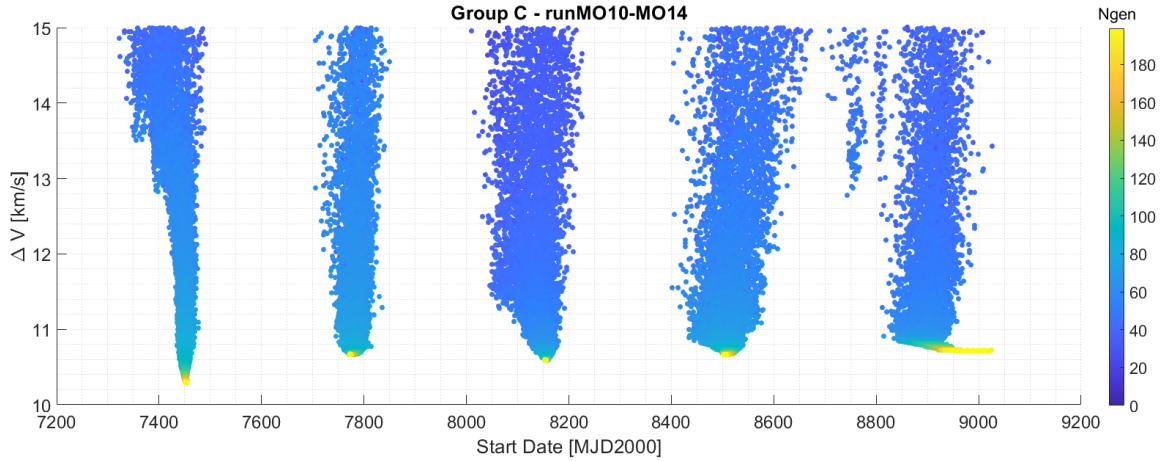



Figure 7.14: Detailed Optimisation - $\Delta V$ plot for the Earth-Tempel-1 trajectories computed with inverse polynomials over the indicated generations (Ngen), for the given optimisation run.

Figure 7.15 shows the compilation of computed peak thrust accelerations from five different optimisations with their respective ranges for start date. It is seen that the optimiser converges on five points, however it shows again that it does not converges on the lowest values. This suggests again that the optimiser is dominated by the optimisation of $\Delta V$ and that the peak acceleration mostly aids with convergence speed. The interaction between thrust and $\Delta V$ will be discussed further after the main found optima inputs, trajectories and thrust profiles have been discussed.

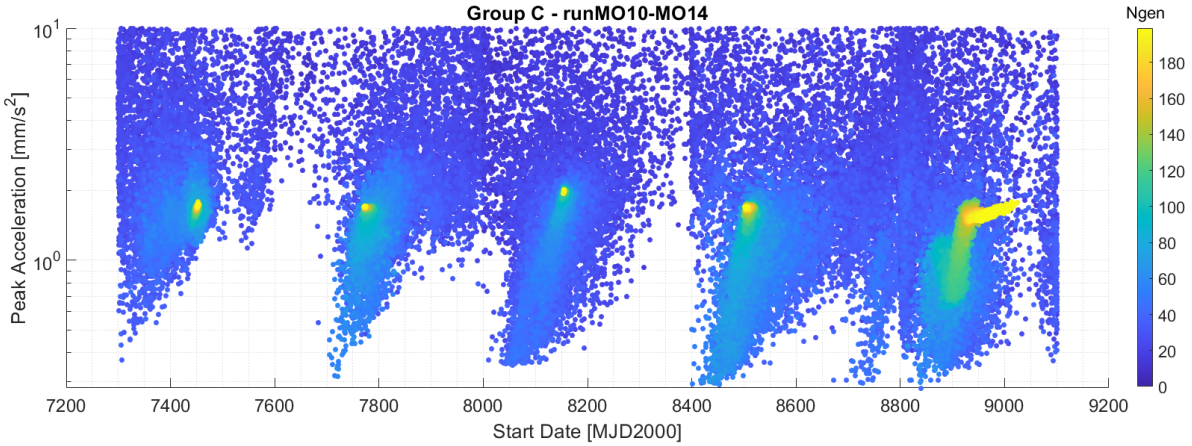



Figure 7.15: Detailed Optimisation - Peak Acceleration plot for the Earth-Tempel-1 trajectories computed with inverse polynomials over the indicated generations (Ngen), for the given optimisation run.

Table 7.13 contains the five found optima, their run numbers, their inputs and final $\Delta V$ and peak thrust acceleration. Comparing these optima with the ones from group B (Table 7.13), it can be seen that the same optima are found, and the pairs being (MO10-O36), (MO11-O38), (MO12-O17), (MO13-O18), (MO14-O19). This is to be expected as the problem set-up is equal and inverse polynomials are used for both. The only difference is that this is multi-objective optimisation, however it does use DE internally. It also confirms that the methodology performed for group B is valid and that the found optima are robust.

Table 7.13: Group C - Inputs for each of the trajectories for the local minima with respect to $\Delta V$, specified per optimisation run.

| Run Nr | Start Date [MJD2000] | TOF1 [days] | TOF2 [days] | TOF3 [days] | TOF [days] | $r$ [AU] | $\theta$ [deg] | $\phi$ [deg] | $V_r$ [km/s] | $V_{theta}$ [km/s] | $V_{phi}$ [km/s] | $\Delta V$ [km/s] | $a_{peak}$ [mm/s$^2$] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MO10 | 7452.6 | 784.7 | 496.1 | 737.5 | 2018.3 | 1.11 | 245.767 | -1.807 | -1.006 | 35.911 | -1.299 | 10.286 | 1.748 |
| MO11 | 7772.0 | 632.8 | 957.6 | 919.1 | 2509.6 | 1.43 | 127.820 | 0.034 | -8.688 | 24.635 | 0.026 | 10.661 | 1.705 |
| MO12 | 8155.1 | 749.9 | 375.9 | 1000.0 | 2125.8 | 1.94 | 121.824 | 0.049 | -8.456 | 19.246 | 0.042 | 10.585 | 2.017 |
| MO13 | 8502.8 | 590.8 | 276.1 | 911.6 | 1778.5 | 1.62 | 107.740 | 0.040 | -7.862 | 21.666 | 0.028 | 10.665 | 1.705 |
| MO14 | 8986.3 | 315.4 | 101.2 | 878.1 | 1294.6 | 1.12 | 264.604 | 0.085 | 4.827 | 31.651 | -0.026 | 10.717 | 1.663 |

As the results found here are the same as in group B, the trajectories will not be shown again. While the optimisation is dominated by the $\Delta V$ fitness value, the interaction with the peak acceleration will be investigated with a Pareto front as shown in Figure 7.16. It shows all the computed $\Delta V$s from runMO1 to runMO14 with their accompanying peak thrust accelerations, where the points belonging to the Pareto front are shown in red. The plot on the right shows a zoomed-in version, for a better view of the best results. Overall, the larger concentration of points in the negative y-direction or for low $\Delta V$ indicates again the dominance of the $\Delta V$ fitness value in the optimisation process.

What can be seen is that there are indeed trajectories with a low peak thrust acceleration (0.3-0.5 mm/s$^2$) and a relatively low $\Delta V$ (10.7-11 km/s), although they are not part of the final generations of the optimisations. The levels of thrust used on past missions is 0.037 to 0.37 mm/s$^2$, it is seen that many of the displayed results have higher peak thrusts than that. This indicates that with a different strategy for optimisation, with a larger focus on peak acceleration, better and more feasible results could be achieved. This analysis will be performed with group E.

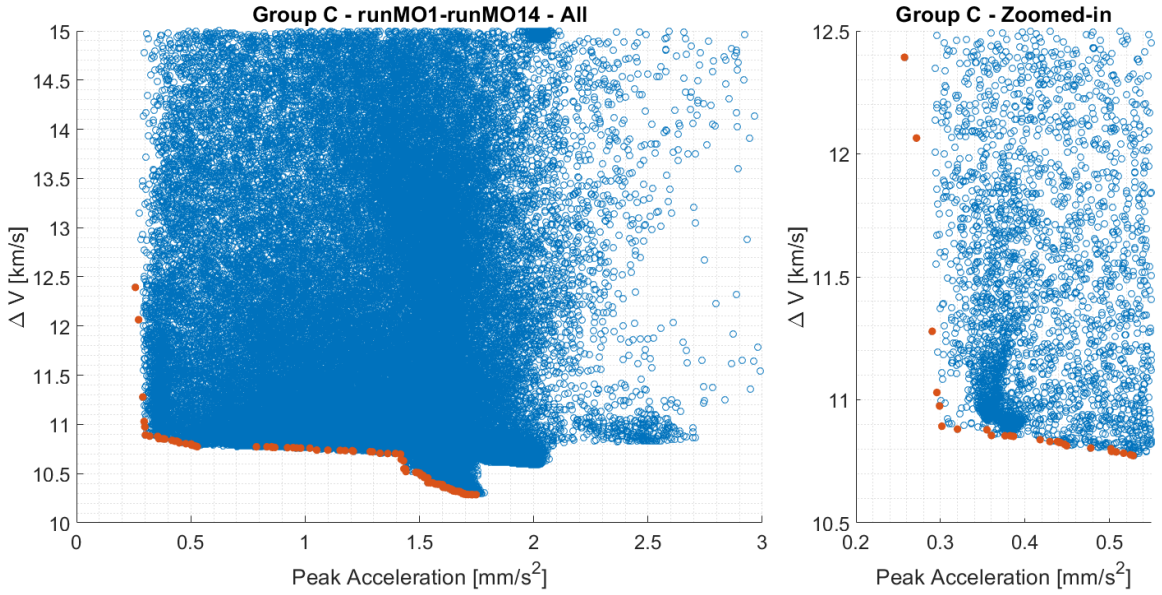



Figure 7.16: Group C - Pareto front indicated in red for all runs as indicated.

## 7.6. RESULTS IP-C-SS-MOO TRANSFER

This section presents the results for analysis group D, with the transfer consisting of an inverse polynomial for the first arc and a spherical shaping for the last arc; the analysis uses a multi-objective optimiser.

Figure 7.17 shows the used $\Delta V$ for the trajectory with a given start date, the results are separated per generation and denoted with colour. Five regions of interest are found again although some are less pronounced. Because of earlier conclusions (the presence of the 'pillars' is likely linked to the relative positions of the bodies) the lower start date regions are still further investigated.

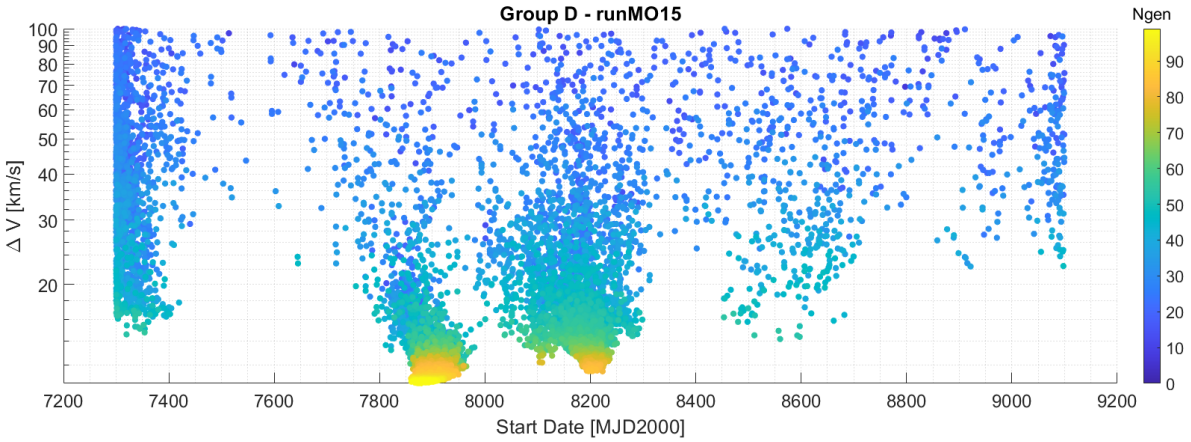



Figure 7.17: General Optimisation - $\Delta V$ plot for the Earth-Tempel-1 trajectories computed with inverse polynomials and spherical shaping over the indicated generations (Ngen), for the given optimisation run.

When looking at the peak acceleration thrust in Figure 7.18, a similar phenomenon can be seen: the $\Delta V$ drives the optimiser. What can be seen as well is the low thrust values of the pillar on the right which does seem to have influenced the optimiser to focus on that pillar. When looking back at group C, a similar but less pronounced effect can be seen: due to these low values found with respect to thrust the optimiser focuses more on those pillars, and then the $\Delta V$ decides the final optima.

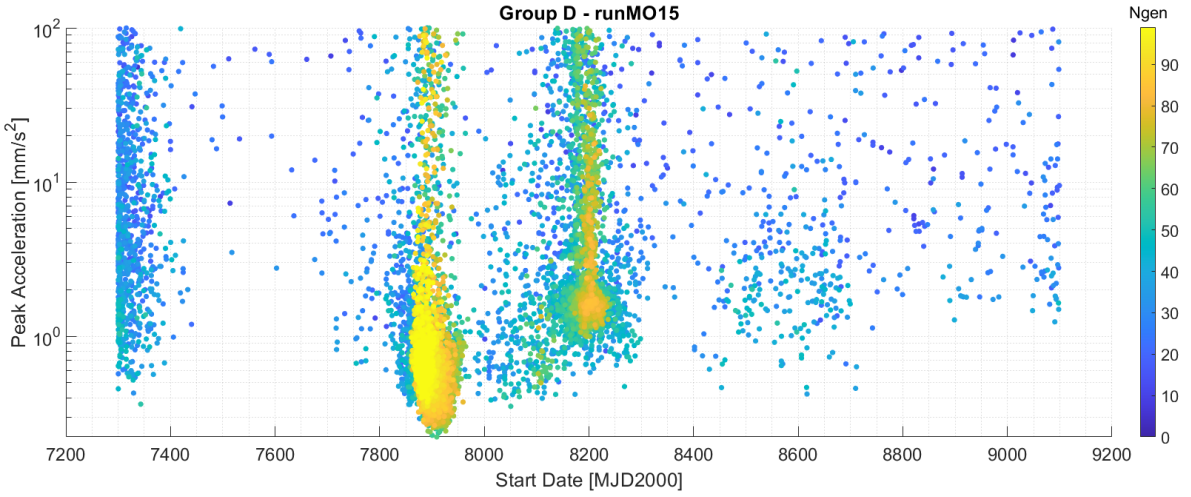



Figure 7.18: General Optimisation - Peak acceleration plot for the Earth-Tempel-1 trajectories computed with inverse polynomials and spherical shaping over the indicated generations (Ngen), for the given optimisation run.

Figure 7.19 shows the compilation of computed $\Delta V$ from five different optimisations with their respective ranges for start date for a more thorough optimisation. The figure is focused on the 10 to 15 km/s range, after 50-60 generations the optimiser is already very close to the final found optima. The found optima are between group A and B, which makes sense as these trajectories combine inverse polynomials and spherical shaping.

Figure 7.15 shows the compilation of computed peak thrust acceleration from five different optimisations with their respective ranges for start date. Once again similar effects can be seen as with group C, the peak thrust acceleration follows the lead of the $\Delta V$. Since there are more pillars per area of interest, there are more peak thrust spots as well. While the pillars $\Delta V$ graph converges to a circular cluster, the peak thrust converges to a more ellipse-like shape. This allows for more options for trajectories with limited change in $\Delta V$ requirements. After the discussion of the trajectories the balance between thrust and $\Delta V$ will be discussed using a Pareto front.

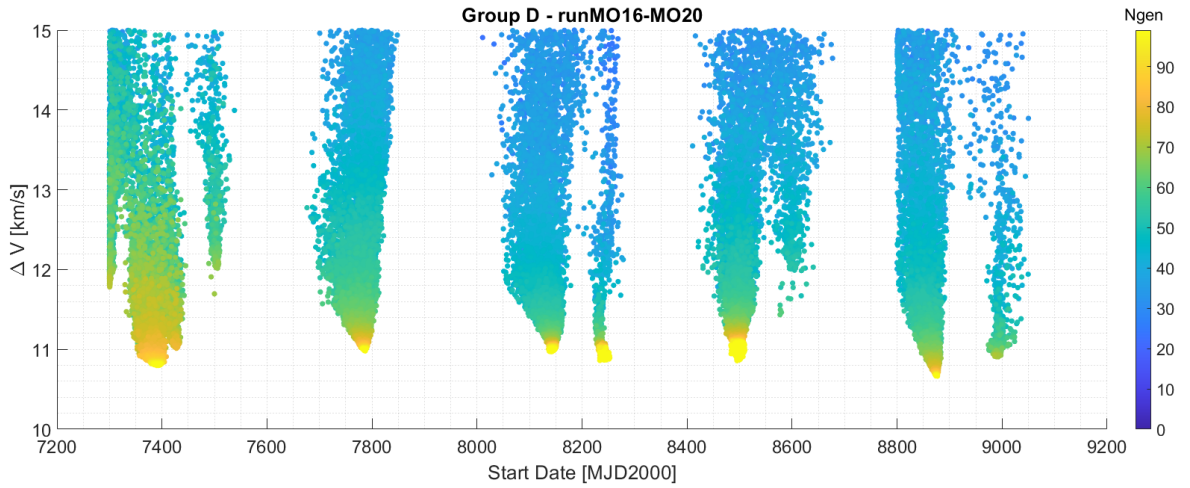



Figure 7.19: Detailed Optimisation - $\Delta V$ plot for the Earth-Tempel-1 trajectories computed with inverse polynomials and spherical shaping over the indicated generations (Ngen), for the given optimisation run.

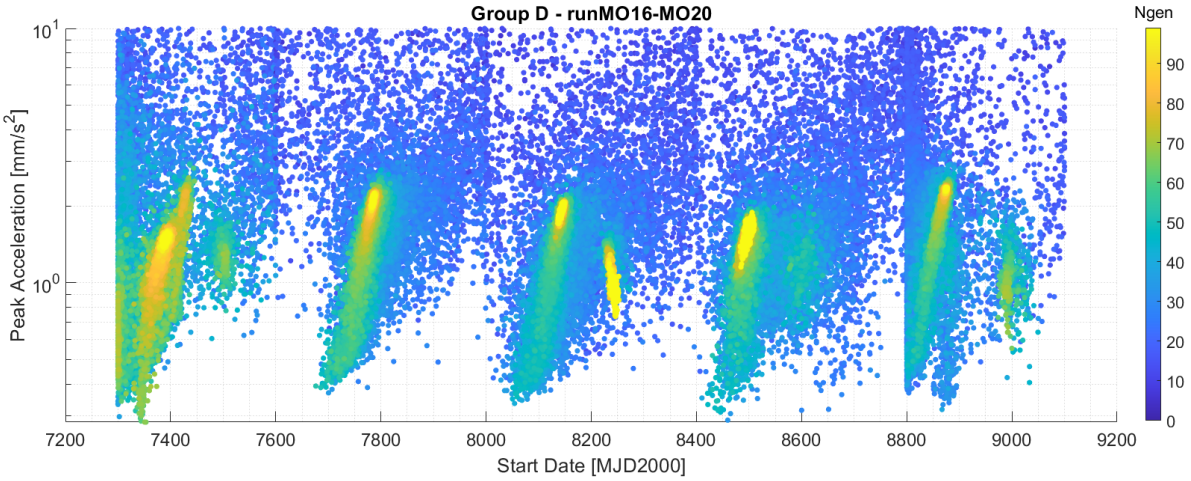



Figure 7.20: Detailed Optimisation - Peak acceleration plot for the Earth-Tempel-1 trajectories computed with inverse polynomials and spherical shaping over the indicated generations (Ngen), for the given optimisation run.

Table 7.14 shows the inputs and performance parameters of the trajectory with the lowest $\Delta V$ requirement per optimisation run. Using the five main areas of interest as before, five optima are found and as expected strong similarities to the previous analyses groups can be observed.

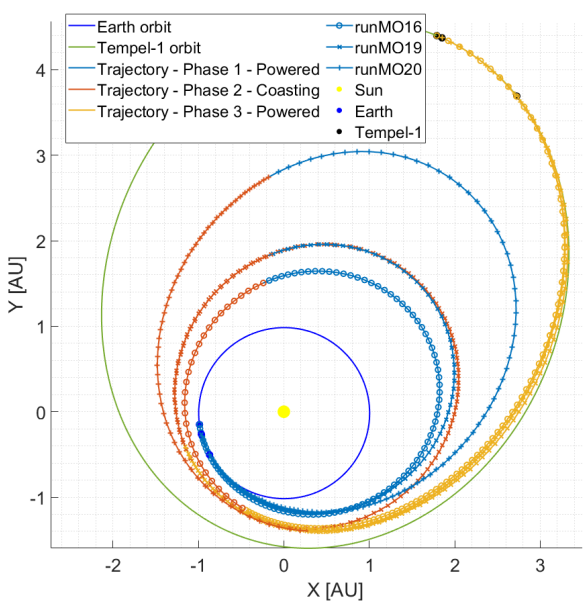
Run MO16, 19 and 20 as shown in Figure 7.21a, a similar behaviour between the three can be observed. It also shows similar trajectories as found in the previous analysis groups, as now the previously investigated shapes are combined. For run MO17 and MO18 in Figure 7.21b the same observation can be made.

The main difference of these trajectories compared to those resulting in previous analysis groups is the change of the trajectories in the z-direction as shown in Figures 7.21c and 7.21d. The inclination change now occurs from the middle to the end of the first phase, while usually they occurred at the start or end of the trajectory.

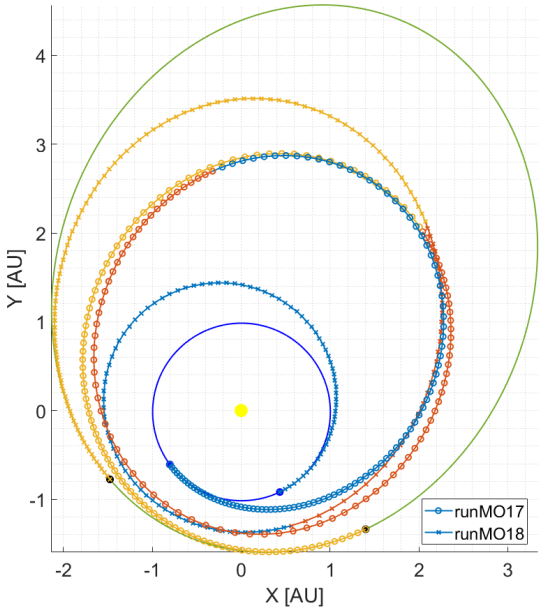
Table 7.14: Group D - Inputs for each of the trajectories for the local minima with respect to $\Delta V$, specified per optimisation run.

| Run Nr | Start Date [MJD2000] | TOF1 [days] | TOF2 [days] | TOF3 [days] | TOF [days] | $r$ [AU] | $\theta$ [deg] | $\phi$ [deg] | $V_r$ [km/s] | $V_{theta}$ [km/s] | $V_{phi}$ [km/s] | $\Delta V$ [km/s] | $a_{peak}$ [mm/s$^2$] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MO16 | 7392.6 | 530.1 | 198.3 | 992.8 | 1721.2 | 1.53 | 99.064 | 5.310 | -6.029 | 23.096 | 3.656 | 10.798 | 1.546 |
| MO17 | 7787.0 | 765.8 | 837.7 | 872.9 | 2476.5 | 2.72 | 96.816 | 4.623 | -4.459 | 14.718 | 2.350 | 10.980 | 2.201 |
| MO18 | 8233.8 | 562.8 | 338.4 | 931.1 | 1832.3 | 1.43 | 293.444 | -8.048 | 4.356 | 29.131 | -3.497 | 10.863 | 1.133 |
| MO19 | 8495.2 | 594.1 | 996.2 | 853.0 | 2443.4 | 1.84 | 96.475 | 5.056 | -5.842 | 20.075 | 3.231 | 10.858 | 1.662 |
| MO20 | 8875.5 | 935.2 | 390.7 | 938.0 | 2263.9 | 2.76 | 93.728 | 4.075 | -6.857 | 14.458 | 2.427 | 10.661 | 2.356 |

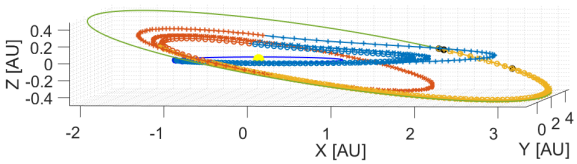
This behaviour can be observed in the thrust profile and inclination over time as shown in Figure 7.22. For 0-500 days it can be seen that the thrust profiles are elevated slightly, thus gradually increasing the inclination, with a rapid but smooth change near the end of the first phase (further from the Sun). The peak at the start of the second powered phase for runMO16 and runMO20 is there as it can be seen in the trajectory that they still require a large orbital radius change to get to Tempel-1, while runMO20 requires smaller increase in orbit radius, thus a peak.



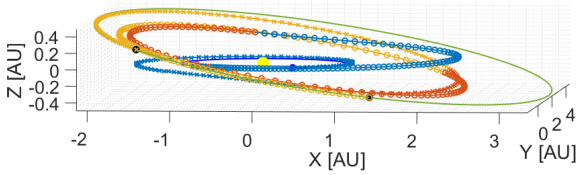

(a) Top view of optimal trajectories from MO16, MO16 and MO20.




(b) Top view of optimal trajectories from MO17 and MO18.




(c) Side view of optimal trajectories from MO16, MO19 and MO20.




(d) Side view of optimal trajectories from MO17 and MO18.

Figure 7.21: The optimal trajectory with respect to $\Delta V$ per optimisation run for group D, with the inputs as defined in Table 7.14.

In order to discuss the balance between $\Delta V$ and peak acceleration thrust, use will be made of the Pareto front using both performance parameters as shown in Figure 7.23. This figure contains all the trajectories from runs MO15 to MO20, a similar shape can be seen as with group C, a block with a larger concentration in the $\Delta V$ direction. This front seems more favourable as the lowest obtained $\Delta V$s (<11 km/s) are accompanied by the lowest attained peak thrust accelerations (0.39-0.5 mm/s$^2$). This could also already be seen in Figure 7.20, where runMO18 already had low peak thrust values (0.7 mm/s$^2$) in the last generation. Comparing these values to the feasible value of 0.37 mm/s$^2$ it again shows that while suitable thrust levels are possible, by allowing the $\Delta V$ to dominate, it leads to more final trajectories with high peak accelerations.

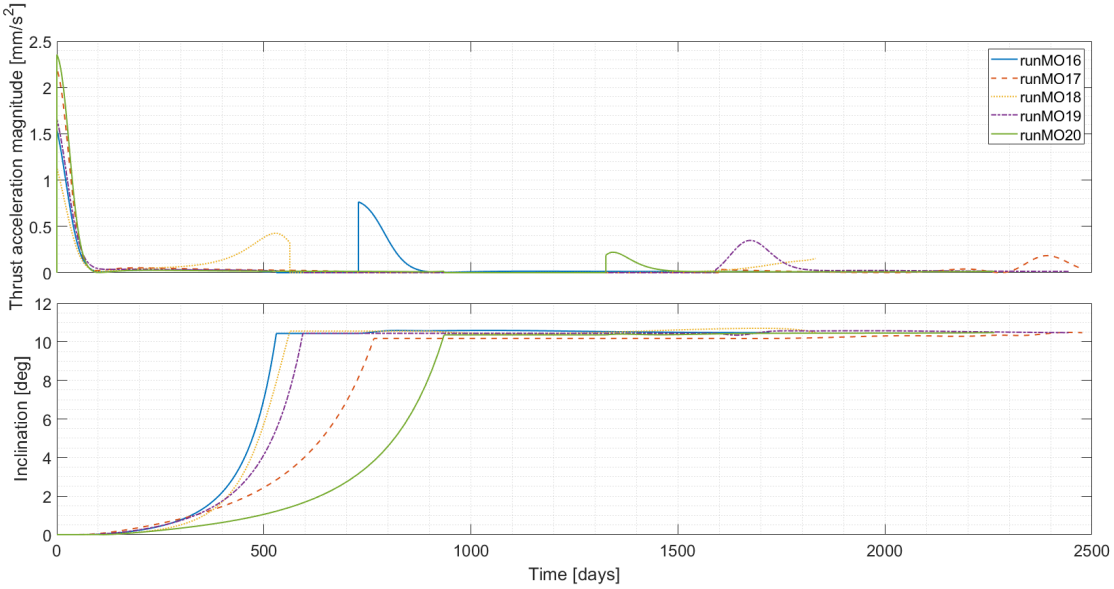



Figure 7.22: The thrust profile (Top). Inclination over time (Bottom). Results for the optimal trajectories with respect to $\Delta V$ and peak thrust per optimisation run for group D, with the inputs as defined in Table 7.14.

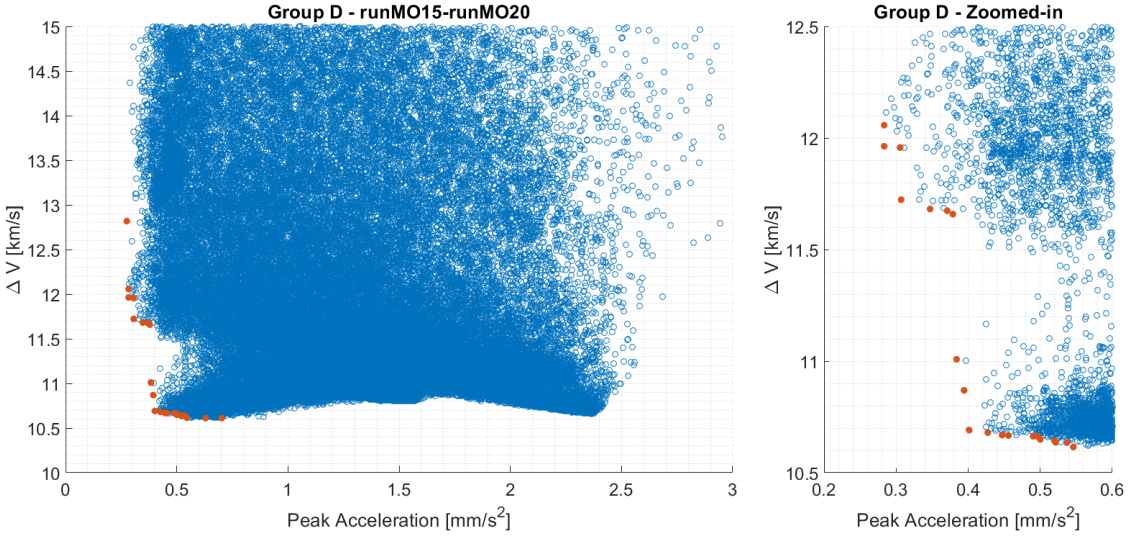



Figure 7.23: Group D - Pareto front indicated in red for all runs as indicated.

## 7.7. RESULTS IP-C-IP-MOO TRANSFER (REBALANCED)

This section presents the results for analysis group E, with the transfer consisting of two inverse polynomial arcs; the analysis uses a multi-objective optimiser with normalised fitness values.

Compared to the previous groups, the units for the fitness values for the multi-objective optimiser were changed, so the $\Delta V$ and peak thrust are of similar value, which should allow for a more balanced optimisation. Figure 7.24 shows the required $\Delta V$ for the trajectory with a given start date, the results are separated per generation and denoted with colour. The change is immediately visible: while pillars are visible, they do not converge into a point. Five main pillars are visible, with two smaller ones that drop off after generation 120. The yellow streak represents the balance between $\Delta V$ and peak thrust. For a better view of the balance a Pareto front representation can be used.

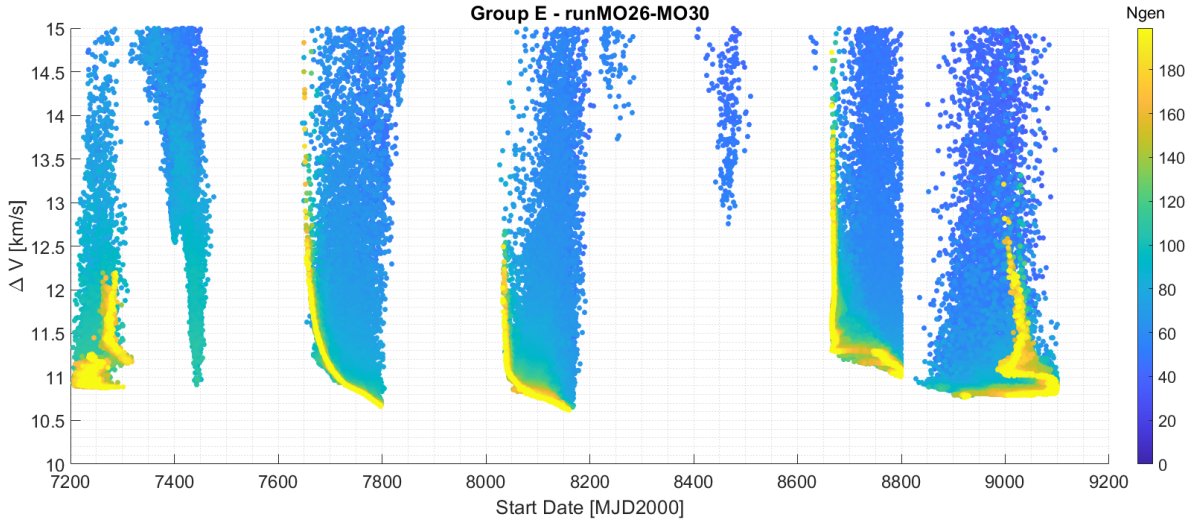



Figure 7.24: General Optimisation - $\Delta V$ plot for the Earth-Tempel-1 trajectories computed with inverse polynomials over the indicated generations (Ngen), for the given optimisation run.

Figure 7.25 shows all trajectories with the required $\Delta V$ and peak acceleration for runMO26-MO30, a Pareto front was then computed per run shown on top. The block shape as seen with the Pareto front from previous groups can be recognised. However now the concentration of points increases for both decreasing $\Delta V$ and peak acceleration thrust, which indicates a proper multi-objective optimisation was performed. It is seen that all runs have a similar behaviour, but the Pareto front of runMO27 and runMO25 dominate for the lower $\Delta V$s. it is seen that trajectories are possible at 0.24 mm/$s^2$ with a $\Delta V$ of 11 km/s with inverse polynomials. At the thrust level of past missions (0.37 mm/s$^2$) a $\Delta V$ of 10.79 km/s is possible.

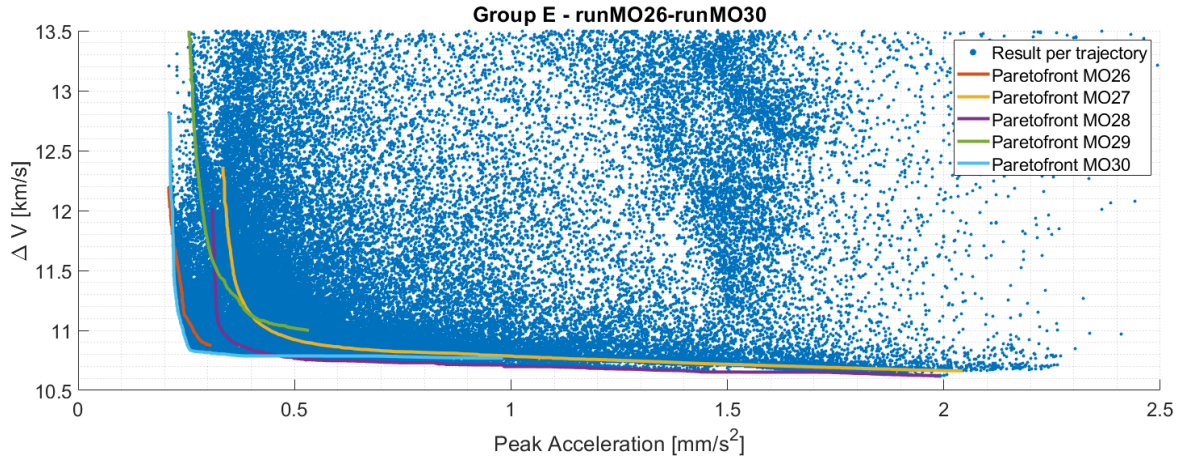



Figure 7.25: Group E - One pareto front per optimisation run.

In Table 7.15, the actual ranges for inputs for the last 10 generations have been provided. As the optimiser does not converge on a single trajectory, no single input for a trajectory can be provided. Note that the ranges of inputs are independent of each other. It can be seen that runs MO26, 29 and 30 have time of flights of the coasting phase with as little as 10 days. An unconsidered aspect of coasting is that even without a coasting phase the transfer consist of two powered phases, which is still different than a single phase. Thus the comparison between coasting and non-coasting is not only that of whether coasting has an influence, but also the fact there is an additional powered arc. It is seen that there is a wide variation in inputs. It highlights that using multiple phases allows for more flexibility and thus the ability to optimise for thrust acceleration.

Table 7.15: Group E - Inputs ranges of the final 10 generations for each of the trajectories for the local minima with respect to $\Delta V$, specified per optimisation run. UB = Upper Bound, LB = Lower Bound.

| Run Nr | UB LB | Start Date [MJD2000] | TOF1 [days] | TOF2 [days] | TOF3 [days] | TOF [days] | $r$ [AU] | $\theta$ [deg] | $\phi$ [deg] | $V_r$ [km/s] | $V_{theta}$ [km/s] | $V_{phi}$ [km/s] | $\Delta V$ [km/s] | $a_{peak}$ [mm/s$^2$] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MO26 | LB | 7215.4 | 303.4 | 10.0 | 490.5 | 934.0 | 1.37 | 0.000 | -0.442 | 1.967 | 21.595 | -0.282 | 10.878 | 0.210 |
| | UB | 7310.8 | 416.2 | 124.7 | 632.8 | 1031.2 | 1.63 | 46.114 | 1.006 | 4.780 | 25.070 | -0.056 | 12.196 | 0.312 |
| MO27 | LB | 7648.0 | 718.5 | 715.9 | 827.1 | 2483.6 | 2.49 | 76.406 | -0.406 | -4.798 | 13.635 | -0.181 | 10.663 | 0.335 |
| | UB | 7797.6 | 949.6 | 849.3 | 1000.0 | 2632.4 | 2.93 | 107.340 | -0.044 | -1.953 | 16.818 | 0.058 | 15.295 | 2.043 |
| MO28 | LB | 8031.8 | 621.8 | 297.9 | 756.3 | 2121.8 | 1.45 | 84.717 | -0.500 | -9.658 | 15.725 | -0.144 | 10.621 | 0.310 |
| | UB | 8160.4 | 857.8 | 667.9 | 980.5 | 2243.7 | 2.45 | 170.519 | 0.088 | -2.702 | 26.341 | 0.056 | 12.489 | 1.981 |
| MO29 | LB | 8666.2 | 383.5 | 10.0 | 808.1 | 1477.2 | 2.19 | 0.000 | -0.101 | 4.972 | 14.616 | -0.063 | 11.000 | 0.248 |
| | UB | 8800.0 | 573.6 | 247.0 | 1000.0 | 1613.7 | 2.74 | 32.564 | 0.428 | 7.200 | 20.402 | -0.005 | 14.718 | 0.534 |
| MO30 | LB | 8921.1 | 303.5 | 10.0 | 577.2 | 1177.6 | 1.11 | 252.571 | -0.071 | 2.961 | 22.389 | -0.095 | 10.769 | 0.210 |
| | UB | 9100.0 | 481.9 | 255.6 | 881.6 | 1358.5 | 1.73 | 336.379 | 0.638 | 6.714 | 31.854 | 0.148 | 12.817 | 0.982 |

Using the Pareto front, one trajectory per Pareto front was chosen, which are shown in Figure 7.26. The inputs and outputs are as shown in Table 7.16. The trajectories were chosen such with a combination of low $\Delta V$ and low peak thrust acceleration.

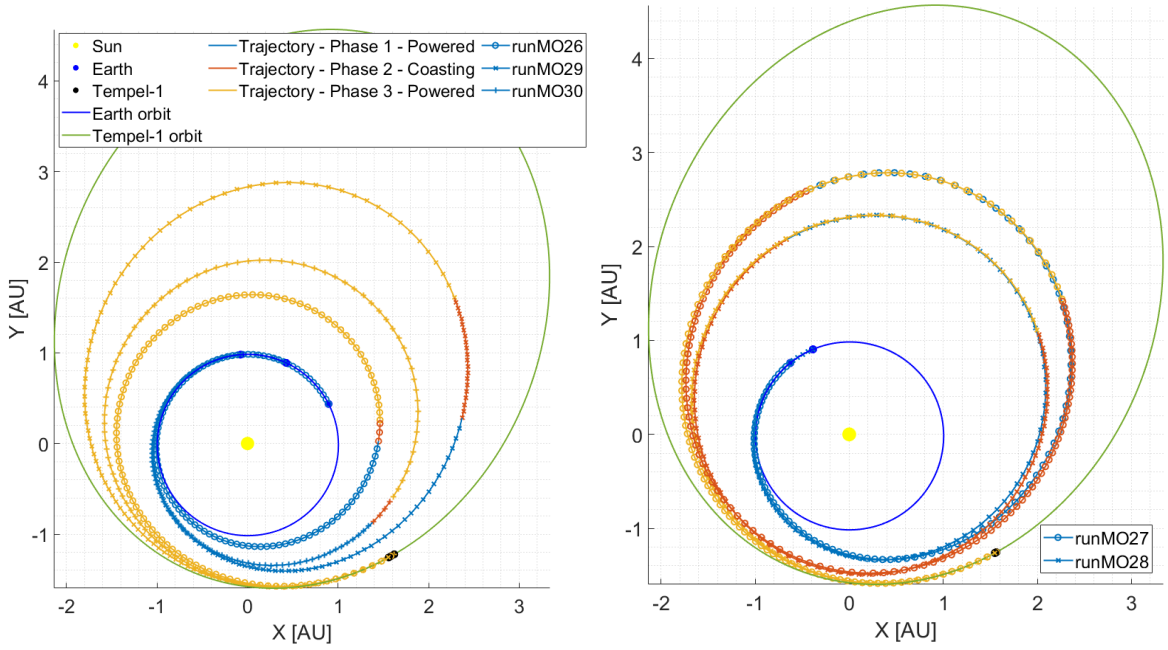
RunMO26, MO29 and MO30 have similar trajectories and are shown in Figures 7.26a and 7.26c. First of all it is observed that the coasting phases are very short (10-100 days), especially compared to its overall time of flight. The total time of flights are short as well, around 1000-1500 days. These three trajectories confirm the notion that the act of coasting is not solely responsible for the improved results. Rather multiple arcs allow for more flexibility as more inputs allow for optimisation, which was not possible for a single-arc transfer. It is seen that the first powered arc is used to increase the radius of the transfer orbit, and the second one is used for the changing of the orbital plane, as seen before with previous results.

RunMO27 and MO28's trajectories are shown in Figures 7.26a and 7.26c. The trajectories are very similar to each other, they both have a longer time of flight for coasting and in overall TOF. Here the coasting phase is just shy of a full revolution, likely allowing for better positioning of Tempel-1 at arrival. The same observation can be made as with the other three trajectories: the first powered arc is primarily for the raising of the orbit and the second one for the inclination change.

Table 7.16: Group E - Inputs for each of the trajectories for the local minima with respect to $\Delta V$, specified per optimisation run.

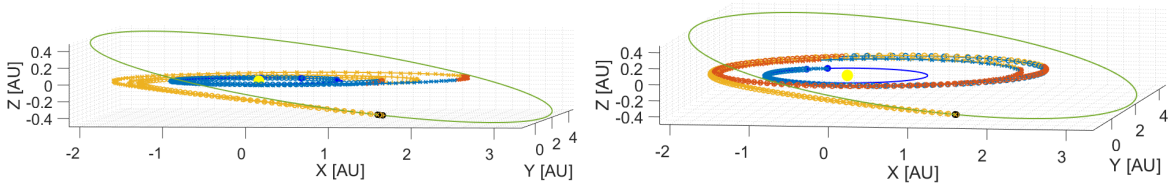
| Run Nr | Start Date [MJD2000] | TOF1 [days] | TOF2 [days] | TOF3 [days] | TOF [days] | $r$ [AU] | $\theta$ [deg] | $\phi$ [deg] | $V_r$ [km/s] | $V_{theta}$ [km/s] | $V_{phi}$ [km/s] | $\Delta V$ [km/s] | $a_{peak}$ [mm/s$^2$] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MO26 | 7231.5 | 381.5 | 16.3 | 611.0 | 1008.7 | 1.45 | 1.048 | -0.139 | 4.141 | 23.950 | -0.080 | 10.920 | 0.283 |
| MO27 | 7699.0 | 893.6 | 765.9 | 917.8 | 2577.3 | 2.63 | 100.022 | -0.181 | -4.270 | 15.659 | -0.081 | 11.130 | 0.400 |
| MO28 | 8048.1 | 797.2 | 648.6 | 781.3 | 2227.2 | 2.18 | 107.733 | -0.229 | -4.015 | 18.359 | -0.101 | 11.000 | 0.332 |
| MO29 | 8760.4 | 440.8 | 139.7 | 935.1 | 1515.7 | 2.39 | 6.843 | 0.279 | 6.351 | 17.352 | -0.048 | 11.260 | 0.370 |
| MO30 | 9096.0 | 333.1 | 25.5 | 826.5 | 1185.1 | 1.64 | 328.360 | 0.117 | 5.191 | 22.985 | -0.050 | 10.840 | 0.258 |

All trajectories have the same relative positioning of Tempel-1, and except for MO26, they all arrive at Tempel-1 at the same time. RunMO26 arrives a full orbit of Tempel-1 earlier. The sequence of manoeuvres is reflected in their thrust profiles and inclination over time as shown in Figure 7.27. The first powered arc has the peak for the orbit raising, and the second arc the next peak for further orbit raising and mainly matching Tempel-1's inclination. The main difference with the other analysis groups is that the peaks are lower and wider, explaining why they still arrive at a similar $\Delta V$, as it is the integral of the thrust acceleration. It shows that the use of multiple arcs allows for more optimal placement of the manoeuvres, and that coasting allows for more options regarding start date and arrival date.




(a) Top view of optimal trajectories from MO26, MO29 and MO30.

(b) Top view of optimal trajectories from MO27 and MO28.




(c) Side view of optimal trajectories from MO26, MO29 and MO30

(d) Side view of optimal trajectories from MO27 and MO28.

Figure 7.26: The thrust profile (Top). Inclination over time (Bottom). Results for the optimal trajectories with respect to $\Delta V$ and peak thrust per optimisation run for group E, with the inputs as defined in Table 7.15.

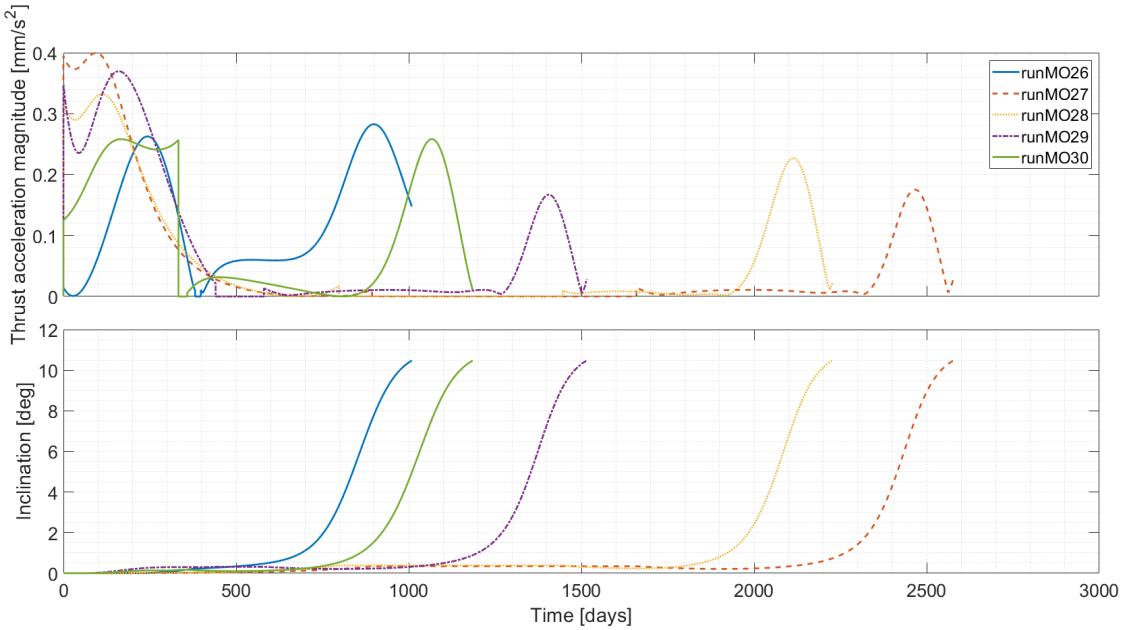



Figure 7.27: The optimal thrust profile with respect to $\Delta V$ per optimisation run for group E, with the inputs as defined in Table 7.16.

## 7.8. COMPARISON OF COASTING AND NON-COASTING METHODS

Fo far, all analysis groups have been discussed separately. This section will compare the analysis groups with each other as well as with the non-coasting results.

The first main comparison that can be made is the start date versus the required $\Delta V$ shown in Figure 7.28. This figure contains the results from the final detailed optimisations to show the most promising results and are thus only a few clusters compared to the non-coasting transfers. As the four shape-based methods have already been discussed in terms of the non-coasting transfer, they themselves will not be discussed in detail.

Comparing inverse polynomials, the IP-C-IP transfer has overall lower $\Delta V$s. However there is one cluster in which they overlap. IP-C-IP-MOO yields the absolute lowest $\Delta V$ of 10.29 km/s at 7453 MJD2000. As seen earlier the IP-C-IP-MOO achieves results which are almost equal and possibly better in terms of $\Delta V$ to those of single-objective optimisation. The results of group E are also part of IP-C-IP-MOO as they do not converge to a single point; they appear as an elongated cluster of solutions. it is seen that a wide range of start dates is possible.

Comparing the spherical shaping transfers, the SS-C-SS has similar start date and $\Delta V$s, sometimes lower, sometimes higher. The IP-C-SS-MOO transfer has overall lower results as the SS-C-SS transfer and non-coasting transfer. While it is expected that IP-C-SS-MOO would have higher values for $\Delta V$, they are closer to both IP-C-IP(-MOO) transfers than was expected. In one case at a start date of 8873 MJD2000, it has even the lowest $\Delta V$ locally (10.67 km/s vs 10.72 km/s). The latter shows the possible benefit of combining methods.

From this figure it can be concluded that IP-C-IP delivers the best results in terms of $\Delta V$, with IP-C-IP-MOO yielding the same results faster. IP-C-SS-MOO has the potential to deliver lower results locally, however could potentially yield even better results.

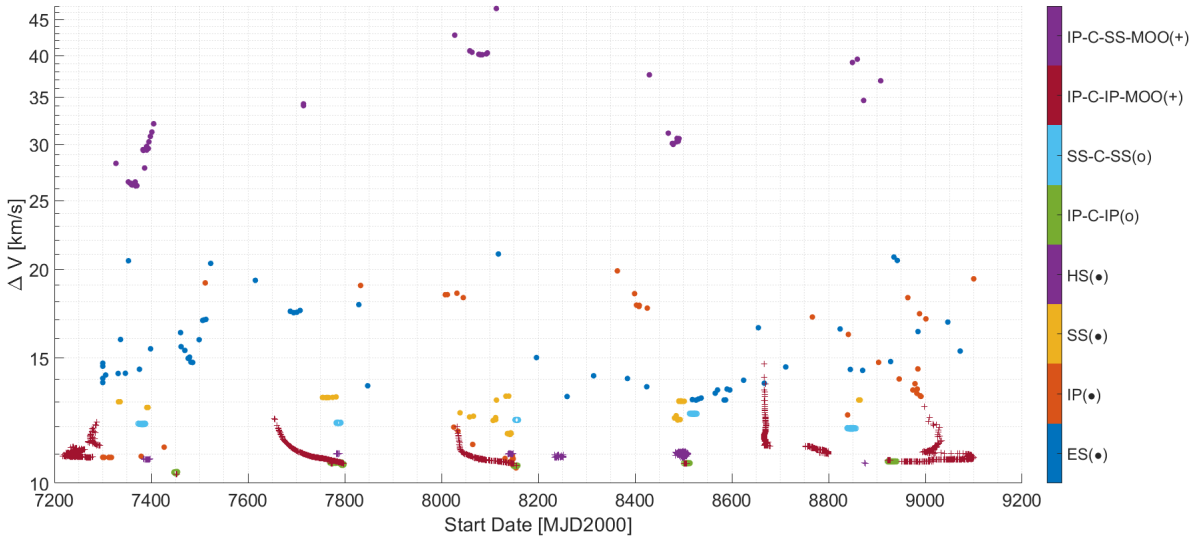



Figure 7.28: Start date vs $\Delta V$ plot combining all Earth-Tempel-1 non-coasting trajectories found with optimisation in Table 6.2 and all coasting transfers of analysis group A-E, of the final generation of the optimisation. An enlarged version is shown in Figure C.1 in Appendix C.

Figure 7.29 shows the TOF versus the required $\Delta V$. Note the non-linear scale for $\Delta V$ and changed colours of the methods compared to the previous figure. The hodographic shaping results together with exponential sinusoids were omitted as they are no longer relevant for the comparison.

Overall it is observed that the non-coasting transfers have lower TOFs compared to the coasting transfers, additionally IP has one of the lowest TOFs with relatively low $\Delta V$s. IP-C-IP(-M) has mostly the same results as IP, It yields the lowest $\Delta V$ of 10.29 km/s, but it sacrifices it for a TOF of 2010 days or 5.5 yrs. Using the rebalanced multi-objective optimiser, results with a very low TOF of 1000 days and relatively low $\Delta V$ of 10.85 kms/s can be achieved.

When comparing the spherical shaping transfer with its coasting variant, the latter has overall similar $\Delta V$s and TOFs, although a TOF never lower than 1500 days or never a $\Delta V$ as low as the non-coasting variant. IP-C-SS-MOO has lower $\Delta V$s due to the use of IP, and it also has a larger spread for TOFs, which allows for more flexibility. Similar to SS-C-SS, it has large TOFs, also never lower than 1500 days.

To trade-off $\Delta V$ and TOF, use is made of a Pareto front. The Pareto points forming this front are encircled in the figure with a large red circle. Trajectories from IP and IP-C-IP-MOO are part of this front, which means that when looking for trajectories with lowest $\Delta V$, TOF, to be able to make a trade-off, one should look at those transfers and the Pareto points. For example for a Pareto point 1: ($\Delta V$ = 11.24 km/s TOF=782 days) vs Pareto point 2:($\Delta V$ = 10.29 km/s TOF=2029 days), one could make the trade-off whether 1 km/s less is worth a TOF 1247 days longer.

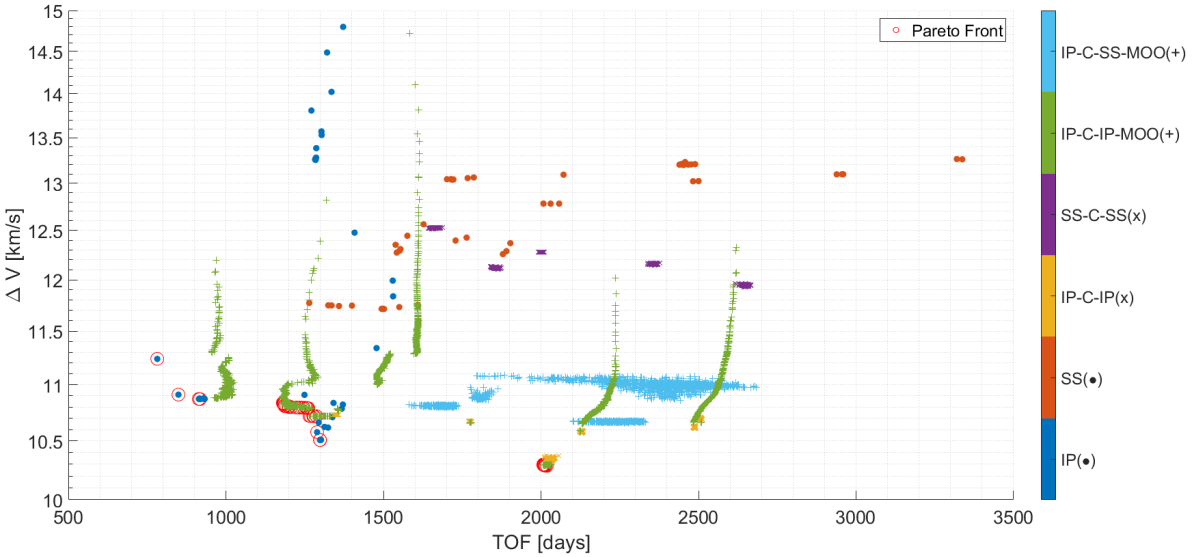



Figure 7.29: TOF vs $\Delta V$ plot combining all Earth-Tempel-1 non-coasting trajectories with IP and SS found with optimisation in Table 6.2 and all coasting transfers of analysis group A-E, of the final generation of the optimisation. An enlarged version is shown in Figure C.2 in Appendix C.

Figure 7.30 shows the peak acceleration versus the required $\Delta V$; the former will have a big influence on the required engines. This figure uses the same colours and markers for the same shape-based methods.

Overall it is seen that the non-coasting methods can achieve reasonably lower peak thrusts, were it not for the results of group E. Due to improving IP-C-IP-MOO, similar $\Delta V$s of 10.8 km/s, but with greatly reduced required peak thrust can be found, even lower peak accelerations than without coasting(<0.3 mm/s$^2$). Comparing IP-C-IP(-MOO) with IP, it is seen that the coasting methods can have similar peak accelerations for similar $\Delta V$s.

Comparing SS-C-SS with SS, they have higher peak thrusts for the same $\Delta V$s. When looking at IP-C-SS-MOO it has the lowest peak accelerations of the coasting methods, though not the lowest, at relatively low $\Delta V$s of 10.29 km/s. Thus the trade-off can be made with IP, whether 0.58 km/s required less is worth the increase in peak thrust to 1.75 mm/s$^2$, which will increase the size or number of engines, thus overall mass.

The differences between the results for trajectories with coasting arcs and without, are not only due to the presence of a coasting arc, but a use of multiple powered arcs. Originally the transfer consisted of one powered arc, with coasting it is three arcs of which two are powered. Thus with zero coasting time, it consists of two times a shape-based model, which provides the opportunity to yield different results in itself. The addition of a state between the two powered arcs allows for a larger flexibility in mission design and optimisation for the performance parameters. A next step could be to investigate the effect of using multiple powered arcs only, to evaluate the impact of using a coasting phase and using multiple phases separately.

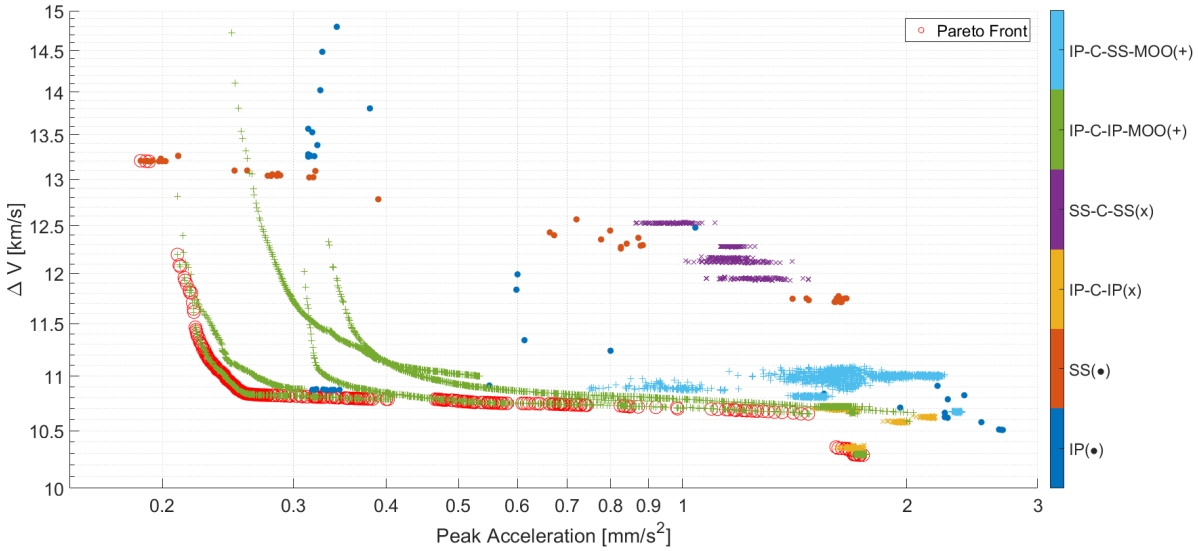



Figure 7.30: Peak thrust acceleration vs $\Delta V$ plot combining all Earth-Tempel-1 non-coasting trajectories with IP and SS found with optimisation in Table 6.2 and all coasting transfers of analysis group A-E, of the final generation of the optimisation. An enlarged version is shown in Figure C.3 in Appendix C.

## 7.9. GENERAL CONCLUSIONS

With all results shown per analysis group and comparing them with each other and the non-coasting methods, this section will present the main conclusions of this chapter.

In general for all coasting methods, the most optimal trajectories are all very similar. They all perform the necessary transfer manoeuvres where they are most optimal in terms of $\Delta V$. The radius of the transfer orbit is increased at high velocities, thus near the central body. The plane change is done at low velocities further away from the central body. This leads usually to the presence of peaks in the thrust profiles, and not really continuous thrust at a same level of magnitude. The main difference between inverse polynomials and spherical shaping in terms of manoeuvres is that the latter has a larger tendency to combine manoeuvres.

As a result of these optimal manoeuvres, the relative positions of Earth and Tempel-1 are similar for the various trajectories. The position of Tempel-1 is usually either near the pericenter or apocenter of its orbit, to facilitate the manoeuvres. Coasting usually (but not always) leads to longer time of flights and might facilitate a better relative position and optimal place for the thrust peaks.

With single-objective optimisation when comparing coasting and non-coasting transfers, it is found that coasting transfers can achieve better results in terms of $\Delta V$, however it does so in a relatively high increase in time of flight and peak thrust acceleration.

Multi-objective optimisation improperly configured in these optimisations leads to the same results as with single-objective optimisation, however it does so faster and with less manual adapting of the input ranges. It is suspected that this is both due to the optimiser and the fact that thrust is incorporated and aids during the first number of generations.

Overall for the coasting methods the early used optimisation strategy focused too much on the $\Delta V$s, which led to very large peak thrusts compared to what is currently feasible from a physical perspective. Once the proper multi-objective optimisation with balanced fitness values was used, results similar in terms of TOF and better in terms of both $\Delta V$ and peak acceleration thrust, to non-coasting transfers were found.

# 8

# CONCLUSIONS

As there are different conclusions to be made, they are separated in their respective sections. First the main research question will be answered (Section 8.1). Following the main conclusion, the conclusions on the separate parts of the thesis are provided. Then the conclusions made on the implementation and addition to the shape-based methods are given (Section 8.2). Next the conclusions on the application of all four shape-based methods to the Earth-Tempel-1 transfer will be given (Section 8.3). Finally this chapter closes with the conclusion on the application of inverse polynomials and spherical shaping on the Earth-Tempel-1 transfer case including coasting (Section 8.4).

## 8.1. RESEARCH QUESTION

***What is the benefit of using analytical low-thrust modelling in the design of interplanetary transfers?***

1. *What current methods for analytically modelling low-thrust transfer orbits in 2D and 3D can be implemented and verified in TUDAT?*
2. *How do the methods compare when applied to the same design mission to an interplanetary transfer?*
3. *Would the inclusion of a coasting arc in the design of the total transfer improve the results with respect to required propellant mass, thrust and time of flight?*

Exponential sinusoids and inverse polynomials have been successfully implemented and verified in TUDAT. Exponential sinusoids (ES), inverse polynomials (IP), spherical shaping (SS) and hodographic shaping (HS) (the latter two already in TUDAT) have been applied to an Earth-Tempel-1 transfer for a wide range of start dates and time of flights. The $\Delta V$s found are comparable to those found in literature for all methods besides HS, as it was applied in a standard setting. IP and SS produced the lowest required $\Delta V$s, with SS having slightly higher required $\Delta V$s, but with a lower peak acceleration thrust. The inclusion of a coasting arc between two powered arcs for IP and SS compared to the standard transfer, results in trajectories with similar and lower propellant mass, similar and lower peak thrust and similar and longer time of flights. It is recommended to use both IP and SS without any coasting when doing a first-order design of an interplanetary transfer, however the inclusion of coasting and the use of multi-objective optimisation with normalised fitness values could lead to improved results for peak acceleration thrust, for a limited increase in computation time.

## 8.2. IMPLEMENTATION OF SHAPE-BASED METHODS

The exponential sinusoids were simple to implement due to the existing framework for shape-based methods in TUDAT, additionally 3D functionality was implemented. The implementation was successfully verified using example calculations and validated using the figures and trajectories shown in the original paper.

The sixth-order inverse polynomials with 3D capabilities were implemented easily due to the existing framework and the previous implementation of the exponential sinusoids. Additional inequalities were derived for the 'd' or TOF coefficient, such that the TOF requirement could be met. The implementation was successfully verified using consistency tests and boundary condition tests. Validation was done successfully by reproducing similar results as produced in literature.

Automatic computation of the bounds of the free coefficient was added to the existing TUDAT spherical shaping code which allows it to be used in a grid search or optimisation program. Inequality equations for the free coefficient could be derived from the governing spherical shaping equations. This addition has been successfully verified and validated by reproducing a grid search of optimal Earth-Tempel-1 trajectories as given in the original paper.

## 8.3. EARTH-TEMPEL-1 TRANSFER

When using all four methods in the same grid search with start date and time of flight and number of revolutions around the Sun as common input, a similar pattern in $\Delta V$ results was found. A pattern with valleys and ridges was found that repeated in both inputs' directions. It was found to be linked to the relative position of Earth and Tempel-1, whereby certain relative positions yielded more optimal trajectories in terms of $\Delta V$ and thus creating these patterns. This was confirmed by observing the period in the start date direction to be the orbital period of Earth. Due to the eccentricity of Tempel-1's orbit however the pattern cannot fully repeat for low time of flights. It was found overall that the TOF's of the valleys with the lowest TOFs correspond to the Hohmann transfer time from Earth to Tempel-1, multiplied if more revolutions are used.

For Exponential Sinusoids (ES) the lowest $\Delta V$s of 14-20 km/s were found for a TOF between 500 and 2000 days and 1 to 2 full revolutions. The poor performance of ES is likely attributed to the unoptimised winding parameter, as this was not changed for the grid search. As ES has no way to satisfy the velocity boundary requirements, a high thrust burn was assumed at the start and end of trajectory, however this reduces the validity of its results during comparison.

For Inverse Polynomials (IP) the lowest $\Delta V$s of 10.68-15 km/s were found for a TOF between 700 and 2200 days and 0 to 1 full revolutions. Trajectories from 10.68-20 km/s were found for the full ranges of start date, TOF and number of revolutions. The trajectories with best $\Delta V$s were found to have peak thrust between 1 and 2 mm/s$^2$, which would put them outside the feasible range with past missions. These results are still valid as starting points for further trajectory design.

Spherical Shaping (SS) has its lowest $\Delta V$s region of 11.72-15 km/s for the full range of start dates and TOF varying from 600 to 4000 days for 0 to 2 full revolutions. For 0 and 1 full revolutions the thrust level for most of these lower $\Delta V$s is between 0.8 and 1.7 mm/s$^2$. While for 2 revolutions the thrust profiles found for TOF higher than 1000 to 2000 days stay below a peak thrust of 0.35 mm/$s^2$, the latter of which is feasible based on past missions.

For Hodographic Shaping (HS) the lowest $\Delta V$ computed is 26.8 km/s. The poor results are likely due to the improper selection of the default TUDAT base function combination, which of course dominates its behaviour. When looking at literature [28, Fig. 9.3] better results between 13-40 km/s for $\Delta V$ are found for similar inputs. This really highlights the importance of using the proper base functions.

For the thrust profiles in general it was found that they usually consisted of one or two peaks, used for increasing the transfer orbit radius, meeting the velocity of Tempel-1 and changing the inclination, the latter either combined with another manoeuvre or more gradually over time. The orbit raising was often done at high speeds and thus close to the central body as it is more efficient, and inclination changes usually at lower speeds thus farther from the central body.

When comparing the four methods it is found that the following order can be established in methods that produce lowest to highest $\Delta V$, based on an unoptimised grid search: Inverse Polynomials, Spherical Shaping, Exponential Sinusoids, Hodographic Shaping. While IP produce the single best results in $\Delta V$ with 10.68 km/s, it does so for a limited number of inputs and in combination with high peak thrust accelerations. SS on the other hand produces consistently low $\Delta V$s for a very wide range of input values and still relatively low peak thrusts.

When optimising the transfer for $\Delta V$, the input ranges are reduced according to the results found in the grid search. Using multiple seeds and optimisers, overall similar results were found. The influence of optimiser is visible, it was found that Differential Evolution (DE) found the best results in terms of $\Delta V$ compared to particle swarm optimisation and simple genetic algorithm. The results found by DE, were not influenced by the seed numbers, contrary to the other optimisers.

The ranking for best results in $\Delta V$ is the same as with the grid search. Again, the same conclusions can be made regarding the two best methods, IP and SS. It is observed that for increasing number of full revolutions the best found $\Delta V$ increases, while the peak thrust decreases. Shorter travel times require higher peak thrust but less overall integrated thrust levels and vice versa.

## 8.4. EARTH-TEMPEL-1 TRANSFER WITH COASTING

The Earth-Tempel-1 transfer with coasting consists of a powered phase, a coasting phase, and a powered phase again. Using this set-up, it was found this adds eight input variables to a total of ten, as the number of revolutions per powered phase was set to zero.

Several of the optimiser settings had to be experimented with to ensure the optimisation itself was done correctly. It was found that the size of the population needs to be at least 100 trajectories for single-optimisation and 1000 for multi-objective optimisation. The number of generations needed to converge is 100 for the single-objective optimisation and 1000 for the multi-objective optimisation. The single-objective optimisation was performed in steps, manually adjusting input ranges between the steps. Multi-objective optimisation had the advantage of being very fast and efficient. By using these numbers for populations and generations, the effect of using a different seed number and thus spread of the initial trajectories could be avoided. It was found that differential evolution and MSOED-DE works best for single and multi-objective optimisation respectively.

For spherical shaping with coasting, five distinct (in start date) trajectories with local minima of $\Delta V$ were found. They have $\Delta V$s of 11.9-12.5 km/s, and 0.97-1.31 mm/s$^2$ peak thrust. Their inputs are numerically different, however the five trajectories can be grouped in two sets of very similar trajectories, with similar thrust profiles as well.

For inverse polynomials with coasting, 6 distinct (in start date) trajectories were found. They have similar $\Delta V$s between 10.3 and 10.7 km/s, the same is true for their peak thrusts, 1.6-2.175 mm/s$^2$. Five of the six trajectories are very similar in shape and arrive at Tempel-1 at the same time. Two have the similar start date (7772-7796 MJD2000) and TOF (2485-2509 days), yet they have completely different TOFs of their individual phases. This is possible due to the thrust profiles consisting of one or two brief peaks, which can occur during any of the two powered phases.

The results for inverse polynomials with multi-objective optimisation yield the same results as with single-objective optimisation. it converges much faster and without changing the input ranges due to the addition of peak thrust as a fitness value. However, the influence on the final convergence is minimal as the peak thrust remains as with the previous optimisation, indicating the dominance of $\Delta V$.

Combining inverse polynomials and spherical shaping in one transfer and including coasting results in five distinct (in start date) trajectories with multi-objective optimisation. The found $\Delta V$s are between 10.6-11.0 km/s and 1.13-2.3 mm/s$^2$ for the peak thrust acceleration. Three of these trajectories can be grouped, as they have similar shapes and thrust profiles.

Properly using multi-objective optimisation with inverse polynomials and coasting, results in a wide range of values for $\Delta V$s and peak thrusts. $\Delta V$s between 11.0 km/s with a peak thrust of 0.24 mm/s$^2$ and 10.6 km/s and a peak thrust of 2 mm/s$^2$ can be found. Considering past missions had a peak acceleration up to 0.37 mm/s$^2$, the found results are very desirable and prove the utility of multi-objective optimisation with inverse polynomials and coasting. Trajectories are found for a wide range of inputs. Although it is confirmed that coasting is not the sole reason for better results; the mere fact there are multiple arcs, and thus shape solutions, allows for flexibility and optimisation.

In general, for all coasting methods, when optimised for $\Delta V$ the most optimal trajectories all have similar shapes. They all perform the necessary transfer manoeuvres where they are most optimal in terms of $\Delta V$, as was found during the normal transfers. Note that SS has a larger tendency to combine manoeuvres.

As a result of these optimal manoeuvres, the relative positions of Earth and Tempel-1 are similar for the various trajectories. The position of Tempel-1 is usually either near the pericenter or apocenter of its orbit, to facilitate the manoeuvres. Coasting usually leads to longer time of flights and might facilitate a better relative position and optimal place for the thrust peaks.

It was found that IP and SS in coasting transfers resulted in similar and sometimes lower $\Delta V$s than when used in the same but non-coasting transfer. The difference in lowest $\Delta V$ between coasting and non-coasting is 0.58 km/s.

The TOF for trajectories when optimising for $\Delta V$ only, is overall higher for coasting vs non-coasting trajectories. When optimising for peak acceleration as well the TOFs found for IP coasting transfers are more similar and can be higher compared to non-coasting.

The peak accelerations achieved in coasting trajectories when only optimising for $\Delta V$ are overall similar for both coasting and non-coasting transfers. For IP when optimising for both $\Delta V$ and peak acceleration, the trajectories found have similar $\Delta V$s, but lower peak accelerations when comparing coasting vs non-coasting transfers. For IP, for a $\Delta V$ of 10.87 km/s, a peak thrust was found of 0.318 mm/s$^2$ and 0.254 mm/s$^2$ for non-coasting and coasting transfers respectively.

# 9

# RECOMMENDATIONS

This chapter provides the recommendation that originate from the work done during the master thesis. It is both on the implementation of the shape-based methods in TUDAT as well as on further research that can be performed using the TUDAT shape-based methods.

For exponential sinusoids and inverse polynomials only the output functions necessary for the analysis of the transfer cases were implemented. The TUDAT framework for shape-based methods provides many more which can be defined. For the exponential sinusoids and inverse polynomials to be able to be fully used by others in TUDAT, all outputs functions need to be implemented.

The chosen base functions for the hodographic shaping method in this master thesis did not yield good results. In order to achieve desirable results, one should first experiment with these base functions. Once suitable functions are found, they can be used in transfers with coasting and compared to the already found results.

Only near the end of the work in this master thesis the multi-objective optimisation (MOO) was executed with a rebalanced fitness function. As such not a large amount of experimentation was possible with the final settings, while it is expected they would yield promising results. With this knowledge further MOO can be performed on the Earth-Tempel-1 transfer case with coasting and spherical shaping. Especially since it was found that with the non-coasting transfer spherical shaping had already a wider range of solutions with low peak thrust in combination with low $\Delta V$s.

Further single-objective optimisation could still benefit from using the multi-objective optimiser where one of the fitness values dominates, as the other could speed up the convergence reducing overall optimisation time.

It was seen that the combination of spherical shaping and inverse polynomials into one transfer can produce better results than non-coasting, however the balanced MOO was not yet applied to this problem.

The effect of using multiple powered arcs versus the use of a coasting arc in between is the topic for further research. It should be investigated in how far the use of multiple arcs or the addition of a coasting arc drive the flexibility of the transfer and its possible results.

# BIBLIOGRAPHY


[1] A. S. Moreno Gonzalez, *Implementation of and Experimentation With Shape-Based Methods in TUDAT, Literature Study, 1 September 2019, Delft University of Technology,* (2019).

[2] K. F. Wakker, *Fundamentals of astrodynamics,* Institutional Repository Library Delft University of Technology, Delft, The Netherlands (2015).

[3] R. Noomen, *Flight and Orbital Mechanics - AE2-104, lecture hours 17+18: Perturbations, Lecture Notes, 25 October 2012, Delft University of Technology,* (2012).

[4] A. E. Petropoulos and J. A. Sims, *A review of some exact solutions to the planar equations of motion of a thrusting spacecraft,* (Jet Propulsion Lab, Pasadena, CA, United States, 2002).

[5] D. J. Gondelach and R. Noomen, *Hodographic-shaping method for low-thrust interplanetary trajectory design,* Journal of Spacecraft and Rockets **52**, 728 (2015).

[6] A. E. Petropoulos, *A shape-based approach to automated, low-thrust, gravity-assist trajectory design*, Ph.D. thesis, School of Aeronautics and Astronautics, Purdue University (2001).

[7] D. Izzo, *Lambert's problem for exponential sinusoids,* Journal of Guidance, Control, and Dynamics **29**, 1242 (2006).

[8] T. Paulino, *Analytical representation of low-thrust trajectories*, Master's thesis, Delft University of Technology (2008).

[9] E. Vellutini and G. Avanzini, *Shape-Based Design of Low-Thrust Trajectories to Cislunar Lagrangian Point,* Journal of Guidance, Control, and Dynamics **37**, 1329 (2014).

[10] B. J. Wall and B. A. Conway, *Shape-based approach to low-thrust rendezvous trajectory design,* Journal of Guidance, Control, and Dynamics **32**, 95 (2009).

[11] B. Wall, *Shape-based approximation method for low-thrust trajectory optimization,* in *AIAA/AAS Astrodynamics Specialist Conference and Exhibit* (Honolulu, Hawaii, USA, 2008) p. 6616.

[12] B. J. Wall and D. Novak, *A 3D Shape-Based Approximation Method For Low-Thrust Trajectory Design,* Advances in the Astronautical Sciences **142**, 1163 (2011).

[13] M. Vasile, P. De, *et al.*, *On the Optimality of a Shape-Based Approach,* (International Astronautical Congress 2006, Valencia, Spain, 2006).

[14] D. M. Novak and M. Vasile, *Improved shaping approach to the preliminary design of low-thrust trajectories,* Journal of Guidance, Control, and Dynamics **34**, 128 (2011).

[15] A. Vroom, *Development of an Improved Spherical Shaping Method for High-Inclination Trajectories*, Master's thesis, Delft University of Technology (2017).

[16] O. Abdelkhalik and E. Taheri, *Shape based approximation of constrained low-thrust space trajectories using Fourier series,* Journal of Spacecraft and Rockets **49**, 535 (2012).

[17] O. Abdelkhalik and E. Taheri, *Approximate on-off low-thrust space trajectories using Fourier series,* Journal of Spacecraft and Rockets **49**, 962 (2012).

[18] Q. Fang, X. Wang, C. Sun, and J. Yuan, *A Shape-Based Method for Continuous Low-Thrust Trajectory Design between Circular Coplanar Orbits,* International Journal of Aerospace Engineering **2017** (2017).

[19] T. Roegiers, *Application of the Spherical Shaping Method to a Low-Thrust Multiple Asteroid Rendezvous Mission*, Master's thesis, Delft University of Technology (2014).

[20] C. Xie, G. Zhang, and Y. Zhang, *Simple shaping approximation for low-thrust trajectories between coplanar elliptical orbits,* Journal of Guidance, Control, and Dynamics **38**, 2448 (2015).

[21] K. Zeng, Y. Geng, and B. Wu, *Shape-based analytic safe trajectory design for spacecraft equipped with low-thrust engines,* Aerospace Science and Technology **62**, 87 (2017).

[22] K. Zeng, Y. Geng, B. Wu, and C. Xie, *A Novel Shape-Based Approximation Method for Constrained Low-Thrust Trajectory Design,* in *AIAA/AAS Astrodynamics Specialist Conference* (Long Beach, California, USA, 2016) p. 5637.

[23] J. Wertz and W. Larson, *Space Mission Analysis and Design*, Space Technology Library (Microcosm Press and Kluwer Academic Publishers, 1999).

[24] H. Kuninaka, K. Nishiyama, Y. Shimizu, I. Funaki, H. Koizumi, S. Hosoda, and D. Nakata, *Hayabusa asteroid explorer powered by ion engines on the way to Earth,* Proceedings of the 31st Intern. Electric Propulsion Conference, Ann Arbor, MI (2009).

[25] *Dawn at Ceres Presskit,* https://dawn.jpl.nasa.gov/news/pdf/Dawn-Ceres-presskit_20150227.pdf (2015), accessed: 17-07-2018.

[26] *By Sun power to the Moon SMART-1,* https://esamultimedia.esa.int/docs/Smart1/Final_layout_A404Ju02.pdf (2003), accessed: 17-07-2018.

[27] *GOCE Facts and Figures,* https://www.esa.int/Our_Activities/Observing_the_Earth/GOCE/Facts_and_figures (2014), accessed: 17-07-2018.

[28] D. Gondelach, *A hodographic-shaping method for low-thrust trajectory design*, Master's thesis, Delft University of Technology (2012).

[29] *JPL Planetary Data System Navigation Node,* https://naif.jpl.nasa.gov/pub/naif/generic_kernels/spk/comets/a_old_versions/, accessed: 22-08-2020.

[30] *JPL Small Body Database,* http://ssd.jpl.nasa.gov/?sb_elem, accessed: 02-07-2018.

[31] E. Gill and O. Montenbruck, *Satellite Orbits Models, Methods, and Applications* (New York: Springer-Verlag, 2000).

[32] B. De Vogeleer, *Automatic and fast generation of sub-optimal and feasible low-thrust trajectories using a boundary-value pseudo-spectral method*, Master's thesis, TU Delft, Aerospace Engineering, Astrodynamics and Satellite Systems (2008).

[33] E. Mooij and D. Dirkx, *AE4-866 - Propagation and Optimisation in Astrodynamics - Formulation of Equations of Motion, Lecture Notes, 14 February 2017, Delft University of Technology,* (2017).

[34] *Differential Evolution, official website,* http://www1.icsi.berkeley.edu/~storn/code.html (), accessed: 16-07-2018.

[35] *Particle Swarm Optimisation, repository of information,* http://www1.icsi.berkeley.edu/~storn/code.html (), accessed: 17-07-2018.

[36] P. S. Oliveto, J. He, and X. Yao, *Time complexity of evolutionary algorithms for combinatorial optimization: A decade of results,* International Journal of Automation and Computing **4**, 281 (2007).

[37] H. Li and Q. Zhang, *Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II,* IEEE Transactions on Evolutionary Computation **13**, 284 (2009).

[38] A. E. Petropoulos and J. M. Longuski, *Shape-based algorithm for the automated design of low-thrust, gravity assist trajectories,* Journal of Spacecraft and Rockets **41**, 787 (2004).

[39] *TUDAT - Shape-based method documentation* , http://tudat.tudelft.nl/tutorials/tudatFeatures/lowThrustTrajectoryDesign/shapeBasedMethods/index.html, accessed: 09-08-2020.

[40] R. Noomen, *Mission Geometry and Orbit Design - ae4-878 , Space Mission Design: Exponential Sinusoids V4-20, Lecture Notes, 19 March 2018, Delft University of Technology,* (2018).

[41] D. Gondelach and R. Noomen, *Analytical low-thrust transfer design based on velocity hodograph,* Proceedings of the AAS/AIAA Astrodynamics Specialist Conference (AAS, 2015).

[42] J. Wertz, *Orbit & Constellation Design & Management, second printing ed. El Segundo* (California: Microcosm Press, 2009).

# A

# APPENDIX A - COORDINATE SYSTEM TRANSFORMATIONS

In this appendix, the coordinate transformations that have been used are provided. The equations for the transformations are taken from [42, Appendix B, sec. 2.7.1.1 ].

### CARTESIAN - CYLINDRICAL/SPHERICAL

Using Equations A.1 and A.2, the coordinates can be transformed to go from cylindrical/spherical to Cartesian and Cartesian to cylindrical/spherical respectively.

$$\begin{cases} x = r\cos\alpha\cos\delta, & x = r\cos\theta \\ y = r\sin\alpha\cos\delta, & y = r\sin\theta \\ z = r\sin\delta, & z = z \end{cases} \tag{A.1}$$

$$\begin{cases} \alpha = \text{atan2}(x,y), & \theta = \text{atan2}(x,y) \\ \delta = \arcsin(z), & z = z \\ r = \sqrt{x^2+y^2+z^2}, & r = \sqrt{x^2+y^2} \end{cases} \tag{A.2}$$

### CARTESIAN - KEPLERIAN ELEMENTS

Using Equations A.3 to A.5, the Keplerian elements can be converted to Cartesian coordinates. Some intermediate symbols are used as well as the variables $H$ and $\mu$, the first is the specific angular momentum of the vehicle with respect to the origin of the coordinates system and the latter $\mu$ represents the standard gravitational parameter and is the product of the gravitational constant G and the mass of the central body.

$$\begin{aligned} l_1 &= \cos\Omega\cos\omega - \sin\Omega\sin\omega\cos i; \\ l_2 &= -\cos\Omega\sin\omega - \sin\Omega\cos\omega\cos i; \\ m_1 &= \sin\Omega\cos\omega + \cos\Omega\sin\omega\cos i; \\ m_2 &= -\sin\Omega\sin\omega + \cos\Omega\cos\omega\cos i; \\ n_1 &= \sin\omega\sin i; \\ n_2 &= \cos\omega\sin i; \end{aligned} \tag{A.3}$$

$$r = \frac{a(1-e^2)}{1+e\cos\theta} \qquad \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} l_1 & l_2 \\ n_1 & n_2 \\ m_1 & m_2 \end{bmatrix} \begin{bmatrix} \xi \\ \eta \end{bmatrix} \tag{A.4}$$

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} = \begin{bmatrix} r\cos\theta \\ r\sin\theta \end{bmatrix}$$

$$H = \sqrt{\mu a(1-e^2)}, \quad \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \frac{\mu}{H} \begin{bmatrix} l_1 & l_2 \\ n_1 & n_2 \\ m_1 & m*2 \end{bmatrix} \begin{bmatrix} -\sin\theta \\ (e+\cos\theta) \end{bmatrix} \tag{A.5}$$

Using Equations A.6 to A.10 the Cartesian coordinates can be converted to orbital elements. The arrows indicate that the variable is a vector, the hat on top indicates it is a unit vector. When computing $\Omega, \omega, \theta, E$ and $M$ these should be between 0 and $2\pi$ rad, thus for negative angles $2\pi$ rad is added, where $M$ represent the mean anomaly and $E$ the eccentric anomaly. The last unseen variable is N, which represents the nodal vector. The last expression in Equation A.10 contains also the iterative procedure when only M is given. It is initialised with $E_0 = M$ and then for every iteration, $E_{i+i}$ is compared with $E_i$ until the difference is small enough. Note that these equations are valid for ellipse trajectories, not for parabolae nor hyperbolae. The other versions are not presented as the missions considered during the thesis are between orbits and are not escape trajectories.

$$r = \left\|\vec{r}\right\|, \quad V = \left\|\vec{V}\right\|, \quad \vec{h} = \vec{r} \times \vec{V}, \quad h = \left\|\vec{h}\right\|, \quad \vec{N} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \vec{h} \tag{A.6}$$

$$a = \frac{1}{\frac{2}{r} - \frac{V^2}{\mu}}, \quad \vec{e} = \frac{\vec{V} \times \vec{h}}{\mu} - \frac{\vec{r}}{r}, \quad e = \left\|\vec{e}\right\|, \quad i = \arccos\frac{h_z}{h} \tag{A.7}$$

$$N_{xy} = \sqrt{N_x^2 + N_Y^2}, \quad \Omega = atan2\left(\frac{N_y}{N_{xy}}, \frac{N_x}{N_{xy}}\right), \quad \hat{\vec{e}} = \frac{\vec{e}}{e}, \quad \hat{\vec{N}} = \frac{\vec{N}}{N}, \quad \hat{\vec{r}} = \frac{\vec{r}}{r} \tag{A.8}$$

$$\begin{aligned} \omega &= sign \cdot \arccos\left(\hat{\vec{e}} \cdot \hat{\vec{N}}\right) \quad (sign = +1 \text{ if } (\hat{\vec{N}} \times \vec{e}) \cdot h > 0; \text{otherwise} - 1) \\ \theta &= sign \cdot \arccos\left(\hat{\vec{r}} \cdot \hat{\vec{e}}\right) \quad (sign = +1 \text{ if } (\vec{e} \times \vec{r}) \cdot h > 0; \text{otherwise} - 1) \end{aligned} \tag{A.9}$$

$$E = 2 \cdot \arctan\left(\tan\left(\frac{\theta}{2}\right) \cdot \sqrt{\frac{1-e}{1+e}}\right), \quad M = E - e \cdot \sin(E), \quad E_{i+1} = M + e \cdot \sin(E_i) \tag{A.10}$$

# B

# Appendix B - Spherical Shaping - TUDAT Free Coefficient Boundary Derivation

In this appendix the full derivation for finding the bounds of the free coefficient $a_2$ will be given. The theory and source equations not part of the derivation were taken from [19, Cha. 7] and can be used as a source for more details on each of the variables.

The way the bounds will be determined is to start from the point that the curvature, the parameter $D$ can never be negative, with its equation given in Equation B.1.

$$D = -R'' + 2\frac{R'^2}{R} + R'\Phi'\frac{\Phi'' - \sin\Phi\cos\Phi}{\Phi'^2 + \cos^2\Phi} + R(\Phi'^2 + \cos^2\Phi) \tag{B.1}$$

The following equations will be used in the derivation by rewriting the parameter $D$ equation. The variables shown below are also defined in Equations B.20, B.21 and B.22.

$$\begin{cases} R' = -Z'R^2 & Z' = \mathbf{R}'_k \cdot \mathbf{a}_k \\ R'' = -Z''R^2 + 2ZR'^2 & Z'' = \mathbf{R}''_k \cdot \mathbf{a}_k \end{cases} \tag{B.2}$$

Now $R'$ and $R''$ are replaced in Equation B.1 with their expressions as defined in Equation B.2. $C_1$ and $C_2$ are as defined in Equation B.15, and are independent of $a_2$.

$$0 < Z''R^2 - 2Z(R')^2 + 2(-Z'R^2)^2 + (-Z'R^2)C_1 + RC_2 \tag{B.3}$$

Once again $R'$ is replaced in the previous equation and the right hand side is cleaned up.

$$0 < Z''R^2 - 2ZZ'^2R^4 + 2\frac{Z'^2R^4}{R} - Z'R^2C_1 + RC_2 \tag{B.4}$$

It is seen that in the third term an $R$ cancels out. $Z$ is replaced with $1/R$. It is seen that all terms contain at least one $R$, thus it can be removed from each term. This does mean that the inequality sign stays the same if $R$ is positive and switches sign if it is negative, creating the first possible sign switch.

$$0 < Z''R - 2\frac{1}{R}Z'^2R^3 + 2Z'^2R^2 - Z'RC_1 + C_2 \tag{B.5}$$

It is seen that after cancelling out the $R$ in the second term, the second and third term add to zero removing them from the right hand side.

$$0 < Z''R - Z'RC_1 + C_2 \tag{B.6}$$

Now the Z', Z" and R variables are replaced with their components according to Equation B.2.

$$0 < \frac{\mathbf{R}''_k \cdot \mathbf{a}_k}{\mathbf{R}_k \cdot \mathbf{a}_k} - \frac{\mathbf{R}'_k \cdot \mathbf{a}_k}{\mathbf{R}_k \cdot \mathbf{a}_k} C_1 + C_2 \tag{B.7}$$

With a common denominator, the inequality can be rewritten.

$$0 < \frac{\mathbf{R}''_k \cdot \mathbf{a}_k - (\mathbf{R}'_k \cdot \mathbf{a}_k) C_1}{\mathbf{R}_k \cdot \mathbf{a}_k} + C_2 \tag{B.8}$$

Knowing that $\mathbf{R}_{k_L}$ is simply $\mathbf{R}_k$ without the term linked to the free coefficient and $\mathbf{a}_{k_L}$ is the coefficient vector $\mathbf{a}_{k_L}$ without the free coefficient $a_2$. The inequality can be fully rewritten.

$$0 < \frac{\mathbf{R}''_{k_L} \cdot \mathbf{a}_{k_L} + 2a_2 - \left[\mathbf{R}'_{k_L} \cdot \mathbf{a}_{k_L} + 2\theta a_2\right] C_1}{\mathbf{R}_{k_L} \cdot \mathbf{a}_{k_L} + a_2\theta^2} + C_2 \tag{B.9}$$

The following expression is true for the coefficients of the spherical shaping method.

$$[\mathbf{a}_{k_L}, \mathbf{b}_k]^T = \mathbf{A}^{-1}(\mathbf{B} - \mathbf{A}_{a_2}) = \mathbf{A}^{-1} \cdot \mathbf{B} - \mathbf{A}^{-1} \cdot \mathbf{A}_{a_2} \tag{B.10}$$

In Equations B.23, B.24 and B.25 it can be seen that the first 6 rows are for the coefficients $a$ and the last 4 for the coefficients $b$. It is seen that the coefficients $b$ are not dependant on the equations for the $a_2$ elements. Thus the equation for finding the coefficients can be safely split into two parts, of which the first part is used for further derivation.

$$\begin{bmatrix} \mathbf{a}_{k_L} \\ \mathbf{b}_k \end{bmatrix} = \mathbf{A}^{-1} \cdot \mathbf{B} - \mathbf{A}^{-1} \cdot \mathbf{A}_{a_{2_L}} a_2 = \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_1 b \end{bmatrix} - \begin{bmatrix} \mathbf{V}_2 a_2 \\ \mathbf{V}_2 b \end{bmatrix}, \quad \mathbf{a}_{k_L} = \mathbf{V}_1 - \mathbf{V}_2 a_2 \tag{B.11}$$

Using the expression for $\mathbf{a}_{k_L}$, the inequality can be further rewritten.

$$0 < \frac{\mathbf{R}''_{k_L} \cdot (\mathbf{V}_1 - \mathbf{V}_2 a_2) + 2a_2 - \left[\mathbf{R}'_{k_L} \cdot (\mathbf{V}_1 - \mathbf{V}_2 a_2) + 2\theta a_2\right] C_1}{\mathbf{R}_{k_L} \cdot (\mathbf{V}_1 - \mathbf{V}_2 a_2) + a_2\theta^2} + C_2 \tag{B.12}$$

Now expanding the brackets.

$$0 < \frac{R''_{k_L}\mathbf{V}_1 - \mathbf{R}''_{k_L}\mathbf{V}_2 a_2 + 2a_2 - \mathbf{R}'_{k_L}\mathbf{V}_1 \cdot C_1 + R'_{k_L}\mathbf{V}_2 a_2 \cdot C_1 + -2\theta a_2}{\mathbf{R}_{k_L}\mathbf{V}_1 - R_{k_L}\mathbf{V}_2 a_2 + a_2\theta^2} + C_2 \tag{B.13}$$

Now the parts non-dependant on $a_2$ are replaced with easier to handle symbols and are as defined in Equation B.15.

$$0 < \frac{C_A - C_B a_2 + 2a_2 - C_C + C_D a_2 - C_E a_2}{C_F - C_G a_2 + a_2\theta^2} + C_2 \tag{B.14}$$

$$\begin{cases} C_1 = \frac{\Phi'' - \sin\Phi\cos\Phi}{\Phi'^2 + \cos^2\Phi} & C_2 = \Phi'^2 + \cos^2\Phi \\ C_A = \mathbf{R}''_{k_L}\mathbf{V}_1 & C_B = \mathbf{R}''_{k_L}\mathbf{V}_2 \\ C_C = \mathbf{R}'_{k_L}\mathbf{V}_1 C_1 & C_D = \mathbf{R}'_{k_L}\mathbf{V}_2 C_2 \\ C_E = 2\theta C_1 & C_F = \mathbf{R}_{k_L}\mathbf{V}_1 \\ C_G = \mathbf{R}_{k_L}\mathbf{V}_2 \end{cases} \tag{B.15}$$

The last main part is to extract $a_2$ and bring it to one side of the inequality sign. Moving over $C_2$ and the denominator results in the following. Here the second inequality switch can take place, if the denominator is positive the sign remains, if negative the sign switches direction.

$$-C_2\left(C_F - C_G a_2 + a_2\theta^2\right) < C_A - C_B a_2 + 2a_2 - C_C + C_D a_2 - C_E a_2 \tag{B.16}$$

Now the brackets are expanded and all terms with $a_2$ are brought to the left hand side and others to the right hand side.

$$C_2 C_G a_2 - C_2 a_2\theta^2 + C_B a_2 - 2a_2 - C_D a_2 + C_E a_2 < C_A - C_C - C_2 C_F \tag{B.17}$$

The variable $a_2$ is separated from the terms.

$$a_2\left(C_2 C_G - C_2\theta^2 + C_B - 2 - C_D + C_E\right) < C_A - C_C - C_2 C_F \tag{B.18}$$

Now all non-$a_2$ terms are brought to the right hand side. Here the last possible sign switch occurs. If the denominator is positive the sign remains, if negative the sign switches direction. But now the final inequality has been found.

$$a_2 < \frac{C_A - C_C - C_2 C_F}{C_2 C_G - C_2\theta^2 + C_B - 2 - C_D + C_E} \tag{B.19}$$

$$\begin{cases} \mathbf{R}_k &= [1 \quad \theta \quad \theta^2 \quad \cos\theta \quad \theta\cos\theta \quad \sin\theta \quad \theta\sin\theta \quad ]^T \\ \mathbf{R}'_k &= [0 \quad 1 \quad 2\theta \quad -\sin\theta \quad \cos\theta - \theta\sin\theta \quad \cos\theta \quad \sin\theta + \theta\cos\theta \quad ]^T \\ \mathbf{R}''_k &= [0 \quad 0 \quad 2 \quad -\cos\theta \quad -2\sin\theta - \theta\cos\theta \quad -\sin\theta \quad 2\cos\theta - \theta\sin\theta \quad ]^T \end{cases} \tag{B.20}$$

$$\begin{cases} \mathbf{R}_{k_L} &= [1 \quad \theta \quad \cos\theta \quad \theta\cos\theta \quad \sin\theta \quad \theta\sin\theta \quad ]^T \\ \mathbf{R}'_{k_L} &= [0 \quad 1 \quad -\sin\theta \quad \cos\theta - \theta\sin\theta \quad \cos\theta \quad \sin\theta + \theta\cos\theta \quad ]^T \\ \mathbf{R}''_{k_L} &= [0 \quad 0 \quad -\cos\theta \quad -2\sin\theta - \theta\cos\theta \quad -\sin\theta \quad 2\cos\theta - \theta\sin\theta \quad ]^T \end{cases} \tag{B.21}$$

$$\begin{cases} \mathbf{\Phi}_k &= [\quad \cos\theta \quad \theta\cos\theta \quad \sin\theta \quad \theta\sin\theta \quad ]^T \\ \mathbf{\Phi}'_k &= [\quad -\sin\theta \quad \cos\theta - \theta\sin\theta \quad \cos\theta \quad \sin\theta + \theta\cos\theta \quad ]^T \\ \mathbf{\Phi}''_k &= [\quad -\cos\theta \quad -2\sin\theta - \theta\cos\theta \quad -\sin\theta \quad 2\cos\theta - \theta\sin\theta \quad ]^T \end{cases} \tag{B.22}$$

$$[\mathbf{A}] = \begin{bmatrix} R_0(\theta_i) & R_1(\theta_i) & R_3(\theta_i) & \dots & \dots & R_6(\theta_i) & 0 & 0 & 0 & 0 \\ R_0(\theta_f) & R_1(\theta_f) & R_3(\theta_f) & \dots & \dots & R_6(\theta_f) & 0 & 0 & 0 & 0 \\ R_0'(\theta_i) & R_1'(\theta_i) & R_3'(\theta_i) & \dots & \dots & R_6'(\theta_i) & 0 & 0 & 0 & 0 \\ R_0'(\theta_f) & R_1'(\theta_f) & R_3'(\theta_f) & \dots & \dots & R_6'(\theta_f) & 0 & 0 & 0 & 0 \\ -R_i^2 R_0''(\theta_i) & -R_i^2 R_1''(\theta_i) & -R_i^2 R_3''(\theta_i) & \dots & \dots & -R_i^2 R_6''(\theta_i) & \alpha_i \Phi_0''(\theta_i) & \dots & \dots & \alpha_i \Phi_3''(\theta_i) \\ -R_f^2 R_0''(\theta_f) & -R_f^2 R_1''(\theta_f) & -R_f^2 R_3''(\theta_f) & \dots & \dots & -R_f^2 R_6''(\theta_f) & \alpha_f \Phi_0''(\theta_f) & \dots & \dots & \alpha_f \Phi_3''(\theta_f) \\ 0 & 0 & 0 & 0 & 0 & 0 & \Phi_0(\theta_i) & \dots & \dots & \Phi_3(\theta_i) \\ 0 & 0 & 0 & 0 & 0 & 0 & \Phi_0(\theta_f) & \dots & \dots & \Phi_3(\theta_f) \\ 0 & 0 & 0 & 0 & 0 & 0 & \Phi_0'(\theta_i) & \dots & \dots & \Phi_3'(\theta_i) \\ 0 & 0 & 0 & 0 & 0 & 0 & \Phi_0'(\theta_f) & \dots & \dots & \Phi_3'(\theta_f) \end{bmatrix} \tag{B.23}$$

$$\mathbf{A}_{a_2} = \begin{bmatrix} a_2 R_2(\theta_i) & a_2 R_2(\theta_f) & a_2 R_2'(\theta_i) & a_2 R_2'(\theta_f) & a_2 R_2^2 R_2''(\theta_i) & a_2 R_2^2 R_2''(\theta_f) & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{B.24}$$

$$\mathbf{B} = \begin{bmatrix} \frac{1}{R_i} & \frac{1}{R_f} & -\frac{\tilde{v}_{r_i}}{R_i^2} & -\frac{\tilde{v}_{r_f}}{R_f} & C_i - 2\frac{\tilde{v}_{r_i}^2}{R_i} & C_f - 2\frac{\tilde{v}_{r_f}}{R_f} & \Phi_i & \Phi_f & \frac{\tilde{v}_{\varphi_i}}{R_i} & \frac{\tilde{v}_{\varphi_f}}{R_f} \end{bmatrix}^T \tag{B.25}$$

# C

# Appendix C - Earth-Tempel-1 Transfer with Coasting: Inputs and Outputs

This chapter contains the inputs for the optimisations in Chapter 7 (Section C.1) and the enlarged figures for the coasting vs non-coasting transfer comparison as shown in Section 7.8 (Section C.2).

## C.1. Optimisation Inputs

In this section, Table C.1, C.2, C.3, C.4, C.5, C.6, C.7 and C.8 contain the inputs ranges for the optimisations in Chapter 7. The input indices (IID) are linked with the run numbers per optimisation as given in Table 7.3, 7.4, 7.5, 7.6 and 7.7.

Table C.1: Input definition and ranges used for the optimisation of the Earth-Tempel-1 transfer including coasting, Inputs 1-5. LB = Lower Bound, UB = Upper Bound.

| **IID** | **Input 1** | | **Input 2** | | **Input 3** | | **Input 4** | | **Input 5** | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Inputs** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **Units** |
| Start date | 7300 | 9100 | 8000 | 9000 | 7300 | 9100 | 7300 | 9100 | 7300 | 9100 | MJD2000 |
| TOF1 | 200 | 1000 | 400 | 850 | 400 | 1000 | 400 | 1000 | 400 | 1000 | days |
| TOF2 | 200 | 1000 | 200 | 1000 | 100 | 1000 | 100 | 1000 | 100 | 1000 | days |
| TOF3 | 200 | 1000 | 600 | 1000 | 600 | 1000 | 400 | 1000 | 400 | 1000 | days |
| Radius | 0.9 | 5 | 1 | 2 | 1 | 2.5 | 1 | 2.5 | 1 | 2.2 | AU |
| Azimuth | 0 | 360 | 0 | 360 | 0 | 360 | 0 | 360 | 0 | 360 | deg |
| Elevation | -28.65 | 28.65 | -11.46 | 17.19 | -14.32 | 5.73 | -2.86 | 2.86 | -2.29 | 2.29 | deg |
| $V_r$ | -30.00 | 30.00 | -15.00 | 5.00 | -15.00 | 5.00 | -15.00 | 5.00 | -12.00 | 5.00 | km/s |
| $V_\theta$ | 5.00 | 40.00 | 17.50 | 27.50 | 15.00 | 35.00 | 15.00 | 35.00 | 15.00 | 35.00 | km/s |
| $V_\varphi$ | -20.00 | 20.00 | -7.50 | 5.00 | -12.50 | 5.00 | -2.00 | 2.00 | -1.00 | 1.00 | km/s |

Table C.2: Input definition and ranges used for the optimisation of the Earth-Tempel-1 transfer including coasting, Inputs 6-10. LB = Lower Bound, UB = Upper Bound.

| **IID** | **Input 6** | | **Input 7** | | **Input 8** | | **Input 9** | | **Input 10** | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Inputs** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **Units** |
| Start date | 8000 | 8250 | 8400 | 8650 | 8750 | 9000 | 8000 | 8250 | 8400 | 8650 | MJD2000 |
| TOF1 | 400 | 1000 | 400 | 1000 | 400 | 1000 | 650 | 900 | 500 | 750 | days |
| TOF2 | 100 | 1000 | 100 | 1000 | 100 | 1000 | 200 | 600 | 150 | 500 | days |
| TOF3 | 400 | 1000 | 400 | 1000 | 400 | 1000 | 700 | 1100 | 650 | 950 | days |
| Radius | 1 | 2.2 | 1 | 2.2 | 1 | 2.2 | 1.4 | 2.4 | 1.2 | 2 | AU |
| Azimuth | 0 | 360 | 0 | 360 | 0 | 360 | 100 | 200 | 75 | 200 | deg |
| Elevation | -2.29 | 2.29 | -2.29 | 2.29 | -2.29 | 2.29 | -1.15 | 1.15 | -1.15 | 1.15 | deg |
| $V_r$ | -12.00 | 5.00 | -12.00 | 5.00 | -12.00 | 5.00 | -12.00 | -4.00 | -10.00 | -4.00 | km/s |
| $V_\theta$ | 15.00 | 35.00 | 15.00 | 35.00 | 15.00 | 35.00 | 16.50 | 28.00 | 17.50 | 30.00 | km/s |
| $V_\varphi$ | -1.00 | 1.00 | -1.00 | 1.00 | -1.00 | 1.00 | -0.20 | 0.20 | -0.20 | 0.20 | km/s |

Table C.3: Input definition and ranges used for the optimisation of the Earth-Tempel-1 transfer including coasting, Inputs 11-15. LB = Lower Bound, UB = Upper Bound.

| **IID** | **Input 11** | | **Input 12** | | **Input 13** | | **Input 14** | | **Input 15** | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Inputs** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **Units** |
| Start date | 8850 | 8950 | 7300 | 9100 | 7300 | 7500 | 7650 | 8000 | 8000 | 8300 | MJD2000 |
| TOF1 | 300 | 600 | 200 | 1000 | 300 | 1000 | 300 | 1000 | 300 | 1000 | days |
| TOF2 | 50 | 250 | 100 | 1000 | 100 | 1000 | 100 | 1000 | 100 | 1000 | days |
| TOF3 | 650 | 950 | 200 | 1000 | 400 | 1000 | 400 | 1000 | 400 | 1000 | days |
| Radius | 1 | 1.5 | 1 | 5 | 1 | 2.5 | 1 | 2.5 | 1 | 2.5 | AU |
| Azimuth | 200 | 275 | 0 | 360 | 0 | 360 | 0 | 360 | 0 | 360 | deg |
| Elevation | -0.57 | 0.57 | -11.46 | 17.19 | -11.46 | 17.19 | -11.46 | 17.19 | -11.46 | 17.19 | deg |
| $V_r$ | -4.00 | 4.00 | -15.00 | 5.00 | -15.00 | 5.00 | -15.00 | 5.00 | -15.00 | 5.00 | km/s |
| $V_\theta$ | 26.00 | 34.00 | 17.50 | 27.50 | 17.50 | 27.50 | 17.50 | 27.50 | 17.50 | 27.50 | km/s |
| $V_\varphi$ | -0.20 | 0.35 | -7.50 | 5.00 | -7.50 | 5.00 | -7.50 | 5.00 | -7.50 | 5.00 | km/s |

Table C.4: Input definition and ranges used for the optimisation of the Earth-Tempel-1 transfer including coasting, Inputs 16-20. LB = Lower Bound, UB = Upper Bound.

| **IID** | **Input 16** | | **Input 17** | | **Input 18** | | **Input 19** | | **Input 20** | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Inputs** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **Units** |
| Start date | 8400 | 8700 | 8700 | 9000 | 7300 | 7420 | 7650 | 7900 | 8125 | 8175 | MJD2000 |
| TOF1 | 300 | 1000 | 300 | 1000 | 300 | 650 | 600 | 1100 | 450 | 850 | days |
| TOF2 | 100 | 1000 | 100 | 1000 | 25 | 300 | 400 | 1100 | 250 | 750 | days |
| TOF3 | 400 | 1000 | 400 | 1000 | 700 | 1100 | 700 | 1100 | 600 | 1000 | days |
| Radius | 1 | 2.5 | 1 | 2.5 | 1.2 | 1.8 | 1.2 | 2.4 | 1.8 | 2.8 | AU |
| Azimuth | 0 | 360 | 0 | 360 | 25 | 175 | 75 | 225 | 100 | 175 | deg |
| Elevation | -11.46 | 17.19 | -11.46 | 17.19 | -2.86 | 6.88 | -2.86 | 11.46 | -2.86 | 8.59 | deg |
| $V_r$ | -15.00 | 5.00 | -15.00 | 5.00 | -7.50 | 0.00 | -12.50 | 0.00 | -8.00 | -2.00 | km/s |
| $V_\theta$ | 17.50 | 27.50 | 17.50 | 27.50 | 19.00 | 27.50 | 17.50 | 27.50 | 13.00 | 22.00 | km/s |
| $V_\varphi$ | -7.50 | 5.00 | -7.50 | 5.00 | -0.20 | 2.20 | -3.00 | 3.00 | -0.50 | 2.00 | km/s |

Table C.5: Input definition and ranges used for the optimisation of the Earth-Tempel-1 transfer including coasting, Inputs 21-25. LB = Lower Bound, UB = Upper Bound.

| **IID** | **Input 21** | | **Input 22** | | **Input 23** | | **Input 24** | | **Input 25** | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Inputs** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **Units** |
| Start date | 8450 | 8550 | 8725 | 8900 | 7300 | 7600 | 7600 | 8000 | 7350 | 7550 | MJD2000 |
| TOF1 | 350 | 750 | 350 | 600 | 400 | 1000 | 400 | 1000 | 500 | 1000 | days |
| TOF2 | 100 | 750 | 600 | 900 | 100 | 1000 | 100 | 1000 | 50 | 500 | days |
| TOF3 | 500 | 1000 | 1000 | 1300 | 400 | 1000 | 400 | 1000 | 800 | 1200 | days |
| Radius | 1.3 | 2.2 | 1.2 | 1.8 | 1 | 2.5 | 1 | 2.5 | 1 | 2 | AU |
| Azimuth | 50 | 200 | 50 | 175 | 0 | 360 | 0 | 360 | 50 | 300 | deg |
| Elevation | 0 | 8.59 | -2.86 | 5.73 | -5.73 | 5.73 | -5.73 | 5.73 | -5.73 | 5.73 | deg |
| $V_r$ | -8.00 | -2.00 | -7.50 | 0.00 | -15.00 | 5.00 | -15.00 | 5.00 | -15.00 | 5.00 | km/s |
| $V_\theta$ | 16.50 | 27.50 | 18.00 | 28.00 | 15.00 | 35.00 | 15.00 | 35.00 | 20.00 | 40.00 | km/s |
| $V_\varphi$ | -2.00 | 2.20 | -1.00 | 2.20 | -4.00 | 4.00 | -4.00 | 4.00 | -4.00 | 4.00 | km/s |

Table C.6: Input definition and ranges used for the optimisation of the Earth-Tempel-1 transfer including coasting, Inputs 26-30. LB = Lower Bound, UB = Upper Bound.

| **IID** | **Input 26** | | **Input 27** | | **Input 28** | | **Input 29** | | **Input 30** | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Inputs** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **LB** | **UB** | **Units** |
| Start date | 7600 | 8000 | 7600 | 8000 | 7420 | 7465 | 7700 | 7850 | 7700 | 7875 | MJD2000 |
| TOF1 | 400 | 1000 | 400 | 1000 | 660 | 820 | 850 | 1100 | 450 | 800 | days |
| TOF2 | 100 | 550 | 550 | 1100 | 50 | 450 | 400 | 600 | 900 | 1200 | days |
| TOF3 | 500 | 1100 | 500 | 1100 | 850 | 1200 | 900 | 1200 | 600 | 950 | days |
| Radius | 1 | 2.5 | 1 | 2.5 | 1.05 | 1.3 | 1.4 | 2.5 | 1 | 1.75 | AU |
| Azimuth | 0 | 360 | 0 | 360 | 200 | 250 | 100 | 200 | 100 | 250 | deg |
| Elevation | -5.73 | 3.44 | -5.73 | 3.44 | -3.44 | 1.72 | -2.29 | 2.29 | -1.72 | 1.72 | deg |
| $V_r$ | -15.00 | 5.00 | -15.00 | 5.00 | -6.00 | 1.00 | -12.50 | -5.00 | -10.00 | 0.00 | km/s |
| $V_\theta$ | 15.00 | 35.00 | 15.00 | 35.00 | 32.00 | 37.00 | 15.00 | 27.50 | 20.00 | 35.00 | km/s |
| $V_\varphi$ | -4.00 | 2.00 | -4.00 | 2.00 | -2.50 | 1.50 | -0.50 | 0.50 | -0.50 | 0.50 | km/s |

Table C.7: Input definition and ranges used for the optimisation of the Earth-Tempel-1 transfer including coasting, Inputs 31 and 37. LB = Lower Bound, UB = Upper Bound.

| **IID** | **Input 31** | | **Input 37** | | |
|---|---|---|---|---|---|
| **Inputs** | **LB** | **UB** | **LB** | **UB** | **Units** |
| Start date | 7300 | 9100 | 7300 | 9100 | MJD2000 |
| TOF1 | 100 | 1000 | 100 | 1000 | days |
| TOF2 | 100 | 1000 | 10 | 1000 | days |
| TOF3 | 100 | 1000 | 100 | 1000 | days |
| Radius | 0.9 | 5 | 0.9 | 5 | AU |
| Azimuth | 0 | 360 | 0 | 360 | deg |
| Elevation | -28.65 | 28.65 | -28.65 | 28.65 | deg |
| $V_r$ | -30.00 | 30.00 | -30.00 | 30.00 | km/s |
| $V_\theta$ | 5.00 | 40.00 | 5.00 | 40.00 | km/s |
| $V_\varphi$ | -20.00 | 20.00 | -20.00 | 20.00 | km/s |

Table C.8: Input definition and ranges used for the optimisation of the Earth-Tempel-1 transfer including coasting, Inputs 31-37. LB = Lower Bound, UB = Upper Bound.

| | | **Start Date [MJD2000]** | |
|---|---|---|---|
| **IID** | **Run Number** | **LB** | **UB** |
| 31 | runMO1-6 | 7300 | 9100 |
| 32 | runMO7,10,16,21-23,26 | 7300 | 7600 |
| 33 | runMO8,11,17,27 | 7600 | 8000 |
| 34 | runMO9,12,18,28 | 8000 | 8400 |
| 35 | runMO13,19,29 | 8400 | 8800 |
| 36 | runMO14,20,30 | 8800 | 9100 |
| 37 | MO24-25 | 7300 | 9100 |

## C.2. NON-COASTING VS COASTING FIGURES

In this section, Figures C.1, C.2 and C.3 are presented which are the same figures as presented in Section 7.8, but now in a larger scale for better viewing.

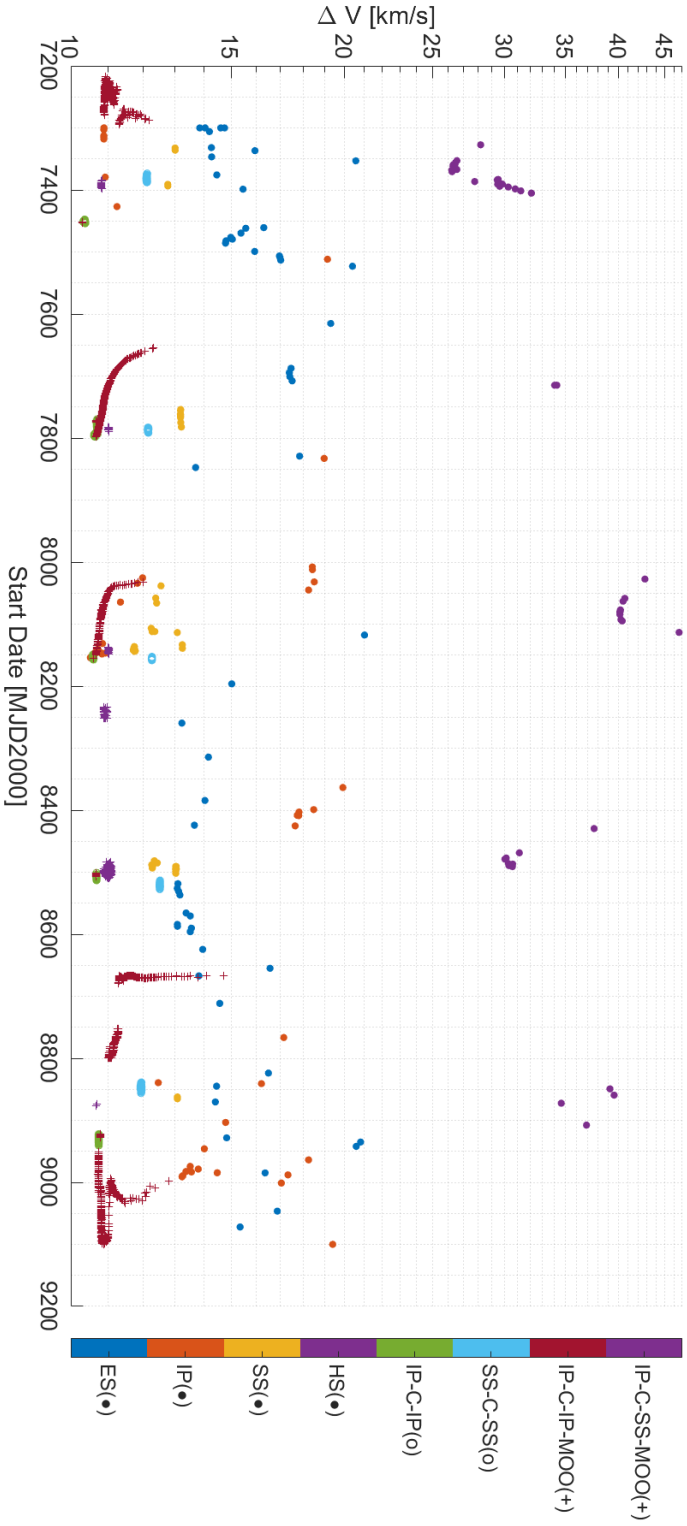



Figure C.1: Start date vs $\Delta V$ plot combining all Earth-Tempel-1 non-coasting trajectories found with optimisation in Table 6.2 and all coasting transfers of analysis group A-E, of the final generation of the optimisation.

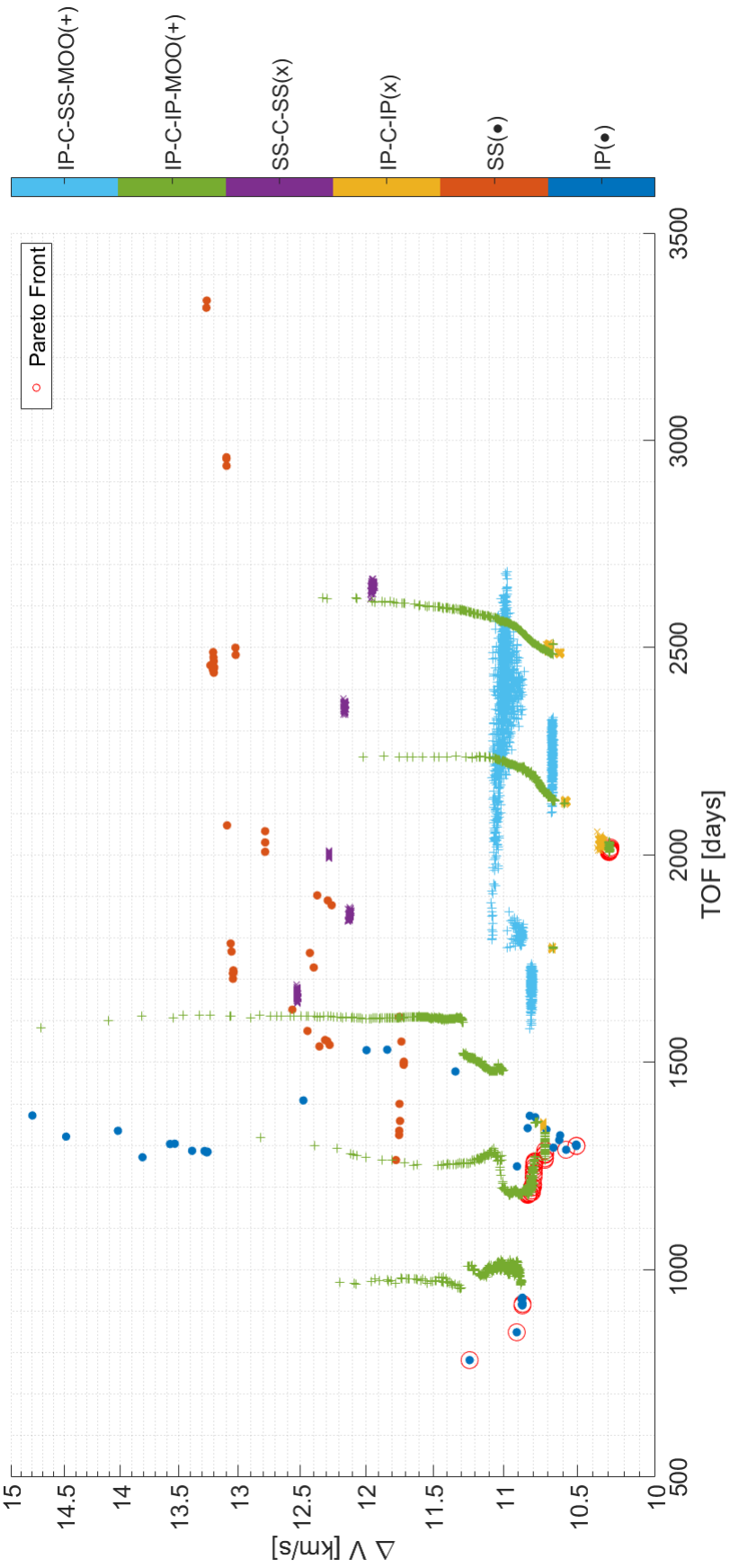



Figure C.2: TOF vs $\Delta V$ plot combining all Earth-Tempel-1 non-coasting trajectories with IP and SS found with optimisation in Table 6.2 and all coasting transfers of analysis group A-E, of the final generation of the optimisation.

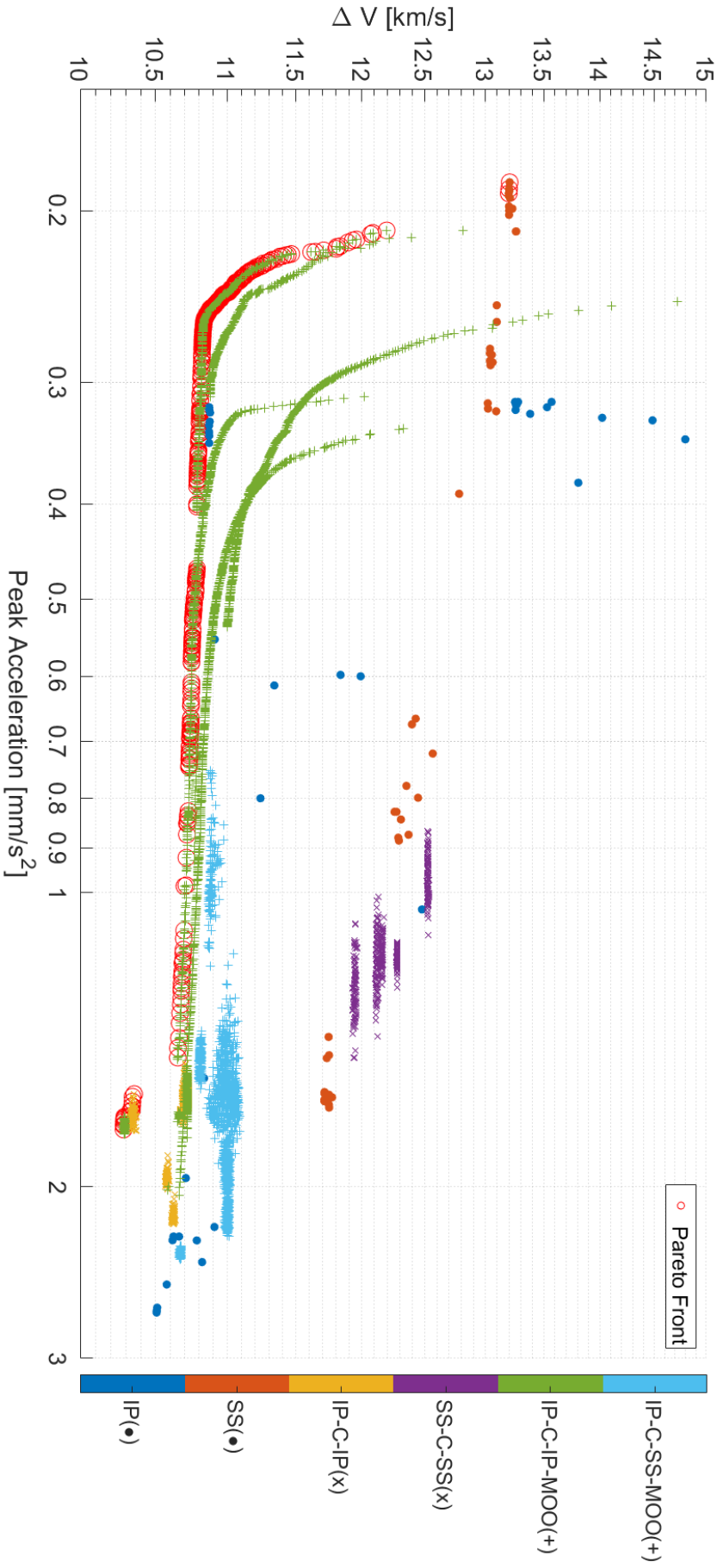



Figure C.3: Peak thrust acceleration vs $\Delta V$ plot combining all Earth-Tempel-1 non-coasting trajectories with IP and SS found with optimisation in Table 6.2 and all coasting transfers of analysis group A-E, of the final generation of the optimisation.