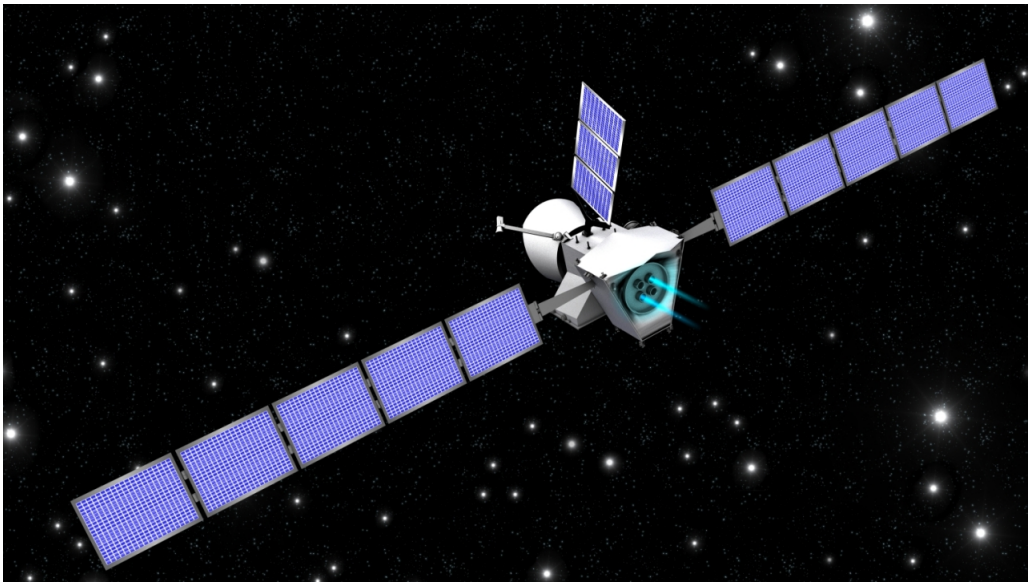
A Hodographic-Shaping Method for Low-Thrust Trajectory Design



MSc Thesis

D.J. Gondelach

July 5, 2012

TU Delft Delft University of Technology

[1]The figure on the title page is an artist's impression of BepiColombo's Mercury Composite Spacecraft during the interplanetary cruise to Mercury, found on `bepicolombo.esa.int`

To my parents

Acknowledgments

This thesis concludes my Master's degree program at the faculty of Aerospace Engineering of the Delft University of Technology. After seven years of studying and enjoying my student life, I look back on my time in Delft with pleasure and satisfaction. Fortunately, I did not experience my high and low points on my own, but shared them with many others, and therefore I'd like to thank a few people.

First of all, I want to thank Ron Noomen for being my motivational supervisor during my master's research. In our weekly meetings he always gave excellent feedback and his enthusiasm and useful tips motivated me throughout my research. Furthermore, my fellow students on the ninth floor (especially in the room 9.03) were very helpful in case of coding or writing issues, but especially made my time on the ninth floor very enjoyable.

Of course, I'd also like to thank all my friends, from Åsene, my student society DSB, Aerospace, my student house in the Zairestraat, my high school Niftarlake, my hometown Vleuten and my internship at the Bodensee; you made my student life a true delight.

Moreover, I want to thank my family, especially my dear sister Sarah who has been a great example for me and her husband Wouter.

Finally, there are two persons I'd like to thank most, and therefore this report is dedicated to them, to my parents. Dear mom and dad, thank you for all the love and support you have given me, I cannot imagine a live without you.

David Gondelach

Summary

The goal of this research was to develop an analytic low-thrust trajectory design method. The term *low thrust* stems form the fact that current electric propulsion systems can only produce low amounts of thrust. Nevertheless, these systems can deliver very high amounts of ΔV using little propellant, which makes them very suitable for interplanetary flight.

The orbital mechanics regarding low-thrust interplanetary flight were considered and it was found that for first-order design it is acceptable to neglect all perturbing forces with respect to the Sun's gravitational attraction and the (low) thrust force. In addition to that, a look was taken at the characteristics of electric propulsion and feasible thrust (acceleration) levels were obtained.

Since only low thrust levels can be achieved by electric propulsion, long periods of thrusting are required to obtain sufficient change in velocity. This continuous-thrusting feature of electrically-propelled spacecraft highly complicates the dynamics of such spacecraft. As a consequence, the design and optimization of low-thrust trajectories is very difficult, and therefore preliminary design of such trajectories is very important. Such first-order designs can give a fast overview of all feasible trajectories and can, moreover, be used as an initial guess for more refined trajectory optimization. For generating and optimizing first-order low-thrust trajectories, analytical methods are very suited, since they are very fast and exact. Several analytical low-thrust methods have been developed recently and most of them are *shape-based*; the trajectory is assumed to have a specific shape which satisfies the boundary conditions and subsequently the thrust profile can be obtained from the dynamics. These shape-based methods are popular for preliminary design, because of their simplicity and ability to obtain feasible trajectories. The currently known shape-based methods were investigated to obtain know-how about their functioning and to find their strengths and weaknesses. It was noticed that regarding the solving of boundary conditions improvements could be made and that most shaping methods lack flexibility.

Inspired by the use *velocity hodographs* for the computation of non-perturbed transfer orbits, a novel low-thrust trajectory design method was developed based on shaping the velocity of spacecraft during the transfer. For the shaping of the velocity components, velocity functions were used which consist of a sum of simple base functions. These base functions can be integrated analytically, such that the change in position can be obtained analytically. Since the velocity is shaped, the conditions on initial and final velocity can be solved very easily and exactly. In addition, the boundary conditions on position can be solved exactly without the need of iterative computations. Next to the parameters required to satisfy the boundary conditions, extra parameters can be added to make the design and optimization of trajectories more flexible. Two different methods have been developed; one which shapes the velocity as a function of time and another one which shapes as a function of the polar angle.

For the computation of the required ΔV, and for the computation of the polar angle in the time-driven method, numerical integration is required. Four simple integration methods were tested and an RK4 integrator was found to perform best based on accuracy and computation speed. Since the derivative of the ΔV and the polar angle (the thrust acceleration and angular velocity, respectively) can be computed as a function of time or polar angle, their values can be computed prior to the integration. As a result, RK4 integration simplifies to Simpson's rule and the integration can be done much faster.

Both hodographic-shaping methods have been verified by numerically propagating the initial spacecraft state vector using thrust profiles found by the shaping methods. Both the time-driven and polar-angle-driven method were found to function well, and the obtained trajectories and thrust profiles agreed almost exactly.

In order to obtain minimum-ΔV trajectories, the free parameters in the velocity functions were optimized. Both a Nelder-Mead (NM) and Differential Evolution (DE) optimizer were tested for this purpose. NM was found to be equally robust, but much faster than DE, and therefore NM was used. In addition, initial guesses for the free parameters were used to speed up the optimization. Finally, the search for the optimal departure date and TOF was done by stepping through the flight window using a grid.

Both hodographic-shaping methods were tested for missions to Mars, the near-Earth asteroid 1989ML, comet Tempel 1 and Mercury. For missions to Mars and 1989ML the two methods gave comparable results. The time-driven method found better trajectories to Tempel 1 (a far outer target), whereas the polar-angle-driven method found better ones to Mercury (an inner target). In comparison with the spherical and pseudo-equinoctial shaping methods and DITAN (the benchmark), the hodographic-shaping methods performed well, especially for mission to Mars and to asteroid 1989ML.

The near-optimal results came at the cost of computational speed. The best results were found using 6 free parameters requiring require a computation time of 2 s per trajectory on average. The lowest-order solutions (0 DoF), on the other hand, required only 1.6 ms per trajectory on average. Finally, the lowest-order solutions were found to be unable to obtain near-optimal trajectories, however, they were able to indicate the correct regions of low ΔV in the flight windows and to give a good indication of the required ΔV and thrust acceleration.

Contents

List of Symbols and Abbreviations

Symbols

a	semi-major axis, m
a	acceleration, m/s^2
a	inverse-polynomial coefficient for 2D shape-based method
a_i	coefficient in shaping function R (spherical shaping)
a_i	coefficient in power series expansion for r (pseudo-spectral shaping)
a_z	coefficient in shaping function for z-direction (inverse polynomial)
b	inverse-polynomial coefficient
b_i	coefficient in shaping function Φ (spherical shaping)
b_i	coefficient in power series expansion for θ (pseudo-spectral shaping)
b_z	coefficient in shaping function for z-direction (inverse polynomial)
c	speed of light, 2.997925×10^8 m/s [Cornelisse et al., 1979]
c	inverse-polynomial coefficient
c_i	coefficient in velocity function (hodographic shaping)
c_i	coefficient in power series expansion for z (pseudo-spectral shaping)
c_z	coefficient in shaping function for z-direction (inverse polynomial)
C	integration constant
C_R	reflectivity
C_R	constant (Differential Evolution)
d	inverse-polynomial coefficient
d_z	coefficient in shaping function for z-direction (inverse polynomial)
e	eccentricity
e	inverse-polynomial coefficient
$\mathbf{e}$	unit vector
E	power density, W/m^2
f	modified equinoctial element
f	perturbing acceleration, m/s^2
f	thrust acceleration, m/s^2
f	inverse-polynomial coefficient
$\mathbf{f}$	thrust acceleration vector, m/s^2
$\mathbf{f}$	function for state derivative
F	scaling factor (Differential Evolution)
g	modified equinoctial element
g	inverse-polynomial coefficient
g_0	standard gravity acceleration, m/s^2
G	universal gravitational constant, 6.670×10^{-11} m^{3}kg^{-1}s^{-2} [Cornelisse et al., 1979]
h	modified equinoctial element
h	size of time-step, s
h_θ	size of polar-angle-step, rad
H	angular momentum per unit mass, m^2/s
i	inclination, rad
$\mathbf{i}$	unit vector in Cartesian x-direction

I_{sp}	specific impulse, s
$\mathbf{j}$	unit vector in Cartesian y-direction
J	objective function (value)
k	modified equinoctial element
k	truncation order of power series expansion for r (pseudo-spectral shaping)
$\mathbf{k}$	unit vector in Cartesian z-direction
k_0	scaling factor (exponential sinusoid)
k_1	dynamic range parameter (exponential sinusoid)
k_2	winding parameter (exponential sinusoid)
$\mathbf{k}_i$	slope, state derivative (RK4)
K_i	coefficient resulting from matrix-vector multiplication (hodographic shaping)
l	truncation order of power series expansion for θ (pseudo-spectral shaping)
L	true longitude, rad
m	mass, kg
m	order of base function (hodographic shaping)
m	truncation order of power series expansion for z (pseudo-spectral shaping)
M	mass, kg
M	mean anomaly, rad
$\dot{M}$	mass flow, kg/s
M_0	mean anomaly at epoch, rad
M_0	initial mass, kg
M_e	final mass, kg
n	number of dimensions
n	order of base function (hodographic shaping)
N	number of revolutions
N	number of solution candidates
p	semi-latus rectum, m
P	total power emitted by the Sun, W
P	penalty factor
$\mathbf{P}$	point
P_0	initial position
P_f	final position
q	exponent of highest-order term for z-direction (inverse polynomial)
r	radial distance, m
r	distance from the Sun, m
$\mathbf{r}$	position vector, m
R	shaping function for spherical coordinate r (spherical shaping)
R	shaping function for radial velocity r' (hodographic shaping)
R^j	random number (Differential Evolution)
s	distance from the Sun, m
$S^\star$	effective cross-sectional area, m^2
t	time, s
t_f	final time, s
T	thrust, N
T	orbital period, s
T	shaping function for time t (spherical shaping)

T	shaping function for time evolution t' (hodographic shaping)
T_{syn}	synodic period, s
$\mathbf{T}$	thrust vector, N
TOF	time of flight, s
u	argument of latitude, rad
$\mathbf{v}$	velocity
v_i	velocity base function (hodographic shaping)
V	velocity, m/s
$\mathbf{V}$	velocity vector, m/s
$\mathbf{V}_\infty$	hyperbolic excess velocity vector, m/s
V_0	initial velocity
V_c	local circular velocity, m/s
V_e	exhaust velocity, m/s
V_f	final velocity
W	power density of incoming radiation, W/m^2
W	input power / available power, W
x	Cartesian coordinate
x	solution candidate
$\mathbf{x}_i$	solution candidate
$\hat{\mathbf{x}}_i$	mutant vector
X	reference-frame axis
y	Cartesian coordinate
$\mathbf{y}$	state vector
$\dot{\mathbf{y}}$	state derivative
$\mathbf{y}_i$	trail vector
Y	reference-frame axis
z	Cartesian coordinate
z	axial distance, m
Z	shaping function for axial velocity z' (hodographic shaping)
Z	reference-frame axis
α	in-plane thrust angle, rad
$\boldsymbol{\alpha}$	vector of equinoctial elements
$\tilde{\boldsymbol{\alpha}}$	vector of pseudo-equinoctial elements
γ	flight path angle, rad
Δt	elapsed time, s
ΔV	velocity change, m/s
η	approximate solution
θ	polar angle (polar, spherical, cylindrical coordinate), rad
θ	true anomaly, rad
$\dot{\theta}$	angular velocity, rad/s
θ_f	final polar angle, rad
λ	constant (Nelder-Mead)
$\boldsymbol{\lambda}$	shaping parameter vector (pseudo-equinoctial shaping)
λ_i	shaping parameter (pseudo-equinoctial shaping)
μ	gravitational parameter, m^3/s^2
μ	gravitational parameter of the Sun, 1.3271244×10^{20} m^3/s^2 [Wakker, 2007]

τ	time of pericenter passage
φ	latitude (spherical coordinates), rad
φ	phase parameter, rad (pseudo-equinoctial shaping)
ϕ	phase angle, rad (exponential sinusoid)
Φ	shaping function for spherical coordinate φ (spherical shaping)
Φ	increment function
ψ	transfer angle, rad
ω	argument of pericenter, rad
Ω	right ascension of ascending node, rad

Subscripts

0	initial value
a	apocenter
d	perturbing body d
f	final value
h	component along the angular momentum vector
i	body i
in	in-plane component
k	body k
max	maximum value
min	minimum value
o	out-of-plane component
p	pericenter
r	radial component, i.e. component along the radial vector
s	spacecraft
t	component along velocity vector
tot	total
x	component along Cartesian **i**-vector
y	component along Cartesian **j**-vector
z	axial component, i.e. component along Cartesian **k**-vector and cylindrical $\mathbf{e}_z$-vector
θ	normal component, i.e. component along polar, cylindrical and spherical $\mathbf{e}_\theta$-vector
φ	component along spherical $\mathbf{e}_\varphi$-vector

Superscripts

$\dot{\square}$	derivative w.r.t. time t
$\ddot{\square}$	second derivative w.r.t. time t
$\square'$	derivative w.r.t. polar angle θ
$\square''$	second derivative w.r.t. polar angle θ
$\tilde{\square}$	integral w.r.t. time t
$\hat{\square}$	integral w.r.t. polar angle θ

Abbreviations

AU	Astronomical Unit, 1.495979×10^{11} m [Cornelisse et al., 1979]
BC	Boundary Condition
CPU	Computation time
DITAN	Direct Interplanetary Trajectory Analysis
DoF	Degrees of Freedom
ESA	European Space Agency
GALLOP	Gravity-Assist Low-thrust Local Optimization Program
IMAGO	Interplanetary Mission Analysis Global Optimization
JAXA	Japan Aerospace Exploration Agency
MJD2000	Modified Julian Date 2000; time in days since the astronomical epoch J2000: January 1, 2000 at 12:00 TT
MUSES-C	Mu Space Engineering Spacecraft-C
NASA	National Aeronautics and Space Administration
NEP	Nuclear Electric Propulsion
RK4	fourth-order Runge-Kutta method
SEP	Solar Electric Propulsion
SMART-1	1^{st} of Small Missions for Advanced Research in Technology
STOUR-LTGA	Satellite Tour Design Program including Low-Thrust and Gravity Assists
TOF	Time Of Flight
Tudat	TU Delft Astrodynamics Toolbox

1. Introduction

One of the major challenges in space engineering is to bring increasingly amounts of useful mass to a distant destination. With the successful application of electric propulsion in the last decades, space traveling has obtained a new way to reduce propellant masses and therefore to increase payload masses [Kawaguchi et al., 2006] [Racca et al., 2002] [Rayman et al., 2006] [Rayman and Williams, 2002].

The application of electric propulsion is characterized by the low amount of thrust that can be delivered. As a consequence long periods of thrusting are required to obtain sufficient change in velocity, see Figure 1.1. On the other hand, since electric propulsion is very efficient, very high ΔV can be achieved using little propellant. Such high ΔV are especially of interest for interplanetary transfers, which are considered in this report.

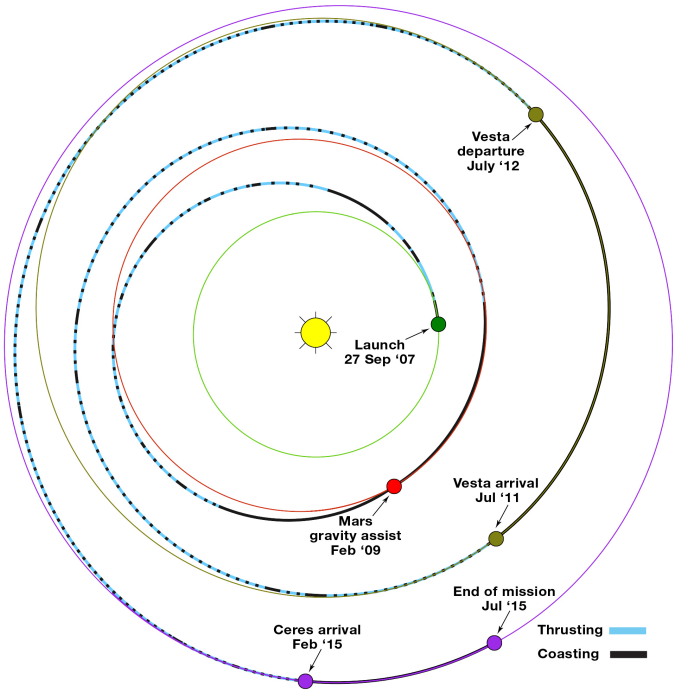


Figure 1.1: Dawn's interplanetary trajectory to Vesta and Ceres indicating thrusting and coasting periods [NASA - JPL, 2009].

However, the continuous-thrusting feature of electrically-propelled spacecraft highly complicates the dynamics of such spacecraft. The problem of dealing with these complex dynamics resulted in a whole new challenging field (of engineering) within astrodynamics, namely the design and optimization of low-thrust trajectories. Because of the complexity of low-thrust trajectory design and optimization, preliminary design of such trajectories is very important. Such first-order designs can namely give a fast overview of all feasible trajectories and can, moreover, be used as an initial guess for more refined trajectory optimization. Since analytical methods are usually very fast and exact, they are very suitable for generating and

optimizing first-order low-thrust trajectories. Several analytical low-thrust methods have been developed recently [Petropoulos and Longuski, 2004] [De Pascale and Vasile, 2006] [Gao, 2007] [De Vogeleer, 2008] [Wall and Conway, 2009] [Lantoine and Russell, 2009] [Novak and Vasile, 2011]. Most of these methods are shape-based; the trajectory is assumed to have a specific shape which satisfies the boundary conditions and subsequently the thrust profile can be obtained from the dynamics. These shape-based methods are popular for preliminary design, because of their simplicity and ability to obtain feasible trajectories. In this report a novel shape-based low-thrust trajectory design method is presented which can generate first-order and near-optimal trajectories.

This introduction forms the initial chapter of this report. In the second chapter a look is taken at the orbital mechanics related to low-thrust trajectory design and non-perturbed transfer orbits are briefly discussed.

After that low-thrust propulsion is treated in Chapter 3 to gain know-how of the characteristics of this technology and the related dynamics. In the next chapter existing shape-based methods for low-thrust trajectory design are discussed and compared.

In Chapter 5 the novel shaping method is presented and discussed extensively. The numerical integration which is required for the method is treated in the subsequent chapter. Chapter 7 deals with the verification of the novel method in order to find out whether the method functions properly and accurately.

In the two following chapters the optimization of the trajectories is considered. First, in Chapter 8 several suitable optimization method are discussed. After that, in Chapter 9 the problem of optimizing the trajectories is assessed and the corresponding optimization process is adjusted.

In Chapter 10 the test cases for examining the novel method are treated. The results of the novel method for these test cases are discussed in detail in Chapter 11. In addition, in this chapter the results are compared with results of other low-thrust trajectory design methods.

Finally, in the last two chapters conclusions are drawn and recommendations for further research are given.

2. Orbital Mechanics

The final goal of the preliminary design of low-thrust trajectories is to obtain a feasible and near-optimal one which fulfills all requirements. During the design process the real world is usually approximated using a simple model to speed up the process. However, in order to obtain feasible solutions it is very important to make proper approximations and assumptions.

In this chapter a basis for the models and assumptions for the novel low-thrust trajectory design method will be set. First a reference frame and several coordinate systems will be discussed. After that a closer look is taken at the mechanics of bodies in space, and the special case of the two-body problem and the related orbital elements are discussed. In addition, orbit perturbations are considered and an examination is done on the possibility to neglect them in the design process. Finally, the characteristics of non-perturbed transfer orbits are briefly discussed.

2.1. Reference Frames

Reference frames are required to measure and express the position and velocity of spacecraft and celestial bodies. Such a frame of reference is attached to a 'fixed' point with respect to which the position and velocity are determined.

Heliocentric reference frame Since interplanetary flights are considered, the motion of spacecraft and celestial bodies can most easily be described using the Sun as the origin of the reference frame. In such a heliocentric reference frame usually the ecliptic plane is taken as the XY-plane. Furthermore, the X-axis points in the direction of the vernal equinox (at a certain epoch) and the Z-axis points in the direction of the ecliptic north pole. However, one may also choose to use the initial orbital plane of the spacecraft as XY-plane, with the Z-axis pointing in the direction of the spacecraft's angular momentum vector and X-axis pointing to the initial position of the spacecraft. In that case positions are expressed with respect to the initial orbital plane and the initial position of the spacecraft.

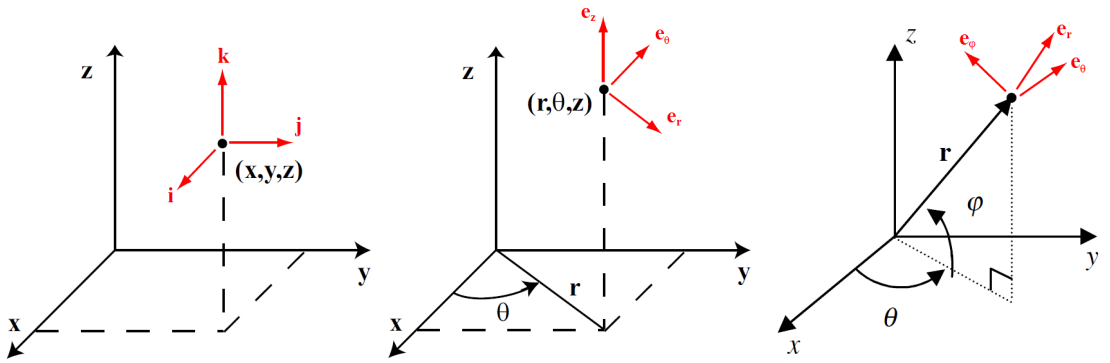
2.2. Coordinate Systems

As said, with respect to reference frames positions and motion can be described. The parameters which are used to describe these positions and motion are called coordinates, or, more general, coordinate systems. Several often-applied coordinate systems are discussed here.

2.2.1. Cartesian coordinates

The three axes of the three-dimensional Cartesian coordinate system, denoted as the x-, y- and z-axes, are linear and perpendicular to each other, see Figure 1(a). Furthermore, the coordinates x, y and z may lie anywhere in the interval $(-\infty, \infty)$.

In addition to positions, also velocities can be specified. The velocity components are directed along the x-, y- and z-axes and are indicated as $\dot{x}$, $\dot{y}$ and $\dot{z}$, respectively.



(a) Cartesian coordinate system (b) Cylindrical coordinate system (c) Spherical coordinate system

Figure 2.1: Three coordinate systems.

The base vectors for the Cartesian coordinate system are **i**, **j** and **k**, which are aligned with the x-, y- and z-axes, respectively, see Figure 1(a).

2.2.2. Cylindrical coordinates

In the two-dimensional case, a position can be specified using the polar coordinates r and θ. Here r is the radial distance from the origin and θ, called the polar angle, is the counterclockwise angle from the x-axis, see Figure 1(b). By superposing a height axis z, the two-dimensional polar coordinates can be expanded to obtain a cylindrical coordinate system, see Figure 1(b). In terms of x, y and z one can write the cylindrical coordinates as:

$$r = \sqrt{x^2 + y^2} \tag{2.1}$$

$$\theta = \arctan \frac{y}{x} \tag{2.2}$$

$$z = z \tag{2.3}$$

where $r \in [0, \infty)$, $\theta \in [0, 2\pi)$, $z \in (-\infty, \infty)$ and the signs of x and y determine in which quadrant θ lies. Vice versa, the Cartesian coordinates can be computed using:

$$x = r\cos\theta \tag{2.4}$$
$$y = r\sin\theta \tag{2.5}$$
$$z = z \tag{2.6}$$

The cylindrical base vectors are $\mathbf{e}_r$, $\mathbf{e}_\theta$ and $\mathbf{e}_z$, see Figure 1(b). These base vectors are not fixed, but are a function of position. The cylindrical velocities are V_r, V_θ and V_z are directed along the corresponding base vectors.

2.2.3. Spherical coordinates

Another way to specify the position of a body in three dimensions is by using spherical coordinates. Spherical coordinates consist of one radial and two angular coordinates. The radial coordinate r is the distance from the origin, and the angular coordinate θ is the azimuthal angle in the xy-plane measured from the x-axis where $\theta \in [0, 2\pi)$. The other angular coordinate φ is the latitude which is positive when z is positive and $\varphi \in [-\pi/2, \pi/2]$, see Figure 1(c). Instead of using the latitude φ, one can also use the colatitude ϕ, which is the complementary angle of the latitude, measured from the z-axis, i.e. $\phi = \pi/2 - \varphi$.

The spherical coordinates (r,θ,φ) are related to the Cartesian coordinates (x,y,z) as follows:

$$r = \sqrt{x^2 + y^2 + z^2} \tag{2.7}$$
$$\theta = \arctan\left(\frac{y}{x}\right) \tag{2.8}$$
$$\varphi = \arcsin\left(\frac{z}{r}\right) \tag{2.9}$$

Vice versa the Cartesian coordinates are expressed in spherical coordinates as:

$$x = r\cos\theta\cos\varphi \tag{2.10}$$
$$y = r\sin\theta\cos\varphi \tag{2.11}$$
$$z = r\sin\varphi \tag{2.12}$$

The spherical base vectors are $\mathbf{e}_r$, $\mathbf{e}_\theta$ and $\mathbf{e}_\varphi$, see Figure 1(c). Again, the base vectors are not fixed, but are a function of position. The spherical velocities are V_r, V_θ and V_φ which are directed along $\mathbf{e}_r$, $\mathbf{e}_\theta$ and $\mathbf{e}_\varphi$.

The transformation from polar, cylindrical and spherical velocities to Cartesian velocity and vice versa can be found in Appendix A.

Instead of conventional coordinates one can also use orbital elements to describe

positions and motion with respect to a reference frame. These orbital elements are discussed in Section 2.4. Transformation from orbital elements to Cartesian coordinates and vice versa can be found in [Wakker, 2007].

2.3. Two-Body Problem

The two-body problem deals with the motion of a body i with respect to a non-rotating reference frame with body k as origin. It is assumed that both bodies are point masses and that no forces are acting on the two-body system except the gravitational forces between bodies i and k. The motion of body i with respect to body k can then be described by [Cornelisse et al., 1979]:

$$\ddot{\mathbf{r}} = -\frac{\mu}{r^3}\mathbf{r} \tag{2.13}$$

where $\mathbf{r}$ is the position vector of body i, $\mu = G(m_k + m_i)$ is the gravitational parameter, G is the universal gravitational constant equal to 6.673×10^{-11} $\mathrm{m^3kg^{-1}s^{-2}}$ and m_k and m_i are the masses of body k and i, respectively. Typically, the mass of body i (a satellite) is negligible with respect to the mass of the central body k, so μ can be approximated by Gm_k

The shape of the orbit of body i with respect to body k is a conic section, i.e. a circle, ellipse, parabola or hyperbola. As a consequence the trajectory can be described by [Cornelisse et al., 1979]:

$$r = \frac{H^2/\mu}{1 + e\cos\theta} = \frac{p}{1 + e\cos\theta} \tag{2.14}$$

where r is the radial distance, θ is the true anomaly, e is the eccentricity, p is the semi-latus rectum and H is the magnitude of the angular momentum (per unit mass) of body i. The trajectory corresponding to an elliptical orbit is shown in Figure 2.2. In this figure also the semi-major axis a, the eccentricity e, the radial distance r and the true anomaly θ are indicated. The orbital elements which describe the geometry of an orbit are discussed in Section 2.4.

For the variation of the velocity along a conic section one can obtain the following relation, called the *vis-viva integral* [Cornelisse et al., 1979]:

$$V^2 = \mu\left(\frac{2}{r} - \frac{1}{a}\right) \tag{2.15}$$

Furthermore, one may write the velocity in a radial and normal component using the flight path angle γ as $V_r = \dot{r} = V\sin\gamma$ and $V_\theta = r\dot{\theta} = V\cos\gamma$, respectively.

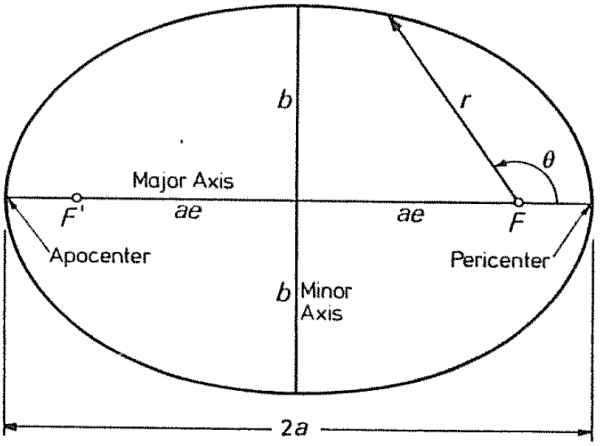


Figure 2.2: Elliptical orbit [Wakker, 2007].

The radial velocity component can be obtained by differentiating Eq. (2.14):

$$\dot{r} = \frac{(H^2/\mu)e\dot{\theta}\sin\theta}{(1+e\cos\theta)^2} = r^2\dot{\theta}\frac{\mu e}{H^2}\sin\theta = \frac{\mu}{H}e\sin\theta \tag{2.16}$$

In addition, since $H = rV_\theta = r^2\dot{\theta}$, the normal velocity component can be obtained as follows:

$$r\dot{\theta} = \frac{H}{r} = \frac{\mu}{H}(1+e\cos\theta) \tag{2.17}$$

One can now eliminate θ from Eqs. (2.16) and (2.17) to obtain [Wakker, 2007]:

$$\dot{r}^2 + \left(r\dot{\theta} - \frac{\mu}{H}\right)^2 = \left(\frac{\mu e}{H}\right)^2 \tag{2.18}$$

This equation corresponds to a circle with radius $\mu e/H$ and the center at a distance μ/H from the origin. The plot of this circle in the $(\dot{r},r\dot{\theta})$-plane is called a *velocity hodograph*. The velocity hodographs for an elliptic, parabolic and hyperbolic orbit are shown in Figure 2.3. For a circular orbit ($e = 0$) the velocity hodograph is simply a point on the $r\dot{\theta}$-axis at a distance μ/H from the origin.

Velocity hodographs, and also acceleration hodographs, can provide insight in the characteristics of orbits and may be used to compute (ballistic) interception and rendezvous trajectories [Altman, 1965]. These two types of trajectories are especially of interest for the application of low-thrust propulsion. Velocity and acceleration hodographs give insight in what the thrust profile for a low-thrust trajectory should look like.

One could therefore think of modifying an acceleration hodograph such that it represents a low-thrust trajectory, and after that analyze how the corresponding velocity hodograph looks like and behaves. Vice versa, a trajectory could

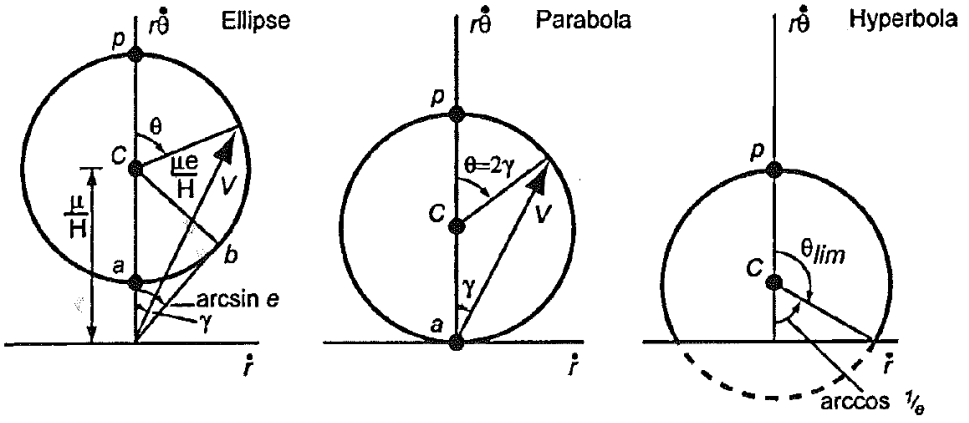


Figure 2.3: Velocity hodographs for elliptical, parabolic and hyperbolic motion [Wakker, 2007].

be designed in a velocity hodograph and the corresponding thrust profile might then be obtained from the acceleration hodograph. As a result, relations between hodographs and thrust profiles might be discovered, which could lead to interesting insights for designing low-thrust trajectories. We will elaborate on the idea of using hodographs for low-thrust trajectory design in Chapter 5.

For elliptical orbits, which are most common for spacecraft and celestial bodies, the orbital period T is given by [Cornelisse et al., 1979]:

$$T = 2\pi\sqrt{\frac{a^3}{\mu}} \tag{2.19}$$

When two bodies orbit the same central body, then the relative positions of these bodies is the same after a certain period. This period is called the *synodic period*, T_{syn}, and depends on the orbital periods of the two orbiting bodies, T_1 and T_2, as follows [Cornelisse et al., 1979]:

$$\frac{1}{T_{syn}} = \left|\frac{1}{T_1} - \frac{1}{T_2}\right| \tag{2.20}$$

The synodic period is of interest for transfer orbits, since the characteristics of a transfer orbit depend on the relative positions of the departure and arrival body.

2.4. Orbital Elements

In the previous section the two-body problem has been discussed. For this problem the orbit of a body is a conic section of which the shape, size and orientation do not change. As a consequence such an orbit can be specified by six orbital elements. These orbital elements are constants of motion, i.e. for every point in the orbit they have the same numerical value. Here, two sets of orbital elements are discussed.

Keplerian orbital elements The Keplerian orbital elements are a, e, i, ω, Ω and τ or M_0. The semi-major axis a and the eccentricity e determine the shape and size of the orbit, e.g. see Figure 2.2 for an ellipse. The argument of pericenter ω, the inclination i and the right ascension of the ascending node Ω determine the orientation of the orbit. Finally, the time of pericenter passage τ or the mean anomaly at epoch M_0, i.e. the position of the orbiting body at a specific time (the "epoch"), relates time with position in the orbit.

Modified equinoctial elements A disadvantage of the Keplerian orbital elements is that Ω and ω become indeterminate as i and e, respectively, tend to zero. Furthermore, the equations of motion written using these elements can easily become singular (see Eqs. (A.14)). By applying the modified equinoctial elements one can avoid the occurrence of indeterminacy and singularity. The modified equinoctial elements are defined as [Walker et al., 1985]:

$$\begin{array}{lll} p = a(1-e^2), & f = e\cos(\omega+\Omega), & g = e\sin(\omega+\Omega), \\ h = \tan(i/2)\cos\Omega, & k = \tan(i/2)\sin\Omega, & L = \Omega+\omega+\theta \end{array} \tag{2.21}$$

The sixth element L is called the true longitude and it can be used as independent variable instead of time.

For spacecraft that fly in an unperturbed (Keplerian) orbit the orbital elements are constant. In case a non-Keplerian orbit is flown, the orbital elements of a fictitious momentary Keplerian orbit can be determined using the position and velocity at that instant. The orbital elements of such a momentary Keplerian orbit are called *osculating orbital elements*.

2.5. Perturbations

In the two-body problem it is assumed that the two bodies are point masses and that the only forces acting on the two-body system are the gravitational forces between both bodies. However, in reality, celestial bodies and spacecraft are not point masses and many other forces besides their mutual gravitational force act on them. The additional forces are called *perturbing forces*. These forces will namely cause the orbit to be perturbed and not be purely Keplerian anymore.

If perturbing forces act on the spacecraft, then the osculating orbital elements change. In such a case the variation of the orbital elements can be expressed as a function of the components of the total perturbing force. Gauss' form of Lagrange's Planetary equations expresses the effect of a perturbing acceleration on the Keplerian osculating orbital elements. In Appendix A these Gauss' planetary equations are given for Keplerian and modified equinoctial elements.

In the design process of low-thrust trajectories it can be very beneficial to neglect perturbing forces since this simplifies the mechanics of the problem significantly.

In order to see if one may neglect perturbing forces with respect to the main gravitational attraction and the low-thrust propulsive force, the main perturbation forces are discussed in the following.

Gravity field perturbing forces In case the mass distribution of a celestial body is perfectly radially symmetrical, one can assume (regarding forces) that the body is a point mass. In general, a celestial body can, however, have an arbitrary mass (density) distribution and therefore no point mass model can be assumed. Nevertheless, since perturbing accelerations due to "irregular gravity" are inversely proportional to the radial distance, these perturbations should only be taken into consideration nearby celestial bodies.

The so-called $J2$-effect is the result of the largest perturbing term of the Earth's gravity field. From Figure 2.4 it is clear that the acceleration due to the $J2$-effect is smaller than 10^{-7} m/s^2 at altitudes above 150,000 km. This acceleration is very small and therefore gravity field perturbations can be neglected for interplanetary flight.

Attraction by other celestial bodies Another perturbation effect, which does apply to interplanetary flights, is the third-body effect. This is the effect which a perturbing body d has on the motion of a spacecraft s about a body k. The maximum relative perturbing acceleration due to the perturbing body d is given by [Wakker, 2007]:

$$\left(\frac{a_d}{a_k}\right)_{max} = \frac{m_d}{m_k}\left(\frac{r_s^2}{r_{sd}^2} - \frac{r_s^2}{r_d^2}\right) \tag{2.22}$$

where a_d is the perturbing acceleration, a_k is the main acceleration of body s due to the attraction by body k, r_s is the distance from body s to body k, r_{sd} the distance from body s to perturbing body d and r_d the distance from body k to perturbing body d.

For interplanetary flights body k is the Sun and therefore the ratio m_d/m_k is very small. The term between brackets on the right-hand side of Eq. (2.22) can, however, become large as the spacecraft gets relatively close to a perturbing body. At the sphere of influence of Earth (9.2×10^5 km from Earth [Cornelisse et al., 1979]) the acceleration to due the Sun's attraction is about 5.9×10^{-3} m/s^2 whereas the Earth's gravitational acceleration is about 4.7×10^{-4} m/s^2. So, the perturbing acceleration due to the Earth's attraction is an order of magnitude smaller than the Sun's attraction, but significant. In general, interplanetary flights are flown much further away from planets, except during departure, arrival and possible swingbys, and therefore third-body perturbation forces can be considered to be small.

Aerodynamic forces The acceleration that a satellite experiences due to aerodynamic forces is a function of the atmospheric density. At altitudes of 1000 km

above the Earth the density is so low that the atmospheric drag can, in general, be neglected completely [Cornelisse et al., 1979]. From Figure 2.4 it is clear that the perturbing acceleration due to atmospheric drag is smaller than 10^{-7} m/s^2 at altitudes above 1000 km. During interplanetary flight aerodynamic forces are even not present at all and therefore perturbations due to aerodynamic forces may be neglected.

Radiation pressure For the acceleration due to radiation pressure, one can write [Wakker, 2007]:

$$f = C_R \frac{WS^\star}{Mc} \tag{2.23}$$

where W is the power density of the incoming radiation, M is the mass of the satellite, $S^\star$ is the effective cross-sectional area, c is the speed of light and C_R is the satellite's reflectivity.

At Earth the intensity of the solar radiation is 1360 W/m^2. When one assumes the conservative values $S^\star/M = 11$ m^2/kg and $C_R = 1.9$ which are applicable to balloon-type satellites, then the acceleration due to solar radiation pressure is 95 μm/s^2.

Near Mercury the solar radiation pressure is five times higher than near Earth, resulting in an acceleration of about 5×10^{-4} m/s^2. This acceleration is rather large, but is only applicable to balloon-type satellites. For most common spacecraft $S^\star/M$ is a thousand times lower and the perturbing acceleration due to radiation is only of the order 10^{-7} m/s^2 or less. Therefore perturbing accelerations due to radiation pressure may be neglected during interplanetary flight.

Electromagnetic forces Near celestial bodies a magnetic field might be present which could generated electromagnetic forces. However, since the accelerations due electromagnetic forces decrease strongly with increasing altitude, they can be neglected during interplanetary flights.

Relativistic effects From the general theory of relativity it is known that Newton's law of gravitation describes a real gravitational field only in a first-order approximation. Due to relativistic effects e.g. precession of the pericenter may occurs. The effect is, however, very small for short periods of time (even for one century) and therefore relativistic effects may be neglected when designing low-thrust trajectories.

Overview In Figures 2.4, 2.5 and 2.6 the magnitudes of the main perturbing forces are shown for the near-Earth and inner and outer solar system regions, respectively.

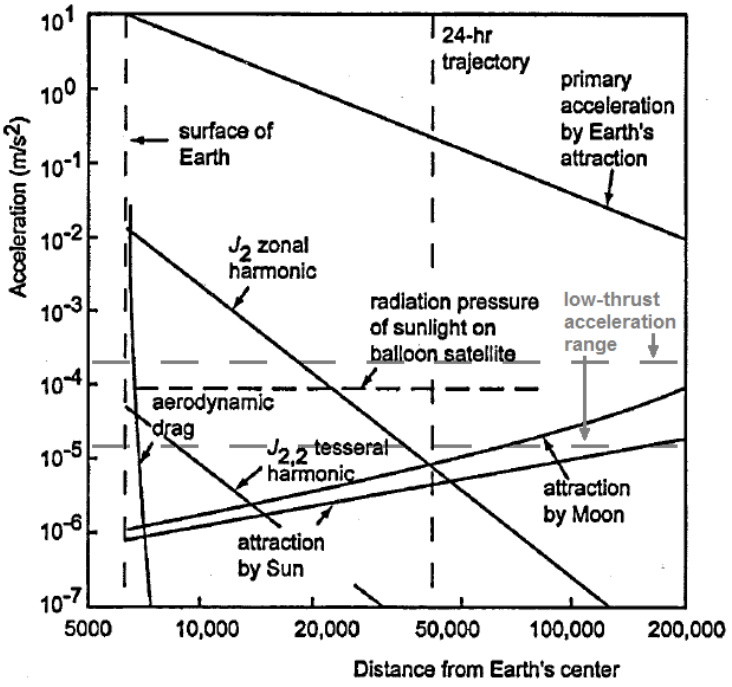


Figure 2.4: Sketch of magnitude of some types of perturbing forces near the Earth, as a function of the orbital altitude [Wakker, 2007].

The accelerations of a spacecraft due to low-thrust propulsion has been plotted as a range with a lower bound of 1.5×10^{-5} m/s^2 (corresponding to MUSES-C [Kawaguchi et al., 2006]) and a conservative upper bound of 2.0×10^{-4} m/s^2 (corresponding to SMART-1 [Racca et al., 2002]). The perturbing acceleration due to solar radiation pressure, as shown in Figures 2.5 and 2.6, is calculated assuming a cross-sectional area $S^\star$ equal to 40 m^2 (BepiColombo [Benkhoff et al., 2010]), a spacecraft mass M of 400 kg and a reflectivity C_R of 1.9. These values can be considered to correspond to a worst-case situation.

From Figure 2.4 it is clear that near Earth perturbing forces may not be neglected with respect to the low-thrust propulsion force. At low altitudes the gravity-field perturbing forces are significantly large and at high altitudes the attraction of the Moon and Sun become significant.

Considering low-thrust during interplanetary flight, Figures 2.5 and 2.6 show that next to the gravitational attraction by the Sun, the acceleration due to low-thrust propulsion is dominant. Third-body perturbing forces and solar radiation pressure only become significantly large, close to planets or close to the Sun, respectively. Elsewhere, the acceleration due to low-thrust is at least an order of magnitude larger than other perturbing forces. Therefore as a first-order estimate perturbing forces may be neglected during interplanetary flight.

For low-thrust trajectory design it is important to quickly obtain feasible and near-optimal trajectories. Obtained solutions can be first-order approximations and need not to be highly accurate. Since perturbing forces during interplanetary

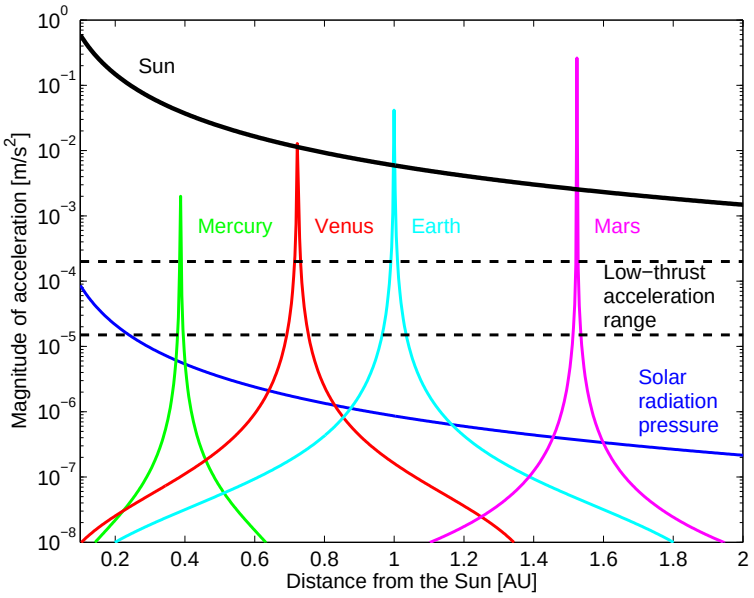


Figure 2.5: Magnitude of various types of perturbing forces in the inner region of the solar system.

flight are much smaller than the low-thrust propulsion force, the improvement of low-thrust trajectory solutions when taking into account perturbing forces is marginal. However, considering perturbing forces in the design of low-thrust trajectories will significantly slow down the design process.

For these reasons all perturbing forces will be neglected in low-thrust trajectory design. As a result, the low-thrust trajectory problem is reduced to a two-body problem with an additional thrust force.

2.6. Non-Perturbed Transfer Orbits

In this section unperturbed transfer orbits are considered. Unperturbed means that during transfer no forces other than the central gravitational attraction act on the spacecraft and that required maneuvers are executed by impulsive shots.

These type of transfer orbits are considered here since they may provide useful first-order information for low-thrust trajectory design. Furthermore, unperturbed transfer orbits are often considered when high-thrust propulsion systems are used. It can therefore be interesting to compare optimal high-thrust transfer trajectories with optimal low-thrust ones.

2.6.1. General

Between the initial and final orbit many different transfer orbits can be flown. However, usually one is interested in optimal transfer orbits, i.e. transfer orbits that require a minimum amount of propellant.

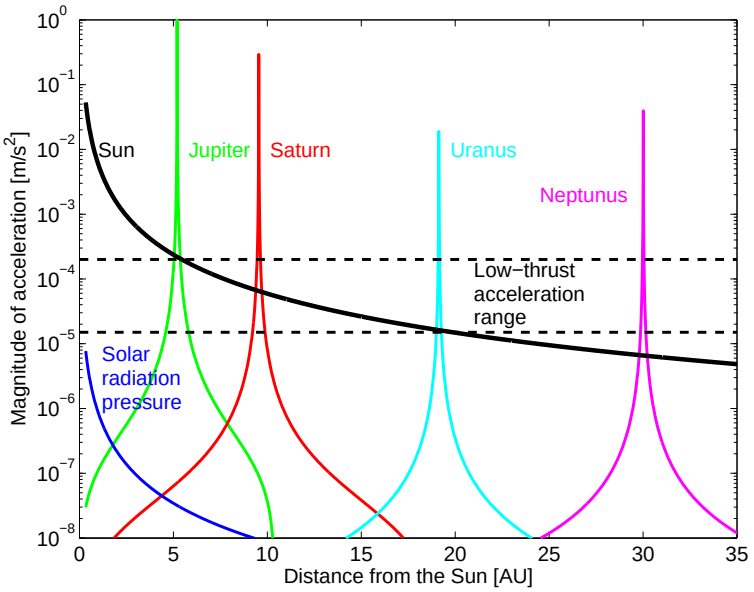


Figure 2.6: Magnitude of various types of perturbing forces in the outer region of the solar system.

As said, it is assumed that the required maneuvers are executed by impulsive shots, which means that all propellant is consumed instantaneously. In addition, the maneuvers are assumed to be accomplished by the same rocket engine and that between successive maneuvers no mass is expelled. Tsiolkowsky's law can be applied which yields the mass ratio between the final mass M_e and initial mass M_0 of the spacecraft after n impulsive shots [Wakker, 2007]:

$$\frac{M_e}{M_0} = \exp\left[-\left\{\sum_{i=1}^{n}(\Delta V)_i\right\}/V_e\right] \tag{2.24}$$

where $(\Delta V)_i$ is the magnitude of the i^{th} impulsive shot and V_e is the (effective) exhaust velocity of the rocket engine. From Eq. (2.24) it follows that the final mass is maximum when:

$$\sum_{i=1}^{n}(\Delta V)_i = minimal \tag{2.25}$$

So maximum M_e is equivalent to minimum $\sum \Delta V_i$.

For transfers between coplanar orbits, both circular or elliptical, the *Hohmann transfer orbit* is optimal, i.e. requires at least amount of ΔV [Wakker, 2007]. Such Hohmann transfer orbits touch the initial and final orbit at pericenter and apocenter. Hohmann transfers are considered now regarding the required total impulse, because the required ΔV for Hohmann transfers can be used as benchmark for the required ΔV for low-thrust transfers.

2.6.2. Optimum transfer between coplanar circular and elliptical orbits

For a Hohmann transfer from a circular orbit to an elliptical orbit, which pericenter distance is larger than the radius of the initial circular orbit, it is optimal to inject in the final orbit at its apocenter. The required total impulse ΔV_{tot} for such a transfer is given by [Wakker, 2007]:

$$\frac{\Delta V_{tot}}{V_{c_1}} = \sqrt{\frac{2}{(1+e)(1+e+r_1/a)}}\left(1+e-\frac{r_1}{a}\right) + \sqrt{\frac{r_1}{a}\left(\frac{1-e}{1+e}\right)} - 1 \tag{2.26}$$

where V_{c_1} is the initial circular velocity, r_1 is the initial radius and a and e are the semi-major axis and eccentricity of the final orbit, respectively. This equation can also be used to compute the required ΔV for a Hohmann transfer from a large elliptic to a small circular orbit.

On the other hand, for a Hohmann transfer from an elliptical to a circular orbit which radius is larger than the apocenter distance of the initial elliptical orbit, it is optimal to depart from the initial orbit at its pericenter. The required ΔV for such a transfer is given by [Wakker, 2007]:

$$\frac{\Delta V_{tot}}{V_{c_2}} = \sqrt{\frac{2r_2/a}{(1-e)[1+a/r_2(1-e)]}}[1-a/r_2(1-e)] - \sqrt{\frac{r_2}{a}\left(\frac{1+e}{1-e}\right)} + 1 \tag{2.27}$$

where V_{c_2} is the final circular velocity, r_2 is the final radius and a and e are the semi-major axis and eccentricity of the initial orbit, respectively. This equation can also be used to compute the required total impulse for a Hohmann transfer from a large circular to a small elliptic orbit. In addition, both Eq. (2.26) and (2.27) can be used to compute the required ΔV for Hohmann transfers between two coplanar circular orbits.

The ΔV required for unperturbed transfer orbits can be used to compare such orbits with low-thrust transfer orbits and, in addition, to make a first-order guess for low-thrust trajectory parameters.

Furthermore, from unperturbed-transfer theory and from Gauss' planetary equations Eqs. (A.14) useful insights can be gained about how to efficiently change an orbit. Especially, changing the semi-major axis a and inclination i are of interest, since these require a large amount of energy. Using Gauss' planetary equations one can write the time-rates of change of a and i as function of a perturbing acceleration as follows [Wakker, 2007]:

$$\frac{da}{dt} = 2f_t\frac{a^2}{\mu}V \tag{2.28}$$

$$\frac{di}{dt} = f_o\frac{r}{\sqrt{\mu p}}\cos u \tag{2.29}$$

where f_t is the acceleration tangential to the velocity vector, f_o is the out-of-plane acceleration, and u is the argument of latitude.

From these equations and for unperturbed-transfer theory in general the following important insights can be found:

- The orbital energy and semi-major axis are changed most efficiently by thrusting tangential to the velocity at the highest possible velocity.
- The orbital plane is changed most efficiently at a large distance from the Sun where the velocity is lowest.
- The velocity is changed most efficiently by applying thrust perpendicular to the radius vector, since gravity losses are omitted in this way.

This discussion on orbit changes concludes the chapter about orbital mechanics. The concepts, physics and results from this chapter will form the basis for a novel low-thrust trajectory design method.

3. Low-Thrust Propulsion

In the previous chapter the orbital mechanics regarding low-thrust trajectory design has been considered. In addition, designing low-thrust trajectories requires making assumptions about and models for the low-thrust propulsion system. In order to do this properly basic knowledge about low-thrust propulsion techniques is required. In addition, understanding of the dynamics of thrusting spacecraft is desired.

The term 'low thrust' originates from that fact that electric propulsion systems produce very low thrust levels: typically about 0.1 N (see Table 3.2). So, when one refers to low-thrust propulsion, one actually refers to electric propulsion. In this chapter the characteristics of electric propulsion are discussed. In addition, the applications of electric propulsion is briefly treated and the motion of spacecraft under low-thrust is considered.

3.1. Electric Propulsion Systems

The thrust of a rocket engine is a function of the mass flow $\dot{M}$ and the exhaust velocity V_e of the propellant [Stuhlinger, 1964]:

$$T = \dot{M} V_e \tag{3.1}$$

In addition, the thrust can also be written as a function of the available power W [Stuhlinger, 1964]:

$$T = \sqrt{2\dot{M}W} = \frac{2W}{V_e} \tag{3.2}$$

On the one hand, from Eq. (3.2) follows that, when the available power is fixed, a higher exhaust velocity results in less thrust. On the other hand, Eq. (3.1) shows that, at a fixed thrust level, a higher exhaust velocity results in less propellant mass flow. As a consequence, little propellant consumption in combination with high thrust levels can only be achieved with high exhaust velocities and high power levels.

Furthermore, from Eq. (3.2) follows that the thrust depends linearly on the input power. In tests of the Deep Space 1 ion thrusters [Sauer Jr., 1997] it was found that this linear dependency of thrust on the input power, in general, also holds in practice.

A characteristic value of a rocket engine is the value of its specific impulse. The specific impulse I_{sp} is the ratio of the thrust to the propellant mass flow in units of weight [Stuhlinger, 1964]:

$$I_{sp} = \frac{T}{\dot{M} g_0} = \frac{V_e}{g_0} \tag{3.3}$$

Table 3.1: Characteristics of various electric propulsion systems: ⋆electrothermal, †electrostatic and ‡electromagnetic [Wertz and Larson, 1999].

Propulsion type	Specific impulse [s]	Electric power/thrust [kW/N]	Energy conversion efficiency [%]	Thrust range [N]
Resistojet⋆	150 - 700	1.3 - 2	60	0.005 - 0.5
Arcjet⋆	450 - 1500	6 - 15	91 - 95	0.05 - 5
Ion†	2000 - 6000	22 - 36	87 - 91	5×10^{-6} - 0.5
Hall effect thruster†	1500 - 2500	16 - 19	91 - 93	5×10^{-6} - 0.1
Magnetoplasma-dynamic‡	2000	10 -19		25 - 200
Pulsed plasma‡	1500	83 - 100	80	5×10^{-6} - 0.005
Pulsed inductive‡	2500 - 4000	40	70	2 - 200

where $g_0 = 9.81$ m/s^2 is the standard gravity acceleration. The specific impulse indicates the efficiency of a rocket engine. Since exhaust velocities of electric propulsion systems are high, the specific impulses are also high, which indicates the high efficiency of electric propulsion.

Types of electric propulsion In general three types of electric propulsion can be distinguished [Stuhlinger, 1964] [ESA, 2004], namely:

- **Electrothermal propulsion**: accelerate the propellant by heating,
- **Electrostatic propulsion**: accelerate the ionized propellant by means of an electric field,
- **Electromagnetic propulsion**: accelerate the ionized propellant using orthogonal electric and magnetic fields.

An overview of the characteristics of several electric propulsion systems is shown in Table 3.1. This table nicely shows that the achievable thrust of current electric propulsion technology is very low. Only electromagnetic systems can achieve high thrust levels, but these require too much power [ESA, 2004]. The specific impulse is on the other hand high. Ion engines can achieve specific impulses of up to 6000 s, which is equivalent to an exhaust velocity of approximately 60,000 m/s, see Eq. (3.3).

In Table 3.2 the characteristics of four ion propulsion systems (to be) used in actual space missions are shown. The highest thrust is 290 mN, which is equivalent to a thrust acceleration of 2.9×10^{-4} m/s^2 for a spacecraft of 1000 kg. This is about similar to the initial thrust acceleration of SMART-1 (370 kg) of 2.0×10^{-4} m/s^2 [Racca et al., 2002]. Therefore, for the design of low-thrust trajectories,

Table 3.2: Characteristics of electric propulsion systems of past and future space missions [Rayman and Williams, 2002] [Kuninaka, 2008] [Rayman et al., 2006] [Benkhoff et al., 2010].

Propulsion type	Specific impulse [s]	Thrust range [mN]	Input power [kW]	Delivered ΔV [km/s]	Propellant mass [%]
Deep Space 1	1900 - 3200	19 - 92	0.5 - 2.3	4.5	17
Hayabusa	2900	5 - 25	0.3 - 1.1	2.2	12
Dawn	1900 - 3200	19 - 92	0.5 - 2.6	10.7	36
BepiColombo	4300	100 - 290	7 - 14	5.8	13

thrust accelerations up to 3.0×10^{-4} m/s^2 will be considered feasible using current technology.

Power Unlike chemical propulsion which is energy limited, electric propulsion is power limited. The propellants of chemical propulsion systems are both the power source and the exhaust fluid. Since chemical propellants have only a fixed amount of energy per unit mass, the exhaust velocity and specific impulse are limited. Therefore these systems are called energy limited.

Electric propulsion systems have a separate power source and power conversion unit to transfer the energy to the propellant. Neglecting the lifetime of the power source, an arbitrary amount of energy can delivered to the propellant. As a result the exhaust velocity and specific impulse can become very large. However, the amount of power is limited by the power source. As a consequence the thrust level is limited, see Eq. (3.2), and therefore electric propulsion is called power limited.

The power for electric propulsion systems is usually supplied by either a solar array, i.e. solar electric propulsion (SEP), or by an onboard nuclear reactor, i.e. nuclear electric propulsion (NEP).

In case of SEP, the available power depends on the local power density of the solar radiation and therefore on the distance from the Sun. This can most simply be modeled as follows:

$$E = \frac{P}{4\pi r^2} \tag{3.4}$$

where E is the power density, P the total power emitted by the Sun and r the distance from the Sun.

Since the available power of a SEP system decreases with increasing distance from the Sun, the power may become too low to be sufficient for operations when flying to the outer region of the solar system. NEP can in that case be more suitable since the available power is constant throughout the mission (neglecting inherent degradation).

For low-thrust trajectory design, we will simply consider achievable thrust or acceleration levels instead of amount of available power or energy. In case of SEP, the relation between the available solar power and thrust is assumed to be linear.

As a consequence, the maximum thrust level for SEP is inversely proportional to the square of the distance from the Sun.

3.2. Applications

Electric propulsion systems are characterized by their low thrust level and their long lifetimes. As a result these engines are specially suitable for applications which require long operation times without significant thrust. These applications include satellite stationkeeping, attitude control, drag compensation and orbit transfers, including plane changes.

Low-thrust propulsion can be also be applied to interplanetary flight in case time is not crucial. Continuous thrusting over long periods can result in high velocity changes, requiring little propellant (see Table 3.2). This feature is the most important benefit of electric propulsion over chemical propulsion.

3.3. Rocket Motion

The motion of a spacecraft in a central gravity field without additional forces is given by Eq. (2.13). In case a thrust force is added, the equations of motion become [Cornelisse et al., 1979]:

$$\ddot{\mathbf{r}} + \frac{\mu}{r^3}\mathbf{r} = \frac{\mathbf{T}}{M} = \mathbf{f} \tag{3.5}$$

where $\mathbf{T}$ is the thrust vector, M is the instantaneous mass of the spacecraft and $\mathbf{f}$ is the thrust-acceleration vector.

Using these equations of motion one is able to compute the motion of a spacecraft in a central gravity field while thrusting. In order to fly a specific trajectory three variables need to be controlled as a function of time, namely either the three components of the thrust force or the thrust magnitude and two thrust angles. The combination of these three variables, as a function of time, is called a thrust profile.

A thrust profile can be constrained by applying e.g. a constant thrust force, radial thrust or thrust tangential to the velocity vector. An advantage of such a constraint is that the number of variables in the trajectory design and optimization is reduced. If, for example, the thrust direction or thrust magnitude is constrained, then it does not need to be optimized. As a consequence, the optimization of a low-thrust trajectory becomes more simple, which is beneficial for the speed of the optimization. A disadvantage is that constraining the thrust profile, in general, results in non-optimal solutions for the low-thrust trajectory. This may, however, be considered as a minor disadvantage, since only first-order design of low-thrust

trajectories is considered in this report. So fully optimal solutions are not of interest.

Often, maximization of the final spacecraft mass is desired. According to Tsiolkovsky's rocket equation this is equivalent to minimizing the required ΔV [Wakker, 2007]:

$$\Delta V = V_e \ln \frac{M_0}{M_e} \tag{3.6}$$

Using this equation together with Eq. (3.1), it can be derived that:

$$\Delta V = \int_0^{t_f} f dt \tag{3.7}$$

where $f = |\mathbf{f}| = T/M$ is the magnitude of the thrust acceleration. This expression can be used to determine the required ΔV and final spacecraft mass when the thrust profile is known.

Now that electric propulsion systems and the dynamics of thrusting spacecraft are discussed, we can have a look at the design of low-thrust trajectories.

4. Shaped-Based Methods

In this chapter shape-based methods for low-thrust trajectory design are discussed. In the shape-based approach, a powered spacecraft trajectory is assumed to be of a certain shape. This shape together with the time evolution of the spacecraft along the shape determines the requisite thrust profile, in order to satisfy the equations of motion Eq. (3.5).

Such a trajectory shape can be defined for the planar or the full three-dimensional case and may be expressed using various coordinate systems. Irrespective of the implementation, the shape needs to be such that it is able to satisfy the boundary conditions on the trajectory.

As discussed in Chapter 2, these shape-based methods only consider the two-body problem with an additional thrust force. The equations of motion corresponding to the two-body problem with an additional thrust force is provided by Eq. (3.5) and is repeated here for the sake of clarity:

$$\ddot{\mathbf{r}} + \frac{\mu}{r^3}\mathbf{r} = \frac{\mathbf{T}}{M} = \mathbf{f} \tag{4.1}$$

Since shape-based methods may be able to satisfy the boundary conditions on the trajectory and since most computations are done analytically, these methods are very suitable for quickly obtaining feasible and possibly near-optimal low-thrust trajectories.

In the following sections, several shape-based methods are discussed and finally a comparison is made.

4.1. Exponential Sinusoid

Petropoulos found that the shape of an exponential sinusoid is very useful for representing planar low-thrust trajectories [Petropoulos and Sims, 2002] [Petropoulos and Longuski, 2002] [Petropoulos and Longuski, 2004]. For the planar case, the equations of motion in polar coordinates are given by:

$$\ddot{r} - r\dot{\theta}^2 + \frac{\mu}{r^2} = f\sin\alpha \tag{4.2}$$

$$\frac{1}{r}\frac{d}{dt}(r^2\dot{\theta}) = f\cos\alpha \tag{4.3}$$

The exponential sinusoid (exposin) shape is written, in polar coordinates, as follows:

$$r = k_0 \exp\left[k_1 \sin\left(k_2\theta + \phi\right)\right] \tag{4.4}$$

where k_0, k_1, k_2 and ϕ are constants. The winding parameter k_2 determines the number of revolutions, the dynamic range parameter k_1 controls the ratio of the apocenter distance to the pericenter distance, the parameter k_0 is simply a scaling factor and the phase angle ϕ controls the orientation of the exponential sinusoid in the plane.

The values of the parameters need to be chosen such that the resulting trajectory fulfills the boundary conditions for the initial and final position and possibly the time of flight.

Petropoulos chose to apply tangential thrusting for steering along the exposin shape, since this is the simplest analytic case. Moreover, as discussed in Section 2.6, tangential thrusting is the most efficient steering law for changing the orbital energy.

For the tangential thrusting case, one obtains the following expression for the angular velocity:

$$\dot{\theta}^2 = \left(\frac{\mu}{r^3}\right) \frac{1}{\tan^2 \gamma + k_1 k_2^2 \sin\left(k_2 \theta + \phi\right) + 1} \tag{4.5}$$

The time required to fly the trajectory can be found by integrating the inverse of the angular velocity over the polar angle:

$$\text{TOF} = \int_0^{t_f} dt = \int_0^{\theta_f} \frac{dt}{d\theta} d\theta \tag{4.6}$$

In addition, the thrust acceleration can be obtained as function of the polar coordinate θ. The ΔV required to fly the trajectory can then be determined by rewriting Eq. (3.7) as follows:

$$\Delta V = \int_0^{t_f} f dt = \int_0^{\theta_f} \frac{f}{\dot{\theta}} d\theta \tag{4.7}$$

A proper trajectory with the shape of an exponential sinusoid is obtained by picking appropriate values for the parameters k_0, k_1, k_2 and ϕ such that the boundary conditions on position are satisfied and to ensure that the denominator in Eq. 4.5 is always positive (else tangential thrust is not possible). In addition, the parameters may satisfy a condition on the initial velocity or on the time of flight. Petropoulos' approach to satisfy these conditions is rather laborious. A more straightforward method which also allows one to satisfy the TOF constraint can be found in [Izzo, 2006].

Out-of-plane motion Petropoulos also presented a way to match the out-of-plane position. For this purpose an additional thrust component acting along or opposite to the spacecraft's angular momentum vector is used. This out-of-plane thrust acceleration is shaped such that the out-of-plane position is matched at final time. For a description of this procedure see [Petropoulos and Longuski, 2004].

Discussion The exponential sinusoid has been proven to be very suitable for preliminary design of low-thrust trajectories including gravity assists. The success of the exposin stems from its simplicity, speed and capability to provide feasible first-order solutions and, in addition, the lack of better shape-based methods.

However, the major drawback of the exponential sinusoid is that it is not able to satisfy all boundary conditions on velocity. As a consequence impulsive maneuvers are required at the start and at the end of a trajectory to meet these requirements. Another disadvantage is that the exposin is limited to planar cases, since the out-of-plane motion is only approximated. Finally, trajectories with a exponential-sinusoid shape are usually not optimal regarding required ΔV.

Pros

- Fast and simple since only 4 shape parameters
- Suitable for gravity-assist trajectories

Cons

- Two-dimensional, the out-of-plane motion is only approximated.
- Velocity and time-of-flight constraints cannot all be satisfied. This may be good enough for swing-by maneuvers.
- Thrust level is a function of shape and can therefore not be prescribed nor limited.
- Only tangential thrusting can be applied.

4.2. Inverse Polynomial

A major disadvantage of the exponential sinusoid is that it is not able to satisfy the boundary conditions on the initial and final velocity.

Wall developed a trajectory shape which is able to satisfy both the position and velocity boundary conditions in three dimensions [Wall and Conway, 2009] [Wall, 2008]. Wall developed this method in cylindrical coordinates in which the equations of motion are:

$$\begin{aligned} \ddot{r} - r\dot{\theta}^2 + \frac{\mu}{s^3} r &= f_{in} \sin\alpha \\ \frac{1}{r}\frac{d}{dt}(r^2\dot{\theta}) &= f_{in}\cos\alpha \\ \ddot{z} + \frac{\mu}{s^3} z &= f_z \end{aligned} \tag{4.8}$$

where $s = \sqrt{r^2 + z^2}$ is the distance from the Sun and f_{in} and f_z are the thrust acceleration in the xy-plane and in the z-direction, respectively.

The shape used to model the planar motion is a sixth-degree inverse polynomial given by:

$$r = \frac{1}{a + b\theta + c\theta^2 + d\theta^3 + e\theta^4 + f\theta^5 + g\theta^6} \tag{4.9}$$

Just like Petropoulos, Wall assumes in-plane tangential thrusting in order to simplify equations. The parameters a to c are used to solve for the boundary conditions on initial position and velocity. The remaining four coefficients need to satisfy the final position and velocity and the time of flight constraint. The coefficient d is chosen as parameter to solve for the time of flight. As a consequence the values of e, f and g are determined by the value of d and final position and velocity constraints.

Again the time of flight can be found by integrating the inverse of the angular velocity $\dot{\theta}$ over the transfer angle. The expression for $\dot{\theta}$ for the sixth-degree inverse polynomial with tangential thrust is given by:

$$\dot{\theta}^2 = \frac{\mu}{r^4} \frac{1}{[(1/r) + 2c + 6d\theta + 12e\theta^2 + 20f\theta^3 + 30g\theta^4]} \tag{4.10}$$

To match the transfer time with the specified TOF, a proper value for d has to be found. This is done by varying the value of d using a root-finding function until the transfer time matches the specified TOF. Simultaneously to finding the value of d, the final boundary conditions are satisfied using the coefficients e, f and g. The values of the coefficients a, b, c, e, f and g which satisfy the boundary conditions can all be found by solving a linear system of equations.

The modeling of the motion in the z-direction is accomplished by the addition of a second shape with four coefficients such that the four out-of-plane boundary conditions can be satisfied:

$$z(\theta) = a_z + b_z\theta + c_z\theta^{q-1} + d_z\theta^q \tag{4.11}$$

Wall chose this particular shape because it was known a priori that the coefficients could be determined via a linear system of equations. The variable q is an integer equal to or greater than three and its value should be chosen based on resulting trajectories.

It is assumed that the axial distance z is small with respect to the radial distance r and that therefore the distance from the Sun s is equal to r. This assumption makes it possible to use all above equations corresponding to the in-plane motion without any modifications. However, due to this assumption the three-dimensional method is only valid for inclination changes less than 15° where the relative error between r and s is less than 5%. This limitation, however, includes most space missions to other celestial objects.

The magnitude of the in-plane thrust acceleration f_{in} is found by substituting the expressions for r and the time derivatives of θ in the equation of motion for the normal direction (4.3). In addition, the magnitude of the out-of-plane thrust acceleration f_z is found by substituting the expressions for z, its derivatives and

the time derivatives of θ in the equation of motion for the z-direction (4.8). Finally, the total required thrust acceleration is found using Pythagoras:

$$f_{tot} = \sqrt{f_{in}^2 + f_z^2} \tag{4.12}$$

and the total ΔV required to fly the trajectory can be found using Eq. (4.7).

Wall uses simple numerical quadratures for computing the time of flight and ΔV instead of more accurate numerical integration methods. According to Wall, simple quadratures are much faster when evaluating analytic trajectories and the integration error is small for good trajectories (since the state variables are more smooth).

Discussion Wall tested the inverse polynomial on several cases, in which the launch date, arrival date and number of revolutions were optimized using a Genetic Algorithm. The method was found to provide good trajectories regarding required ΔV. The trajectories are, however, bad regarding required thrust acceleration. This is a consequence of the fact that the trajectory and thus the profile are fully determined by the boundary conditions.

All together, the strengths of the inverse polynomial are its simplicity and ability to deal with three-dimensional rendezvous problems. However, drawbacks are the lack of flexibility and incapability to obtain feasible thrust levels according to the author.

Wall has improved the inverse-polynomial shaping method recently; the improved version can be found in [Wall and Novan, 2012].

Pros

- Three-dimensional
- Rendezvous possible
- No free parameters need to be optimized

Cons

- Thrust level is a function of shape and can therefore not be prescribed nor limited.
- Limited to inclination changes smaller than 15°
- TOF constraint is satisfied iteratively
- Only tangential thrusting can be applied
- Not flexible because of lack of free parameters

4.3. Spherical Shaping

Novak developed a shaping method based on spherical coordinates (r, θ, ϕ) [Novak and Vasile, 2011]. Again the angle θ is used to parameterize the trajectory and

therefore $r = R(\theta)$, $\phi = \Phi(\theta)$ and $t = T(\theta)$, where R, Φ and T are the shaping functions for the spherical coordinates r, ϕ and time t.

The functions R and Φ model the pure geometry of the trajectory, while the function T shapes the time evolution along the trajectory. R and Φ can be any function in space, however, it is clever to choose expressions for which the boundary constraints on position and velocity can be computed analytically. The evolution of the spacecraft along the trajectory is defined by providing $\dot{\theta}$ as a function of θ. After that $\ddot{\theta}$ can be computed and the thrust profile required to fly the trajectory can be obtained using the equations of motion parametrized with θ:

$$\dot{\theta}\frac{d^2\mathbf{r}}{d\theta^2} + \ddot{\theta}\frac{d\mathbf{r}}{d\theta} = -\mu\frac{\mathbf{r}}{r^3} + \mathbf{f} \tag{4.13}$$

with $\mathbf{r} = [r\cos\theta\cos\varphi, r\sin\theta\cos\varphi, r\sin\varphi]^T$, $\dot{\theta} = 1/t'$ and $\ddot{\theta} = -t''/t'^3$.

The time rate of change of θ is set by defining the time evolution T' where: $T' = \frac{dT}{d\theta} = \frac{1}{\dot{\theta}}$. Novak chose the derivative of T with respect to θ as follows:

$$T' = \sqrt{\frac{DR^2}{\mu}} \tag{4.14}$$

where D is expressed by:

$$D = -r'' + 2\frac{r'^2}{r} + r'\varphi'\frac{\varphi'' - \sin\varphi\cos\varphi}{\varphi'^2 + \cos^2\varphi} + r(\varphi'^2 + \cos^2\varphi) \tag{4.15}$$

This choice for T' results in a thrust vector which has no component normal to the plane spanned by the velocity and angular momentum vector. This result is comparable to tangential thrusting in the planar case.

The shaping functions R, Φ and T need to satisfy ten boundary conditions; four on the initial and final position and six on the initial and final velocity. Only four of six boundary conditions on position have to be satisfied by R, Φ and T, since two conditions are directly satisfied by picking initial and final values for the independent variable θ. The ten boundary conditions can be expressed as four conditions for R, four for Φ and two for T'. The two conditions for T' can, however, be replaced by two conditions for R and Φ using the definition of T', Eq. (4.14).

As a consequence, the functions R and Φ together should have at least ten parameters to satisfy the boundary conditions. In case exactly ten parameters are applied the time of flight is uniquely defined.

Novak presents to following shaping functions:

$$R = \frac{1}{a_0 + a_1\theta + a_2\theta^2 + (a_3 + a_4\theta)\cos\theta + (a_5 + a_6\theta)\sin\theta} \tag{4.16}$$

$$\Phi = (b_0 + b_1\theta)\cos\theta + (b_2 + b_3\theta)\sin\theta \tag{4.17}$$

Novak chose these functions because they can represent Keplerian arcs and since no singularities occurred in tests.

The shaping functions R and Φ together contain 11 free parameters. The additional 11^{th} parameter can be used to satisfy the constraint on the time of flight. The value of this extra parameter is found by varying it using a root-finding function until the time constraint is satisfied (just like Wall does for the inverse polynomial).

Using the equations of motion, Eqs. (4.13), the required thrust acceleration vector $\mathbf{f}$ can be computed. The time of flight and ΔV corresponding to the shaped trajectory are obtained by integrating T' and $|\mathbf{f}|T'$, respectively, over the polar angle θ.

Discussion The spherical shaping method has been tested for several rendezvous missions. On average, compared with the pseudo-equinoctial shaping method (see next section), spherical shaping provides better results regarding required ΔV. However, the best trajectories require higher peak thrusts. Furthermore, for flight times which are short with respect to the number of revolutions about the Sun the time of flight constraint can not be satisfied.

Overall, the best obtained trajectories are near-optimal. However, the thrust level could not be limited and therefore the obtained trajectories might not be feasible.

Pros

- Three-dimensional
- Rendezvous possible
- Better solutions regarding ΔV than pseudo-equinoctial method
- No free parameters need to be optimized
- Can cover unperturbed Keplerian motion

Cons

- Thrust level is a function of shape and can therefore not be prescribed nor limited.
- No flexibility due to lack to free parameters
- TOF constraint is satisfied iteratively
- Higher peak thrusts than pseudo-equinoctial method
- Thrust direction is constrained to the plane spanned by the velocity and angular momentum vector, which is a limitation.

4.4. Pseudo-Equinoctial Shaping

In the previous sections, shaping methods have been discussed which use either polar, cylindrical or spherical coordinates to describe the shape of a low-thrust

trajectory. De Pascale and Vasile [De Pascale and Vasile, 2006] [Vasile et al., 2007] developed a shaping method based on pseudo-equinoctial elements, which were discussed in Section 2.4.

In case modified equinoctial elements are considered, the change of the orbital elements due to perturbing forces is given by Gauss' planetary equations, see Eqs. (A.15) in Appendix A. More specifically, these equations express the time rate of change of the orbital elements as a function of the components of the perturbing acceleration.

By properly shaping the modified equinoctial elements a trajectory can be obtained which is feasible and satisfies the boundary conditions. For the shaping of the elements De Pascale and Vasile chose the true longitude L as independent variable instead of time. This seemed more convenient, because low-thrust trajectories may require continuous thrust over multiple revolutions. The change of the equinoctial elements over true longitude L due to a perturbing acceleration is simply obtained by multiplying Gauss' planetary equations, Eqs. (A.15), by dt/dL which is expressed as follows:

$$\frac{dt}{dL} = \frac{1}{\sqrt{\mu p}} \left(\frac{p}{1 + f\cos L + g\cos L} \right)^2 \tag{4.18}$$

This expression for the time evolution is actually the inverse of dL/dt from Gauss' planetary equations in case perturbing accelerations are neglected, see Eq. (A.15).

The variation of the equinoctial elements as a function of L is then used to obtain the evolution of the Cartesian coordinates of the spacecraft, which is used to compute the required thrust acceleration along the trajectory using Eq. (4.1). In addition, the total required ΔV can be calculated using a modified version of Eq. (3.7):

$$\Delta V = \int_{L_0}^{L_f} |\mathbf{f}| \frac{dt}{dL} dL \tag{4.19}$$

Finally, to make the solution physically possible the following time constraint must be respected:

$$\text{TOF} = \int_{L_0}^{L_f} \frac{dt}{dL} dL \tag{4.20}$$

Summarizing, by defining the change of the equinoctial elements as a function of the true longitude, a low-thrust trajectory and the corresponding thrust profile can be obtained. This is in essence the shaping method for low-thrust trajectories using equinoctial elements.

Shape of trajectory The method described above requires the definition of the shape of the low-thrust trajectory in terms of the equinoctial elements $\boldsymbol{\alpha}$. De

Pascale and Vasile suggested two shapes, namely the linear-trigonometric shape:

$$\tilde{\boldsymbol{\alpha}} = \tilde{\boldsymbol{\alpha}}_0 + \tilde{\boldsymbol{\alpha}}_1(L - L_0) + \boldsymbol{\lambda} \sin(L - L_0 + \varphi) \tag{4.21}$$

and the exponential shape:

$$\tilde{\boldsymbol{\alpha}} = \tilde{\boldsymbol{\alpha}}_0 + \tilde{\boldsymbol{\alpha}}_1 e^{\boldsymbol{\lambda}(L-L_0)} \tag{4.22}$$

where $\boldsymbol{\lambda} = [\ \lambda_1 \quad \lambda_2 \quad \lambda_3\ ]^T$ is a set of shaping parameters. The values of $\tilde{\boldsymbol{\alpha}}_0$ and $\tilde{\boldsymbol{\alpha}}_1$ depend on the initial and final conditions, whereas the value of φ is set empirically for each element based on numerical analysis. The tildes in Eqs. (4.21) and (4.22) indicate that we no longer deal with equinoctial elements, but with pseudo-equinoctial elements, because they do not always exactly satisfy Gauss' planetary equations unless the thrust is equal to zero.

The linear-trigonometric shape corresponds approximately to tangential thrusting with a thrust level that decreases with the square of the distance from the Sun. The exponential shape, on the other hand, corresponds to tangential thrusting with a constant thrust acceleration. Furthermore, the shaping parameter λ_1 is associated to p, λ_2 to the elements f and g, and λ_3 to the elements h and k. These shaping parameters are used to optimize the trajectory for minimum propellant mass ratio subject to constraints on the time of flight and thrust acceleration (where a maximum difference of 2% is allowed with respect to the required TOF). However, also only one shaping parameter can be used to satisfy the time of flight constraint using a root-finding function, see [Novak and Vasile, 2011].

Discussion A strength of the pseudo-equinoctial shaping method is its ability to provide trajectories with feasible thrust profiles for three-dimensional rendezvous problems. In addition, the shaping method is able to satisfy the time of flight constraint, however, this has to be done iteratively or using constraint optimization.

The drawback of pseudo-equinoctial shaping lies in its complexity, since the implementation of the method as well as the optimization of the shaping parameters are difficult tasks.

Pros

- Three-dimensional
- Rendezvous possible
- Flexibility due to free shaping parameters
- Multiple revolutions possible
- Can cover unperturbed Keplerian motion
- Lower peak thrusts than spherical shaping
- Maximum thrust acceleration constraint can be satisfied by applying constraint optimization

Cons

- Free shaping parameters need to be optimized
- Finds higher ΔVs than spherical-shaping method
- TOF constraint is met iteratively or during optimization using a penalty function
- Method is complex to implement

4.5. Pseudo-Spectral Method

The pseudo-spectral method, developed by De Vogeleer [De Vogeleer, 2008], uses three shaping functions to model the low-thrust trajectory in cylindrical coordinates (r,θ,z).

De Vogeleer uses power series for the shaping functions because of their simplicity and their ability to represent trigonometric functions. In addition, the polar angle θ was chosen as independent variable for shaping r and z and time t for shaping θ. The shaping functions for r, θ and z can be expressed as:

$$r = \sum_{i=0}^{k} a_i \theta^i, \quad \theta = \sum_{i=0}^{l} b_i t^i, \quad z = \sum_{i=0}^{m} c_i \theta^i \tag{4.23}$$

where a_i, b_i and c_i are the coefficients for the expansions of r, θ and z, respectively. The coordinates r, θ, z and t shown in Eqs. (4.23) are not the physical coordinates, but normalized ones. The normalization has been applied to improve the accuracy of computations.

The power series are not of infinite length, but are truncated at an arbitrary order k, l and m for r, θ and z, respectively. For each series an order of at least three is required to satisfy the boundary conditions for initial and final position and velocity. In case k, l and m are equal to three, the first four coefficients of each shaping function are fully determined by the boundary conditions for the initial and final position and velocity. If the order of the series is larger than three then the third and fourth coefficients also depend on the remaining free $k-3$, $l-3$ and $m-3$ coefficients. The boundary conditions can simply be satisfied by solving a linear system of equations.

The required thrust acceleration can be computed by differentiating the functions for r, θ and z with respect to time and substituting them into the equations of motion for cylindrical coordinates Eq. (4.8).

The values of the free coefficients are determined during an optimization in which the trajectory is optimized for minimum ΔV or minimum propellant mass. In addition, a penalty function may be implemented for exceeding the maximum thrust level. As an initial guess for the optimization, the higher-order coefficients are taken equal to zero and the first four coefficients are determined using the boundary conditions.

Tests The pseudo-spectral method has been tested for eight different test cases. For all test cases the departure and arrival date were fixed. The trajectories were optimized using two different optimizers and for either minimum ΔV or minimum propellant mass.

For 6 of the 8 test cases the pseudo-spectral method found very good near-optimal solutions. However, for two cases which involved more than two revolutions the method failed to find feasible trajectories. This problem is due to the misfit of the r- and z-components to the oscillating trend which is required to fly many-revolution trajectories efficiently.

Discussion A major strength of the pseudo-spectral shaping method is that it is able to obtain near-optimal three-dimensional low-thrust trajectories which satisfy the time of flight as well as the position and velocity constraints.

A drawback of the method is, however, that a very large number of free parameters needs to be optimized, which results in a complex optimization problem. Furthermore, the method is not capable of dealing with many-revolution trajectories.

Pros

- Three-dimensional
- Rendezvous possible
- High flexibility due to free parameters
- Time of flight constraint is satisfied exactly
- Maximum thrust acceleration constraint can be satisfied by applying a penalty function

Cons

- Very large number of free parameters need to be optimized
- High-order power series are needed to represent periodic behavior
- Not suitable for many revolutions

4.6. Comparison

Now that various shape-based methods have been discussed, they can be compared to clearly see their strengths and weaknesses. In Table 4.1 an overview is shown of the characteristics of the different methods.

From Table 4.1 it is clear that of all shape-based methods only the pseudo-spectral method is able to satisfy all boundary conditions directly. The other methods can either not satisfy a specific boundary condition or can only satisfy a boundary condition in an iterative way or by constraint optimization.

Regarding a possible thrust-level limit, none of the shape-based methods is able to obey such a limit directly, since the required thrust depends directly on the trajectory shape. A thrust limit can, however, be taken into consideration during

Table 4.1: Comparison of shape-based methods for low-thrust trajectory design; † BC satisfied iteratively, ‡ using constraint optimization, * only valid for small inclination changes, * unknown.

Method	Position BC	Velocity BC	Time of flight BC	3D	Many revolutions	Thrust limit
Exposin	Yes	No	Yes	No	Ign*	No
Inverse polynomial	Yes	Yes	Yes†	Yes*	No	No
Spherical	Yes	Yes	Yes†	Yes	Yes	No
Pseudo-equinoctial	Yes	Yes	Yes†‡	Yes	Yes	Yes‡
Pseudo-spectral	Yes	Yes	Yes	Yes	No	Yes‡

optimization by using a penalty function or by rejecting solution candidates which violate the limit. This is only possible for the methods with free parameters, so only for pseudo-equinoctial and pseudo-spectral shaping.

In addition, free parameters result in flexibility required to optimize the trajectory regarding ΔV or propellant mass ratio. The exponential sinusoid, inverse polynomial or spherical shaping lack free parameters and therefore the results of these methods have a limited optimality. However, it should be noticed that as the number of free parameters increases the optimization problem becomes more complex and therefore the computation time increases.

Furthermore, the inverse polynomial and pseudo-spectral method are not well able to deal with many-revolution trajectories (more than 2 revolutions). This is probably due to the type of functions they use to shape the trajectory. The exponential sinusoid, spherical and pseudo-equinoctial shapes contain periodic terms in their shaping functions, and they are therefore better able to handle multiple revolutions.

Finally, of the shaping methods discussed in this chapter, only the pseudo-equinoctial and spherical shaping method are able to cover Keplerian motion. This property is in general, however, useless, since when the trajectory shape coincides with a Keplerian arc no thrust is required. So, this property is only useful when also coasting arcs are considered. From optimal control theory it is known that bang-bang control is the optimal control solution. The application of bang-bang control results in a trajectory which consists of thrust arcs and coasting arcs. However, subsequently flying Keplerian and non-Keplerian trajectories is not possible using a single trajectory shape, since one shape cannot cover both a Keplerian and non-Keplerian arc. As a consequence, bang-bang control can not be achieved by shaping methods(!), but only approximated. Therefore, the ability to cover Keplerian motion is useless.

5. Hodographic Shaping

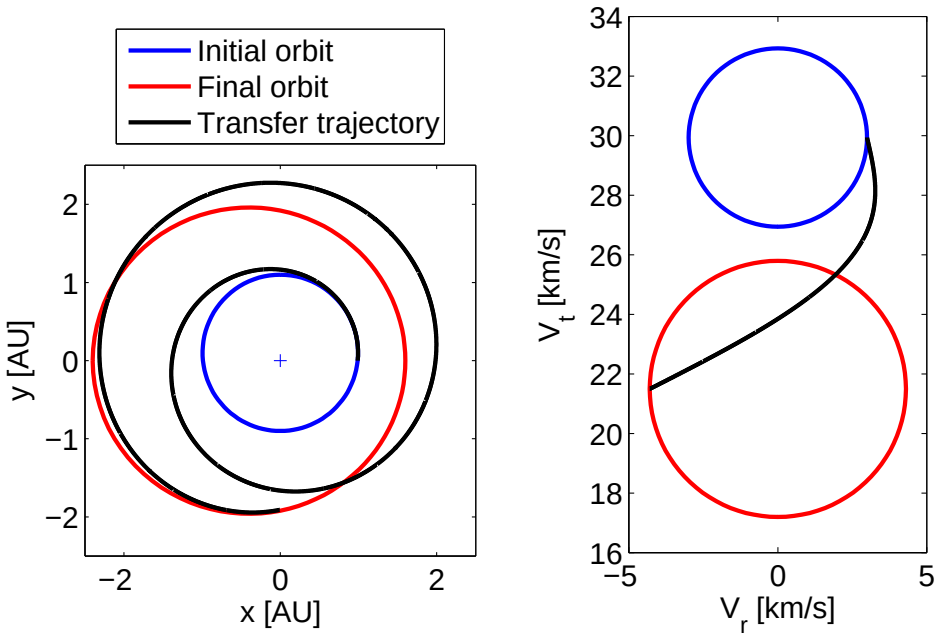
In this chapter a novel shaping method is proposed which is very simple and straightforward and which can satisfy all boundary conditions in three dimensions directly. In addition, the method is possibly suitable to deal with thrust limits and many-revolution trajectories. The method is called the *hodographic-shaping method* and is discussed in the following section.

5.1. Hodographic-Shaping Method

In contrast to the shaping methods discussed in the previous chapter, the hodographic-shaping method shapes the velocity hodograph of the transfer trajectory instead of the trajectory or its orbital elements.

5.1.1. Concept

The idea for the hodographic-shaping method originates from the velocity hodograph (see Figure 2.3). For a transfer, the velocity hodograph of the initial and final orbit can be drawn in one graph. The transfer trajectory can then be indicated in the same graph by simply drawing a line from the hodograph of the initial orbit to the hodograph of the final orbit, see Figure 5.1. By following this 'path' in the velocity domain one obtains a trajectory which fulfills the conditions on initial and final velocity.



(a) Trajectory (b) Velocity hodograph

Figure 5.1: Transfer trajectory with the initial and final orbit and corresponding velocity hodographs.

Instead of drawing a line in the velocity graph, one can define functions for the velocity components, e.g. V_r and V_θ, which match the required initial and final velocities. These functions are called *velocity functions* and express the velocity components as a function of the independent variable, e.g. time. These functions may contain various mathematical terms, as long as they are able to satisfy the boundary conditions on velocity.

Of course, also the boundary conditions on initial and final position have to be satisfied. By integrating the velocity function of each velocity component the spacecraft's position can be obtained. The required change in position can be matched by modifying the velocity functions, e.g. by changing the values of the coefficients or by adding extra terms. In this way the velocity functions are shaped, such that they satisfy both types of boundary conditions. Next to terms required to meet the boundary conditions, additional terms can be added to the velocity functions to improve their flexibility for optimization.

Since, in principle, arbitrary conditions on the initial and final position and velocity can be satisfied, the hodographic-shaping method is not limited to specific initial and final orbits. Basically, any orbit can be used as initial or final orbit.

5.1.2. Shaping methodology

For the shaping of the velocity functions one has to choose which reference frame, coordinate system and independent variable to use. The choice of these are discussed here.

Reference frame Since the hodographic-shaping method is meant for interplanetary flight a heliocentric reference is chosen. The XY-plane is chosen equal to the orbital plane of the departure body, the Z-direction along the angular momentum vector of the departure body and the X-axis points in the direction of the departure body at departure. As a result, the spacecraft coordinates at the start of the transfer are very simply.

Coordinate system For the hodographic-shaping method the use of either the Cartesian, cylindrical or spherical coordinate system is most straightforward. However, in case Cartesian velocities are used the spacecraft's velocities when orbiting the Sun will vary strongly. This is undesirable and therefore the Cartesian coordinate system is discarded.

A velocity hodograph of a Keplerian orbit is usually drawn using the radial and normal velocity components. These velocity components coincide with the velocity components used in the cylindrical and spherical coordinate system in the planar case. In addition to that, the spherical radial direction always coincides with direction from the Sun, which is beneficial since as a result the Sun's gravity term (which has a large influence on the dynamics) is only present in the radial equation

of motion. On the other hand, in the cylindrical coordinate system the in-plane (XY) and out-of-plane (Z) motion are fully decoupled, which makes the cylindrical coordinate system much more simple to use than the spherical one.

In the end, cylindrical coordinates have been selected. The main reason for this is that the cylindrical coordinate system only has one angular coordinate (θ) whereas the spherical coordinate system has two (θ, φ). It was found that the use of angular coordinates led to problems when satisfying the boundary conditions, see Section 5.2.2. Therefore, it was chosen to use the coordinate system with the least number of angular coordinates, which is the cylindrical coordinate system.

Independent variable For the shaping of the velocity functions in cylindrical coordinates it is most straightforward to shape either as a function of time t or as function of the polar angle θ (the second cylindrical coordinate). These variables, in general, increase monotonically when orbiting the Sun. On the one hand, shaping as a function of the polar angle has the advantage that it is correlated to the geometry. This is useful considering e.g. the periodic behavior of an inclined orbit. On the other hand, shaping as a function of time has the advantage that it is more related to the actual time rate of change of the orbit. For example, the time rate of change of the velocity is the acceleration whereas the polar-angle rate of change of the velocity can not be considered as a useful motion-related quantity. Therefore, shaping velocity as a function of time is more related to the real dynamics than shaping as a function of polar angle.

Since it is not known beforehand whether shaping as function of time or as function of polar angle is more suitable, two different methods have been developed. Both methods are discussed in the following. Hodographic shaping as a function of time is discussed first, since it is the most straightforward one.

5.2. Hodographic Shaping as Function of Time

Procedure The following procedure describes the application of hodographic shaping as a function of time in three dimensions:

1. Pick the departure and arrival body.
2. Pick the departure date and time of flight (or arrival date).
3. Compute the corresponding initial and final position and velocity (e.g. using ephemerides).
4. Define the radial, normal and axial velocity during the transfer as a function of time, i.e. $V_r(t) = ...$, $V_\theta(t) = ...$ and $V_z(t) = ...$, respectively.
5. Pick values for the coefficients in the velocity functions which can be chosen freely.

6. Compute the values of some of the coefficients in the velocity functions to satisfy the boundary conditions on the initial and final velocities.
7. Analytically integrate the radial and axial velocity over time to obtain the radial distance r and axial distance z, respectively, as a function of time. Use the results to adjust the radial and axial velocity functions to satisfy the boundary condition on final radial and axial position.
8. Integrate the angular velocity ($\dot{\theta} = V_\theta / r$) to obtain the change in polar angle θ. The polar angle at $t = \text{TOF}$ can be used to adjust the normal velocity function in order to meet the boundary condition on final polar angle.
9. Compute the thrust profile, required to fly the trajectory, using the equations of motion.
10. Compute the ΔV, required to fly the trajectory, by integrating the thrust acceleration over time.

Once the thrust profile and ΔV are known, the characteristics of the specific trajectory are known and it can be compared with other possible solutions. In addition, if needed, an optimization can be done.

Calculations To obtain the polar angle θ and required ΔV, two integrals have to be computed which can not be solved analytically. As a consequence these integrals have to be computed numerically. In the following the calculations are discussed which are required to obtain the trajectory, thrust profile and ΔV.

At each integration step i, first the time is determined as follows:

$$t(i) = t(i-1) + h \tag{5.1}$$

where h is the step-size of the integration. The velocities are then obtained by substituting the time $t(i)$ in the velocity functions:

$$V_r(i) = V_r\left(t(i)\right) \tag{5.2}$$
$$V_\theta(i) = V_\theta\left(t(i)\right) \tag{5.3}$$
$$V_z(i) = V_z\left(t(i)\right) \tag{5.4}$$

In addition, the radial and axial distance are obtained by analytically integrating the radial and axial velocity, respectively:

$$r(i) = r_0 + \int_0^{t(i)} V_r dt \tag{5.5}$$
$$z(i) = z_0 + \int_0^{t(i)} V_z dt \tag{5.6}$$

Furthermore, the angular velocity is obtained by dividing the normal velocity by the radial distance:

$$\dot{\theta}(i) = \frac{V_\theta(i)}{r(i)} \tag{5.7}$$

Subsequently, the polar angle can be computed by numerically integrating the angular velocity:

$$\theta(i) = \theta_0 + \int_0^{t(i)} \dot{\theta} dt = \theta(i-1) + h \cdot \Phi_\theta \tag{5.8}$$

where Φ_θ is the increment function for θ which is a function of $\dot{\theta}$.

In cylindrical coordinates the equations of motion are given by Eqs. (4.2) and (4.3) as follows:

$$\ddot{r} - r\dot{\theta}^2 + \frac{\mu}{s^3} r = f_r \tag{5.9}$$

$$r\ddot{\theta} + 2\dot{r}\dot{\theta} = f_\theta \tag{5.10}$$

$$\ddot{z} + \frac{\mu}{s^3} z = f_z \tag{5.11}$$

where $s = \sqrt{r^2 + z^2}$ is the distance from the Sun. In addition, one can write $\ddot{r} = \dot{V}_r$, $\ddot{z} = \dot{V}_z$ and:

$$\ddot{\theta} = \frac{d}{dt}\dot{\theta} = \frac{d}{dt}\left(\frac{V_\theta}{r}\right) = \frac{\dot{V}_\theta}{r} - \frac{V_\theta V_r}{r^2} \tag{5.12}$$

where $\dot{V}_r$, $\dot{V}_\theta$ and $\dot{V}_r$ can simply be obtained by analytically differentiating the velocity functions V_r, V_θ and V_z, respectively. Using these expressions, the equations of motions can be rewritten to obtain the components of required thrust acceleration at each integration step, as follows:

$$f_r = \dot{V}_r - \frac{V_\theta^2}{r} + \frac{\mu}{s^3} r \tag{5.13}$$

$$f_\theta = \dot{V}_\theta + \frac{V_\theta V_r}{r} \tag{5.14}$$

$$f_z = \dot{V}_z + \frac{\mu}{s^3} z \tag{5.15}$$

The magnitude of the thrust acceleration is then computed using Pythagoras:

$$f(i) = \sqrt{f_r(i)^2 + f_\theta(i)^2 + f_z(i)^2} \tag{5.16}$$

Finally, the required ΔV is obtained by numerically integrating the thrust acceleration over time:

$$\Delta V(i) = \int_0^{t(i)} f dt = \Delta V(i-1) + h \cdot \Phi_{\Delta V} \tag{5.17}$$

where $\Phi_{\Delta V}$ is the increment function for ΔV which is a function of f.

In order to be able to integrate the velocity functions analytically the functions need to be simple. It is therefore chosen to compose the velocity functions of a sum of polynomial, trigonometric and exponential terms. In addition, also multiplications of polynomial or exponential terms with trigonometric terms can be used, e.g. $t\cos t$ or $e^t \sin t$. Even more complex terms can be used, as long as they are analytically integrable.

A velocity function then consists of a sum of simple base functions, v_i, multiplied by their coefficients, c_i:

$$V(t) = \sum_{i=1}^{n} c_i v_i(t) \tag{5.18}$$

or in vector form:

$$V(t) = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_n \end{bmatrix} \bullet \begin{bmatrix} v_1(t) \\ v_2(t) \\ v_3(t) \\ \vdots \\ v_n(t) \end{bmatrix} \tag{5.19}$$

The minimum number of base functions required per velocity function is equal to the number of boundary conditions which the velocity function has to satisfy.

5.2.1. Boundary conditions

Each velocity function has to satisfy three boundary conditions, namely on initial and final velocity, V_0 and V_f, and on the change in position, $P_f - P_0$, i.e. the difference between the initial and final position. In total nine boundary conditions need to be solved:

$$\begin{aligned} &V_r(0) = V_{r,0}, \quad V_r(t_f) = V_{r,f}, \quad \int_0^{t_f} V_r dt = r_f - r_0, \\ &V_\theta(0) = V_{\theta,0}, \quad V_\theta(t_f) = V_{\theta,f}, \quad \int_0^{t_f} \frac{V_\theta}{r} dt = \theta_f = \psi + 2\pi N, \\ &V_z(0) = V_{z,0}, \quad V_z(t_f) = V_{z,f}, \quad \int_0^{t_f} V_z dt = z_f - z_0 \end{aligned} \tag{5.20}$$

where $\psi \in [0, 2\pi)$ is the transfer angle and $N = 0, 1, 2...$ is the number of revolutions. Since per velocity function three boundary conditions need to be solved, the

first three coefficients of each velocity functions are used to satisfy the boundary conditions. The conditions on velocity can be fulfilled by substituting $t = 0$ and $t = t_f$ into the velocity function as follows:

$$V(0) = c_1 v_1(0) + c_2 v_2(0) + c_3 v_3(0) + \sum_{i=4}^{n} c_i v_i(0) = V_0 \tag{5.21}$$

$$V(t_f) = c_1 v_1(t_f) + c_2 v_2(t_f) + c_3 v_3(t_f) + \sum_{i=4}^{n} c_i v_i(t_f) = V_f \tag{5.22}$$

and subsequently solving the equations.

The condition on position is satisfied in a different way. The change in position can be computed by integrating a velocity function over the interval $t = 0$ to $t = t_f$. This change in position has to be equal to the difference between the initial and final position, which can be expressed for the general case as follows:

$$\begin{aligned} \int_0^{t_f} V dt &= \tilde{V}(t_f) - \tilde{V}(0) \\ &= c_1[\tilde{v}_1(t_f) - \tilde{v}_1(0)] + c_2[\tilde{v}_2(t_f) - \tilde{v}_2(0)] + c_3[\tilde{v}_3(t_f) - \tilde{v}_3(0)] \\ &\quad + \sum_{i=4}^{n} c_i[\tilde{v}_i(t_f) - \tilde{v}_i(0)] \\ &= P_f - P_0 \end{aligned} \tag{5.23}$$

where the tilde indicates the integral with respect to time.

One would expect that two boundary conditions on position have to be satisfied, namely on the initial and on the final position. These two conditions are, however, reduced to one as the integration constant C takes care of the initial position as follows:

$$P(0) = \int_0^0 V dt = \tilde{V}(0) - \tilde{V}(0) + C = C = P_0 \tag{5.24}$$

Solving Eq. (5.23) is therefore sufficient to satisfy both position boundary conditions. Since the integration constant is not part of a velocity function, it is not of interest and may therefore be ignored. Consequently, only Eq. (5.23) has to be considered to satisfy the position boundary conditions.

So, for each velocity function three boundary conditions have to be satisfied, namely Eqs. (5.21), (5.22) and (5.23). The system of equations which needs

to solved can be written in matrix form as follows:

$$\begin{bmatrix} v_1(0) & v_2(0) & v_3(0) \\ v_1(t_f) & v_2(t_f) & v_3(t_f) \\ \tilde{v}_1(t_f) - \tilde{v}_1(0) & \tilde{v}_2(t_f) - \tilde{v}_2(0) & \tilde{v}_3(t_f) - \tilde{v}_3(0) \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} V_0 - \sum_{i=4}^{n} c_i v_i(0) \\ V_f - \sum_{i=4}^{n} c_i v_i(t_f) \\ P_f - P_0 - \sum_{i=4}^{n} c_i [\tilde{v}_i(t_f) - \tilde{v}_i(0)] \end{bmatrix} \tag{5.25}$$

where one solves for the coefficients c_1 to c_3. As long as the matrix in Eq. (5.25) is invertible, Eq. (5.25) can be solved in closed form. Note that this needs to be done for each velocity component separately, i.e. for the radial, normal and axial component.

5.2.2. Final polar angle condition

For the normal direction, the position is expressed by the polar angle θ. However, if one applies Eq. (5.23) then one computes the position change as a linear distance and not as an angular distance. So, to obtain the angular position change, one needs to integrate the angular velocity over time instead of the normal velocity. This integral can in general, however, not be computed analytically and therefore the boundary condition on position needs to be solved in an iterative way.

The boundary conditions on the normal velocity can, on the other hand, still be solved in closed form. In matrix form, the system of equations which needs to be solved then becomes:

$$\begin{bmatrix} v_1(0) & v_2(0) \\ v_1(t_f) & v_2(t_f) \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} V_0 - \sum_{i=3}^{n} c_i v_i(0) \\ V_f - \sum_{i=3}^{n} c_i v_i(t_f) \end{bmatrix} \tag{5.26}$$

Satisfying the final polar angle condition is done by varying the value of one of the coefficients of the normal-velocity function using a root finder until the condition on final polar angle is satisfied while solving Eq. (5.26) simultaneously. However, since this is an iterative process it requires a large computational effort and it is inaccurate with respect to solving the boundary conditions in closed form.

Fortunately, it was found that the final polar angle condition can be solved exactly without iteration, because the velocity functions are summations of linear terms. This is done as follows.

The following condition for the final polar angle has to be satisfied:

$$\int_0^{t_f} \dot{\theta} dt = \int_0^{t_f} \frac{V_\theta}{r} dt = \theta_f \tag{5.27}$$

By writing out the normal velocity function this equation can be written as:

$$\int_0^{t_f} \frac{c_1 v_1 + c_2 v_2 + c_3 v_3 + \sum_{i=4}^{n} c_i v_i}{r} dt = \theta_f \tag{5.28}$$

Next to this boundary condition on the polar angle, two boundary conditions exist on the normal velocity itself which can be solved using Eq. (5.26). The first and second coefficient of the normal-velocity function, c_1 and c_2, are used to satisfy the velocity boundary conditions, and the third coefficient c_3 is used to solve the final polar angle condition. Eq. (5.26) is now written with the terms involving c_3 separately:

$$\begin{bmatrix} v_1(0) & v_2(0) \\ v_1(t_f) & v_2(t_f) \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} -c_3 v_3(0) \\ -c_3 v_3(t_f) \end{bmatrix} + \begin{bmatrix} V_0 - \sum_{i=4}^{n} c_i v_i(0) \\ V_f - \sum_{i=4}^{n} c_i v_i(t_f) \end{bmatrix} \tag{5.29}$$

Solving this equation for c_1 and c_2 results in:

$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = c_3 \begin{bmatrix} v_1(0) & v_2(0) \\ v_1(t_f) & v_2(t_f) \end{bmatrix}^{-1} \begin{bmatrix} -v_3(0) \\ -v_3(t_f) \end{bmatrix} + \begin{bmatrix} v_1(0) & v_2(0) \\ v_1(t_f) & v_2(t_f) \end{bmatrix}^{-1} \begin{bmatrix} V_0 - \sum_{i=4}^{n} c_i v_i(0) \\ V_f - \sum_{i=4}^{n} c_i v_i(t_f) \end{bmatrix} \tag{5.30}$$

These equations can be expressed in more simple form by replacing the matrix-vector multiplications by the resulting vectors.

$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = c_3 \begin{bmatrix} K_1 \\ K_2 \end{bmatrix} + \begin{bmatrix} K_3 \\ K_4 \end{bmatrix} \tag{5.31}$$

where K_1 to K_4 are the coefficients of the resulting vectors. These solutions for c_1 and c_2 are then substituted into Eq. (5.28) to obtain an equation in which c_3 is the only unknown:

$$\int_0^{t_f} \frac{c_3 K_1 v_1 + K_3 v_1 + c_3 K_2 v_2 + K_4 v_2 + c_3 v_3 + \sum_{i=4}^{n} c_i v_i}{r} dt = \theta_f \tag{5.32}$$

Subsequently, the terms involving c_3 are written separately:

$$c_3 \int_0^{t_f} \frac{K_1 v_1 + K_2 v_2 + v_3}{r} dt + \int_0^{t_f} \frac{K_3 v_1 + K_4 v_2 + \sum_{i=4}^{n} c_i v_i}{r} dt = \theta_f \tag{5.33}$$

and finally the value for c_3 which satisfies the final polar angle condition can be computed as follows:

$$c_3 = \frac{\theta_f - \int_0^{t_f} \frac{K_3 v_1 + K_4 v_2 + \sum_{i=4}^{n} c_i v_i}{r} dt}{\int_0^{t_f} \frac{K_1 v_1 + K_2 v_2 + v_3}{r} dt} \tag{5.34}$$

The boundary conditions applicable to the normal velocity function can now be solved using Eqs. (5.30) and (5.34). As a result, the computational effort required to solve the boundary conditions is little and the conditions are solved exactly.

Angular velocity shaping Another solution to satisfy the final polar angle condition is to shape the angular velocity: $\dot{\theta} = \dot{\theta}(t)$, instead of the normal velocity V_θ. If a simple linear function is used to shape the angular velocity, then the polar angle θ can be obtained analytically just like the radial and axial distance. As a result one is able to satisfy the condition on final polar angle straightforwardly by solving Eq. (5.23).

A drawback of shaping the angular velocity $\dot{\theta}$ is that it proves to be harder to find a good shaping function than for the normal velocity. This may be due to the fact that changing the angular velocity is related to the radial distance and changing the normal velocity is not. For example, changing the normal velocity with 1 m/s requires approximately the same ΔV at any radial distance. However, changing the angular velocity with a specific $\Delta\dot{\theta}$ requires a totally different ΔV at a small radial distance than at a large radial distance.

Therefore it is chosen to shape the normal velocity and not the angular velocity and thus Eqs. (5.34) and (5.30) are used to satisfy the boundary conditions in the normal direction.

5.3. Hodographic Shaping as Function of Polar Angle

Instead of shaping the velocities as a function of time, one can also use the polar angle as independent variable. In that case, the velocities have to be analytically integrable over the polar angle instead of over time to obtain the change in position. This is only possible when the velocities themselves are derivatives with respect to the polar angle, $\frac{d}{d\theta}$, instead of derivatives with respect to time, $\frac{d}{dt}$. This is possible for the radial and axial direction; however, not for the normal direction, since the normal coordinate is θ which is the independent variable. Therefore, instead of shaping the normal velocity, the time evolution is shaped. The "velocity" functions then become:

$$r' = \frac{dr}{d\theta} = R(\theta) \tag{5.35}$$

$$t' = \frac{dt}{d\theta} = T(\theta) \tag{5.36}$$

$$z' = \frac{dz}{d\theta} = Z(\theta) \tag{5.37}$$

where the apostrophe indicates differentiating with respect to θ.

Instead of stating that one shapes the time evolution, one could also say that one shapes the angular velocity:

$$\dot{\theta} = \frac{d\theta}{dt} = \frac{1}{T(\theta)} \tag{5.38}$$

In the remainder of this report, the functions R and Z are called the radial and axial-velocity function, respectively (since they represent derivatives of traveled distance), and T is called the time-evolution function. The true radial and axial velocities can be computed as follows:

$$\dot{r} = \frac{dr}{dt} = \frac{dr}{d\theta}\frac{d\theta}{dt} = \frac{R(\theta)}{T(\theta)} \tag{5.39}$$

$$\dot{z} = \frac{dz}{dt} = \frac{dz}{d\theta}\frac{d\theta}{dt} = \frac{Z(\theta)}{T(\theta)} \tag{5.40}$$

Furthermore, the radial and axial distances are obtained by analytically integrating Eqs. (5.35) and (5.37), respectively, over the polar angle:

$$r(\theta) = r_0 + \int R(\theta)d\theta \tag{5.41}$$

$$z(\theta) = z_0 + \int Z(\theta)d\theta \tag{5.42}$$

To compute the required thrust acceleration, the accelerations in all three directions need to be known. These accelerations can be computed as follows:

$$\ddot{r} = \frac{d}{dt}\left(\frac{dr}{dt}\right) = \frac{d}{dt}\left(\frac{dr}{d\theta}\frac{d\theta}{dt}\right) = \frac{d^2r}{d\theta^2}\left(\frac{d\theta}{dt}\right)^2 + \frac{dr}{d\theta}\frac{d^2\theta}{dt^2} = r''\dot{\theta}^2 + r'\ddot{\theta} \tag{5.43}$$

$$\ddot{\theta} = \frac{d}{dt}\left(\frac{d\theta}{dt}\right) = \frac{d}{dt}\left(\frac{1}{\frac{dt}{d\theta}}\right) = -\frac{\frac{d^2t}{d\theta^2}\frac{d\theta}{dt}}{\left(\frac{dt}{d\theta}\right)^2} = \frac{-t''}{t'^3} \tag{5.44}$$

$$\ddot{z} = \frac{d}{dt}\left(\frac{dz}{dt}\right) = \frac{d}{dt}\left(\frac{dz}{d\theta}\frac{d\theta}{dt}\right) = \frac{d^2z}{d\theta^2}\left(\frac{d\theta}{dt}\right)^2 + \frac{dz}{d\theta}\frac{d^2\theta}{dt^2} = z''\dot{\theta}^2 + z'\ddot{\theta} \tag{5.45}$$

By substituting these equations into the equations of motion (Eqs. (5.9) to (5.11)) the thrust acceleration components can be computed as follows:

$$f_r = (r'' - r)\dot{\theta}^2 + r'\ddot{\theta} + \frac{\mu}{s^3}r \tag{5.46}$$

$$f_\theta = r\ddot{\theta} + 2r'\dot{\theta}^2 \tag{5.47}$$

$$f_z = z''\dot{\theta}^2 + z'\ddot{\theta} + \frac{\mu}{s^3}z \tag{5.48}$$

or in terms of the velocity functions R, T and Z:

$$f_r = \frac{1}{T^2}\left(\frac{dR}{d\theta} - r - \frac{R}{T}\frac{dT}{d\theta}\right) + \frac{\mu}{s^3}r \tag{5.49}$$

$$f_\theta = \frac{1}{T^2}\left(2R - \frac{r}{T}\frac{dT}{d\theta}\right) \tag{5.50}$$

$$f_z = \frac{1}{T^2}\left(\frac{dZ}{d\theta} - \frac{Z}{T}\frac{dT}{d\theta}\right) + \frac{\mu}{s^3}z \tag{5.51}$$

Finally, the required ΔV is computed by integrating the thrust acceleration:

$$\Delta V = \int f dt = \int f|\frac{dt}{d\theta}|d\theta = \int f|T|d\theta \tag{5.52}$$

In this equation the absolute value of T has to be used, since the ΔV always increases independent of the sign of the angular velocity. In addition, the elapsed time is found by analytically integrating Eq. (5.36):

$$\Delta t(\theta) = \int |T(\theta)|d\theta \tag{5.53}$$

Since the time always increases independent of the sign of T, the absolute value of T has to be used.

Actually, the evolution T always has to be positive since time always increases. However, since T is simply the inverse of the angular velocity it may become negative; which means that the rotation of the spacecraft about the Sun is reversed. Such a motion is undesirable, since it is very inefficient and since it gives problems for the computation of the elapsed time Eq. (5.53). Fortunately, trajectories which require T to change sign require a huge ΔV: when T changes sign, the angular velocity becomes singular and therefore the thrust acceleration components too, see Eqs. (5.38), (5.49) to (5.51), which all contain the term $1/T$. So, when T becomes negative the trajectory will require an unfeasibly high ΔV and therefore these trajectories will be filtered out during optimization. The time evolution function T and Eqs. (5.49) and (5.53) can thus be used as long as the optimal trajectory does not require T to become negative.

Boundary conditions The boundary conditions which need to be satisfied are the following:

$$\begin{aligned} &R(0) = V_{r,0}/\dot{\theta}_0, \quad R(t_f) = V_{r,f}/\dot{\theta}_f, \quad \int_0^{\theta_f} V_r d\theta = r_f - r_0, \\ &T(0) = 1/\dot{\theta}_0, \qquad T(t_f) = 1/\dot{\theta}_f, \qquad \int_0^{\theta_f} T d\theta = t_f = TOF, \\ &Z(0) = V_{z,0}/\dot{\theta}_0, \quad Z(t_f) = V_{z,f}/\dot{\theta}_f, \quad \int_0^{\theta_f} V_z d\theta = z_f - z_0 \end{aligned} \tag{5.54}$$

These conditions can be solved similar to the time-driven method as follows:

$$\begin{aligned} &\begin{bmatrix} v_1(0) & v_2(0) & v_3(0) \\ v_1(\theta_f) & v_2(\theta_f) & v_3(\theta_f) \\ \hat{v}_1(\theta_f) - \hat{v}_1(0) & \hat{v}_2(\theta_f) - \hat{v}_2(0) & \hat{v}_3(\theta_f) - \hat{v}_3(0) \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} \\ &\qquad = \begin{bmatrix} V_0 - \sum_{i=4}^{n} c_i v_i(0) \\ V_f - \sum_{i=4}^{n} c_i v_i(\theta_f) \\ P_f - P_0 - \sum_{i=4}^{n} c_i [\hat{v}_i(\theta_f) - \hat{v}_i(0)] \end{bmatrix} \end{aligned} \tag{5.55}$$

where the hat symbol indicates the integral with respect to the polar angle.

As discussed, in these equations the velocities V and v are not time rates of change of position, but polar-angle rates of change of position.

5.4. Velocity Functions

The velocity functions are built up as a sum of simple mathematical functions which can be differentiated and integrated analytically. These simple functions are called base functions, and they can for example be a sine, cosine, power, exponential or multiplication of a power with a sine or cosine term, as long as they are analytically integrable. These base functions are used for both shaping as a function of time and shaping as a function of the polar angle.

For the base functions it was found useful to scale the driving parameters t and θ by dividing them by their final values, TOF and θ_f, respectively. The ranges of t/TOF and θ/θ_f are by definition $[0, 1]$. The benefit of this scaling is that as a consequence the magnitude of the different base functions is similar. As a result the ranges of the coefficients corresponding to the base functions are similar, which simplifies the optimization of the free coefficients.

An exception for this scaling is made for trigonometric base functions. In their standard form, trigonometric functions are 2π-periodic. So, by using the unscaled polar angle for trigonometric functions in the polar-angle-driven method, these functions become one-revolution periodic. This is beneficial since often in orbital mechanics periodic phenomena are related to the geometry.

Table 5.1: Base functions and their derivatives and integrals, where u can be either t or θ.

Base function	$v(u)$	$\frac{dv}{du}$	$\int v du$
Constant	1	0	u
Power	u^n	nu^{n-1}	$\frac{1}{n+1}u^{n+1}$
Exponential	e^{nu}	ne^{nu}	$\frac{1}{n}e^{nu}$
Sine	$\sin(2\pi nu)$	$2\pi n\cos(2\pi nu)$	$-\frac{1}{2\pi n}\cos(2\pi nu)$
Cosine	$\cos(2\pi nu)$	$-2\pi n\sin(2\pi nu)$	$\frac{1}{2\pi n}\sin(2\pi nu)$
Power times sine	$u^n\sin(2\pi mu)$	Eq. (5.62)	Eq. (5.63)
Power times cosine	$u^n\cos(2\pi mu)$	Eq. (5.65)	Eq. (5.66)
Exponential times sine	$e^{nu}\sin(2\pi mu)$	Eq. (5.68)	Eq. (5.69)
Exponential times cosine	$e^{nu}\cos(2\pi mu)$	Eq. (5.71)	Eq. (5.72)

The elapsed time is, on the other hand, not related to the geometry. For the time-driven method it was chosen to scale the time by a factor $2\pi n/TOF$ for trigonometric functions to make these functions $1/n$-periodic instead of 2π-periodic. The trigonometric functions then complete n full periods as time runs from 0 to TOF.

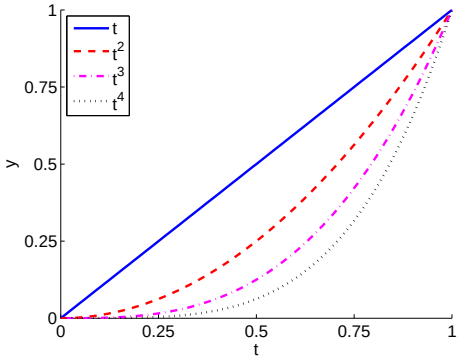
5.4.1. Base functions

The base functions which have been applied in velocity functions in this research are the following: a constant, sine, cosine, power, exponential, power times sine or cosine and exponential times sine or cosine, see Table 5.1. The behavior of some of these base functions is shown in Figure 5.2. In addition, the characteristics of these base functions are discussed in the following. For these discussions it is assumed that time is the independent variable, i.e. $u \to t$. However, most considerations are also valid when the polar angle is the independent variable.

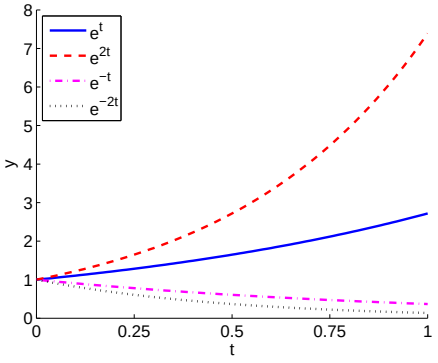
Constant The constant base function simply equals the value 1. As a consequence the value of the corresponding term is fully determined by the value of its coefficient. The constant base function is very often applied in velocity functions, since it is very useful for satisfying the initial velocity condition. The constant base function, its derivative and its integral are shown in Table 5.1.

Power The power base function is simply a power term: $v_{\text{pow}}(t) = t^n$, where n is the order of the power and $n \geq 1$. The order is not allowed be lower than 1 since else the function or its derivative becomes singular when $t = 0$. The derivative and integral of a power base function can be found in Table 5.1.

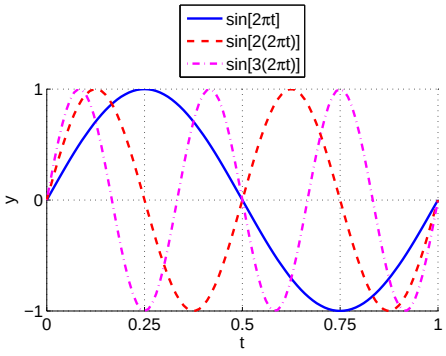
Power base functions are very suitable for dealing with velocity and position changes over time, since their values at $t = 0$ and $t = \text{TOF}$ differ.



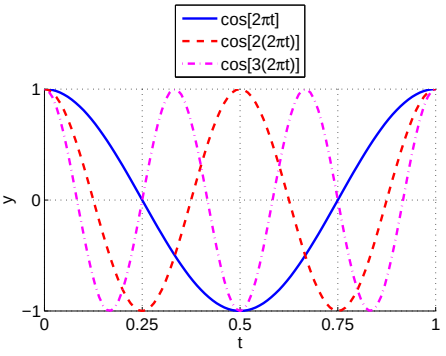
(a) Power



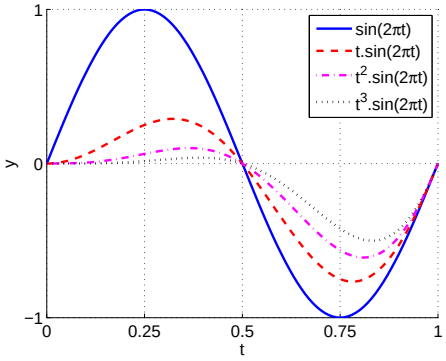
(b) Exponential



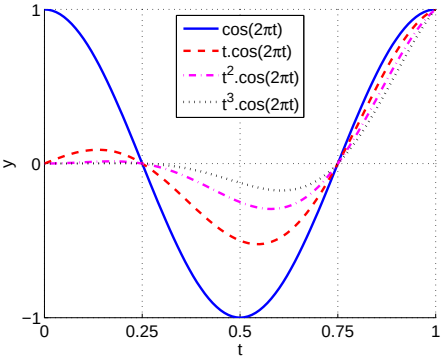
(c) Sine



(d) Cosine



(e) PowerSine



(f) PowerCosine

Figure 5.2: Behavior of six base functions.

Exponential Just like power functions, exponential terms (e^{nt}) can take care of velocity and position changes. However, in contrast to power terms the value of exponential base functions is 1 instead of 0 at $t = 0$ and furthermore the behavior of exponential terms over time differs from that of power terms. Therefore exponential functions might be more suitable for hodographic shaping than power functions. The derivative and integral of an exponential base function can be found in Table 5.1.

Trigonometric functions The variation of the velocity along a Keplerian orbit is periodic, see Eqs. (2.16) and (2.17). Furthermore, along transfer trajectories often periodic trends can be observed [De Pascale and Vasile, 2006]. Such periodic features of velocity functions can be generate by trigonometric functions like sines and cosines. As discussed time is scaled by $2\pi n/TOF$ and the polar angle is not scaled in trigonometric base functions. In the following the term $2\pi n$ is shown in trigonometric base functions to indicate the scaling of time.

By applying sine and cosine terms with different periodicities, multiplications of trigonometric functions can be simulated. Examples are the following trigonometric formulas:

$$\sin(2u) = 2\sin u \cos u \tag{5.56}$$

$$\cos(2u) = 2\cos^2 u - 1 \tag{5.57}$$

$$= 1 - 2\sin^2 u \tag{5.58}$$

In a similar way also higher-order trigonometric functions and multiplications can be expressed as the sum of sine and cosine terms with different frequencies. One could even construct a Fourier series which in principle can represent any function.

Sine The sine base function is a simple sine term with periodicity $\frac{1}{n}$:

$$v_{\sin}(t) = \sin(2\pi n t) \tag{5.59}$$

where $n > 0$.

Cosine The cosine base function is similar to the sine base function:

$$v_{\cos}(t) = \cos(2\pi n t) \tag{5.60}$$

where $n > 0$. The derivatives and integrals of the sine and cosine base functions can be found in Table 5.1.

The sine and cosine base function are both $\frac{1}{n}$-periodic. In case $n = 1, 2, 3...$, their integral over the normalized time equals zero and consequently they do not

contribute to the change in position. The initial values of the sine and cosine base functions equal 0 and 1, respectively. Therefore, a sine does not contribute to the initial velocity, whereas a cosine does. When n is not an integer the sine and cosine base functions do contribute to the change in position and their initial and final values differ. This is especially the case when $n = \frac{1}{2}, 1, \frac{3}{2}, 2....$ Finally, the derivatives of sine and cosine functions at $t = 0$ are equal to 1 and 0, respectively. Therefore, a sine base function contributes to a change in velocity, i.e. acceleration, at departure, whereas a cosine does not.

Power times trigonometric The characteristics of the periodic variation of the velocity may change along a transfer trajectory. Therefore it can be desirable to change the effect of periodic terms over time. This can be achieved by changing the amplitude by multiplying power or exponential functions with trigonometric functions.

Power times sine The power-times-sine base function is as follows:

$$v_{\text{powsin}}(t) = t^n \sin(2\pi m t) \tag{5.61}$$

where $n \geq 1$ and $m \geq 1$.

The derivative is computed by applying the product rule, resulting in the following equation:

$$\dot{v}_{\text{powsin}}(t) = \frac{dv}{dt} = nt^{n-1} \sin(2\pi m t) + t^n 2\pi m \cos(2\pi m t) \tag{5.62}$$

The integral of the base function can be computed analytically by applying integration by parts. The result of integration by parts is the following:

$$\begin{aligned}\tilde{v}_{\text{powsin}}(t) = \int v dt = &\sum_{i=1,3,5..n} (-1)^{(i+1)/2} \frac{n!}{(n+1-i)!} t^{n+1-i} \left(\frac{1}{2\pi m}\right)^i \cos(2\pi m t) \\ &+ \sum_{j=2,4,6..n} (-1)^{(j+2)/2} \frac{n!}{(n+1-j)!} t^{n+1-j} \left(\frac{1}{2\pi m}\right)^j \sin(2\pi m t)\end{aligned} \tag{5.63}$$

Power times cosine The power-times-cosine base function is expressed as:

$$v_{\text{powcos}}(t) = t^n \cos(2\pi m t) \tag{5.64}$$

where $n \geq 1$ and $m \geq 1$.

The derivative is computed as follows:

$$\dot{v}_{\text{powcos}}(t) = \frac{dv}{dt} = nt^{n-1}\cos(2\pi mt) - t^n 2\pi m \sin(2\pi mt) \tag{5.65}$$

Again the integral is computed using integration by parts:

$$\begin{aligned}\tilde{v}_{\text{powcos}}(t) = \int v dt = &\sum_{i=1,3,5..n} (-1)^{(i-1)/2} \frac{n!}{(n+1-i)!} t^{n+1-i} \left(\frac{1}{2\pi m}\right)^i \sin(2\pi mt) \\ &+ \sum_{j=2,4,6..n} (-1)^{(j+2)/2} \frac{n!}{(n+1-j)!} t^{n+1-j} \left(\frac{1}{2\pi m}\right)^j \cos(2\pi mt)\end{aligned} \tag{5.66}$$

Calculating the integral of a power times trigonometric function is computationally expensive, since it requires calculating the values of two series. So regarding CPU time it is better not to use (higher-order) power-times-trigonometric functions.

In contrast to cosine functions the initial value of power-times-cosine terms is zero, and therefore power-times-cosine terms do not contribute to the initial velocity. On the other hand, power-times-cosine terms can satisfy final velocity conditions since their initial and final values differ. Furthermore, the integral of power-times-trigonometric functions over time is usually not zero in contrary to trigonometric functions with $n = 1, 2, 3....$ So power-times-trigonometric functions are suitable for dealing with position changes over time.

Exponential times trigonometric In addition to power-times-trigonometric functions also exponential-times-trigonometric functions can be applied in velocity functions.

Exponential times sine The exponential-times-sine base function is as follows:

$$v_{\text{expsin}}(t) = e^{nt}\sin(2\pi mt) \tag{5.67}$$

where $n \neq 0$ and $m \geq 1$.

The derivative of an exponential-times-sine function is as follows:

$$\dot{v}_{\text{expsin}}(t) = e^{nt}\left[n\sin(2\pi mt) + 2\pi m\cos(2\pi mt)\right] \tag{5.68}$$

and the integral is computed as:

$$\tilde{v}_{\text{expsin}}(t) = \int v dt = \frac{e^{nt}}{n^2 + (2\pi m)^2} \left[n \sin(2\pi m t) - 2\pi m \cos(2\pi m t) \right] \tag{5.69}$$

Exponential times cosine The exponential-times-cosine base function is expressed as:

$$v_{\text{expcos}}(t) = e^{nt} \cos(2\pi m t) \tag{5.70}$$

where $n \neq 0$ and $m \geq 1$.

The derivative of a exponential-times-cosine function is as follows:

$$\dot{v}_{\text{expcos}}(t) = e^{nt} \left[n \cos(2\pi m t) - 2\pi m \sin(2\pi m t) \right] \tag{5.71}$$

and the integral is computed as:

$$\tilde{v}_{\text{expcos}}(t) = \int v dt = \frac{e^{nt}}{n^2 + (2\pi m)^2} \left[n \cos(2\pi m t) + 2\pi m \sin(2\pi m t) \right] \tag{5.72}$$

Exponential-times-trigonometric functions have similar characteristics as power-times-trigonometric functions. However, exponential-times-cosine terms have a non-zero initial value in contrast to power-times-cosine terms.

In the remainder of this report base functions are abbreviated in order to express velocity functions shortly. The way of abbreviating is shown in Table 5.2. Using these abbreviations one expresses e.g. $1 + t^2 + \cos(2\pi t)$ as CPow2Cos. In this table no abbreviations for the exponential and exponential-times-trigonometric base functions are shown, since these are not considered in the remainder of this report. The exponential and exponential-times-trigonometric base functions were found to do not provide (very) good results.

5.4.2. Application

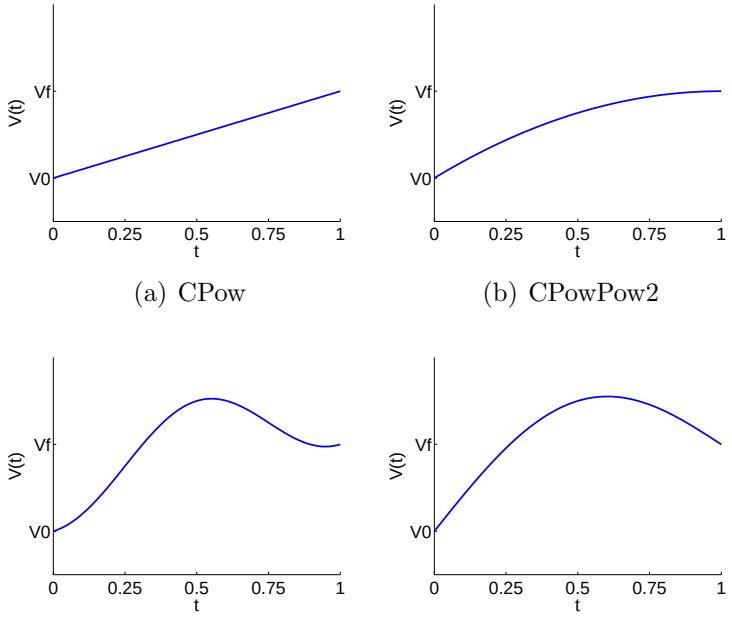
In the following, the application of base functions is treated. This includes a discussion on which base functions can be used to solve the boundary conditions and which base functions are suitable to apply to which velocity function.

The boundary conditions on initial and final velocity can be solved easily using a

Table 5.2: Abbreviations used for base functions; † 05 refers to the factor 0.5, R refers to the factor N and R5 refers to the factor $N + 0.5$, where N is the applied number of revolutions; ‡ X refers to the power order X.

Base function	Abbreviation	Base function	Abbreviation
1 (constant)	C	$\cos(N * 2\pi t)$	CosR †
t (1^{st}-order power)	Pow	$t \sin(2\pi t)$	PSin
t^2	Pow2	$t^2 \cos(2\pi t)$	P2Cos
t^5	Pow5	$t^X \cos(2\pi t)$	PXCos ‡
$\sin(2\pi t)$	Sin	$t^3 \sin(1.5 * 2\pi t)$	P3Sin15
$\sin(0.5 * 2\pi t)$	Sin05 †	$t^6 \cos((N + 0.5) * 2\pi t)$	P6CosR5 †
$\cos(2\pi t)$	Cos	$t^X \cos(0.5 * 2\pi t)$	PXCos05 †‡

constant and a power term, see Figure 5.3a. For the initial velocity condition one simply requires a term which is nonzero at $t = 0$. So, the use of a constant for satisfying the initial velocity condition is most straightforward, but also a cosine is suitable. For the final velocity condition one requires another term which has different values at $t = 0$ and $t = \text{TOF}$ to allow the initial and final velocity to have different values. Therefore, a power term is suitable to satisfy the final velocity condition. However, also exponential or power-times-cosine functions can be used. Finally, one requires a third base function which can adjust the average velocity, such that the condition on the change in position can be satisfied. Such a base function can for example be a 2^{nd}-order power, a sine with $n = 0.5$ or a constant term. Depending on the chosen base functions the velocity changes differently over time, see Figures 5.3b to 5.3d.



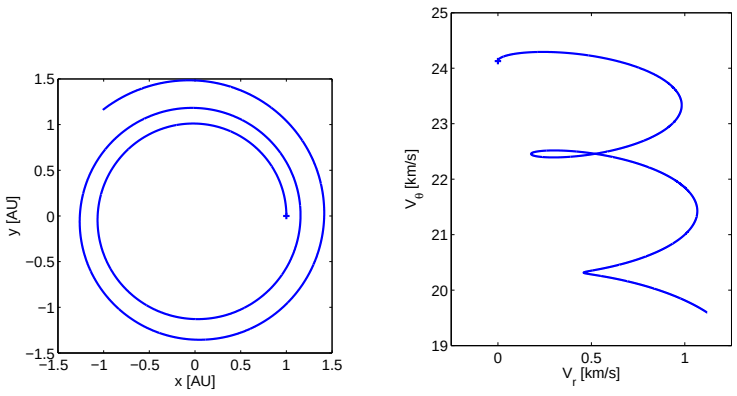
(a) CPow (b) CPowPow2

(c) CPowCos (d) CPowSin05

Figure 5.3: Solving of boundary conditions using different functions.

Shaping the radial and normal velocity For most interplanetary transfers the change in inclination is small, whereas the change in especially semi-major axis and eccentricity is large. For example, a 2D Hohmann transfer to Mars requires a ΔV of 5.50 km/s whereas the inclination change for transfer to Mars requires a ΔV of only 0.68 km/s. The required ΔV is therefore dominated by the change in semi-major axis and eccentricity. Furthermore, since the inclination change is small, the semi-major and eccentricity change are provided (fully) by radial and normal velocity changes, and hardly by axial velocity changes. Therefore, the choice of the radial and normal velocity profiles will have the largest effect on the resulting trajectory regarding ΔV.

The semi-major axis is changed most efficiently by thrusting tangentially to the spacecraft's velocity, as is shown in Eq. (2.28). A flight with constant tangential thrust has been simulated and the resulting trajectory and hodograph are shown in Figure 5.4. It is interesting to see that the velocity hodograph has a curly character. First the normal velocity increases. As a result the radial velocity increases and kinetic energy is converted to potential energy while the normal velocity decreases. This pattern is repeated as the normal velocity keeps decreasing and the radial as well as the normal velocity show period behavior. From this tangential thrust example it is clear that periodic terms in the radial and normal-velocity functions may provide good results. However, also monotonic base functions are required in the radial and normal-velocity functions to allow different initial and final velocities.



(a) Trajectory (b) Velocity hodograph

Figure 5.4: Trajectory and velocity hodograph for a constant tangential thrust acceleration.

Shaping the axial velocity The change of the axial velocity over time is characterized by the required change in inclination. In cylindrical coordinates an inclined orbit has an oscillating axial velocity V_z. Therefore, it seems logical to let V_z oscillate over time with possibly a changing amplitude when an inclination change is required.

The change of the inclination over time can be found in Eq. (2.29). From this equation it is clear that the inclination is changed most efficiently at large distances

from the Sun. When flying to outer planets, this means that the amplitude of the oscillating axial velocity should be changed at the end of the flight when radial distance is largest and the velocity is smallest. This can be done by using e.g. higher-order power terms for shaping the axial velocity. Shaping the motion in axial direction using higher-order power terms has already been applied by Wall [Wall, 2008]. As said, the axial velocity has a periodic behavior when the orbit is inclined. Therefore for the axial velocity function it seems logical to apply periodic base functions with varying amplitude over time, e.g. power-times-trigonometric functions.

Different frequencies for different number of revolutions In the axial velocity function trigonometric base functions may be suitable to enable periodic motion to efficiently change the inclination. In case of an inclined orbit the periodicity of the axial velocity equals the orbital period, the time to complete one revolution. However, when shaping as function of time, the independent variable is not related to the geometry of the problem. Therefore, it may be efficient to apply a periodicity in the axial velocity which is related to the number of revolutions. For shaping as a function of the polar angle, there is no need to do this, since the trigonometric terms have a periodicity which is equal to one revolution by default.

5.5. Implementation

The hodographic-shaping methods have been implemented in the framework of the TU Delft Astrodynamics Toolbox (Tudat). Tudat is C++-library developed by students and staff of the Astrodynamics & Space Missions section of the Faculty of Aerospace Engineering, Delft University of Technology, which contain functions and utilities for astrodynamic simulations. The hodographic-shaping methods have therefore also been programmed in C++ and several functionalities from Tudat, such as coordinates transformations, have been used.

In Figure 5.5 the architecture of the hodographic-shaping method is shown in the form of a flow diagram. The hodographic-shaping method is designed to obtain near-optimal low-thrust *rendezvous* trajectories, therefore the initial and final conditions are determined by the states of the departure and arrival body at departure and arrival, respectively. First, the departure and arrival body and the flight window have to be set. After that a departure date and time of flight are picked and the states of the departure and arrival body at departure and arrival, respectively, are computed. This can be done using ephemerides or by computing the osculating orbital elements and converting these to position and velocity in cylindrical coordinates.

The first three terms of the velocity functions are used to satisfy the boundary conditions. The other terms are used to optimize the trajectory. The values of the free coefficients are picked by the user or in general by an optimizer. Once the values of the free coefficients are set, the boundary conditions are solved (i.e. c_1, c_2

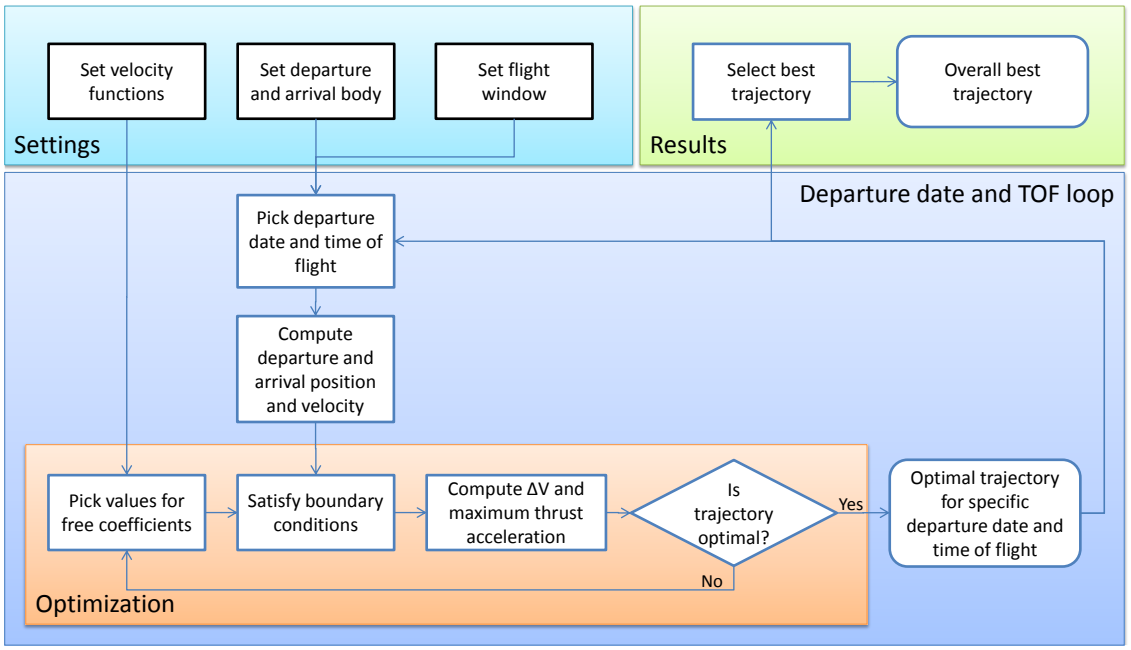


Figure 5.5: Architecture of the hodographic shaping method.

and c_3 are computed) and subsequently the required ΔV and thrust acceleration are computed. After that the user or an optimizer can decide whether the obtained trajectory is optimal. In case the trajectory is found to be non-optimal the process of picking free-coefficient values, satisfying boundary conditions and computing ΔV and thrust acceleration can be repeated. When the optimal trajectory is found, the result is stored. Next, the required initial and final conditions for another departure and time of flight are computed and the trajectory optimization is done again. Once the optimal trajectories for all grid points in the flight window are found, the simulation stops and the best overall trajectory is obtained.

Lowest-order and higher-order solutions As previously discussed, only three base functions per velocity function are required to satisfy the boundary conditions. Therefore, three velocity functions, one for each direction, with each three base functions form the minimum required set of velocity functions to obtain a three-dimensional rendezvous trajectory. Such a set is called a lowest-order solution.

Lowest-order solutions can be improved be adding extra terms to the velocity functions. These extra terms result in degrees of freedom (DoF) and the coefficients corresponding to these additional terms need to be optimized to obtain (near-)optimal trajectories. A set of velocity functions which contains extra terms with free coefficients is called a higher-order solution. The (extra) DoF makes the shaping of velocity functions (more) flexible and therefore results in better trajectories. This flexibility, however, comes at the cost of computational effort, since optimization of the free coefficients is required. Therefore a trade-off between extra computational time and improved results can be made.

5.6. Conclusions

Hodographic shaping is based on shaping velocity hodographs instead of trajectories. As a result the boundary conditions on velocity can be solved very easily and exactly. In addition, it was found that also the boundary conditions on position can be solved fast and exactly. In total nine boundary conditions need to be satisfied for a three-dimensional rendezvous trajectory. Furthermore, for the computation of the required ΔV an integral has to be computed numerically. Also the computation of the polar angle in the time-driven method requires the numerical computation of an integral.

Hodographic shaping is much more simple than spherical or pseudo-equinoctial shaping, since the shaping functions and equations of motion, respectively, are much simpler. In addition, the boundary conditions can be solved more easily and, moreover, exact and fast without the need of iterative computations or constraint optimization. Finally, no constraint on the thrust direction is required to obtain solutions.

The theory of the hodographic-shaping methods has now been discussed. However, the way in which the numerical integration required in the methods is done is not discussed yet. Numerical integration is therefore the topic of the next chapter.

6. Numerical Integration

As discussed in the previous chapter, numerical integration is required in the hodographic-shaping method to compute the ΔV and the final polar angle. The corresponding integration problems are discussed in this chapter. In addition, a few numerical integrators are treated and a trade-off between these is done.

6.1. Problem Statement

In the time-driven method the integrals in Eqs. (5.17) and (5.34) have to be computed numerically to obtain the required ΔV and satisfy the final polar angle condition, respectively. In the polar-angle-driven method only the computation of the required ΔV, Eq. (5.52), requires numerical integration.

Satisfying the final polar angle condition and as well as computing the ΔV have to be done to be able to obtain the ΔV required to fly a trajectory. As a consequence when optimizing for minimum ΔV, the final polar angle and ΔV have to be computed many times. For the comparison of two trajectories during optimization one only needs to know which of two trajectories requires the least ΔV. Therefore, the computed values need not to be extremely accurate. Furthermore, since many evaluations need to done during the optimization, it is desired that the ΔV is computed very fast. Therefore, a numerical integration method is needed that is moderately accurate, but fast.

It is assumed that the ΔV needs to be known with an accuracy of 0.1% to be able to compare the ΔVs of different trajectories well. So, when the ΔV is 5 km/s then a maximum error of 5 m/s is allowed. Next to that, the required accuracy of computed final polar angles is set as follows. A final position accuracy of 100,000 km at a distance of 40 AU from the Sun (Neptune) is assumed. Consequently, a six-decimal-accurate final polar angle is required. Since low-thrust trajectories are considered, the evolution of the polar angle is usually very smooth and monotonous. Therefore, the polar angle is usually obtained more accurately than the ΔV. As a consequence, a numerical integration method which is suitable for accurately computing the ΔV will probably also be suitable to compute the polar angle.

As discussed, an integration method is required which is very fast and moderately accurate. A few of such numerical integration methods are discussed in the following.

6.2. Numerical Integration Methods

In general, numerical integration methods can be expressed mathematically as follows [Montenbruck and Gill, 2000]:

$$\mathbf{y}(t_0 + h) \approx \mathbf{y}_0 + h \cdot \mathbf{\Phi} = \eta(t_0, \mathbf{y}_0) \tag{6.1}$$

where $\mathbf{y}$ is the state vector, $\mathbf{y}_0$ is the initial state at time t_0, h is size of the time-step, $\mathbf{\Phi}$ is the increment function and η is the approximate solution.

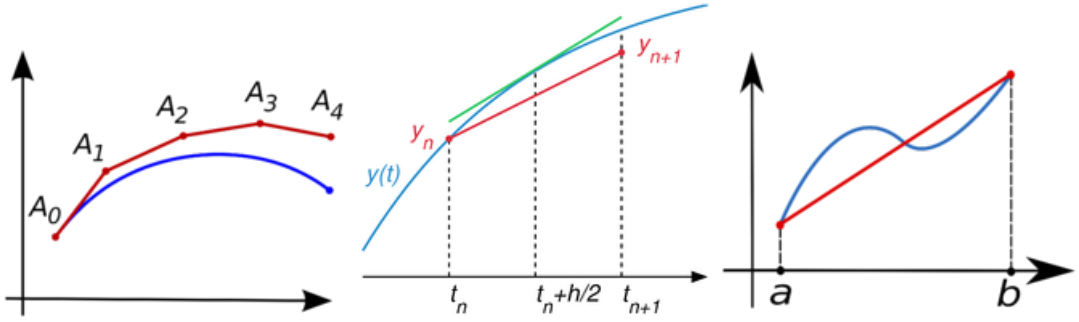
The increment function $\mathbf{\Phi}$ often involves the state derivative $\dot{\mathbf{y}}_0$, which is most generally assumed to be a function of the state and time, i.e. $\dot{\mathbf{y}}_0 = \mathbf{f}(t_0, \mathbf{y}_0)$. The derivatives of the ΔV and the polar angle are the thrust acceleration and the angular velocity, respectively. These derivatives can both be computed analytically as a function of time (or polar angle) only, which is beneficial for the integration accuracy and efficiency.

6.2.1. Euler method

The Euler method uses a very simple integrator which is based on a first-order Taylor expansion and can be written as:

$$\mathbf{y}(t_0 + h) \approx \mathbf{y}_0 + h\mathbf{f}(t_0, \mathbf{y}_0) \tag{6.2}$$

This equation can be interpreted as starting at $(t_0, \mathbf{y}_0)$ and making a time-step h along the tangent of $\mathbf{y}$ at $\mathbf{y}_0$, see Figure 6.1(a).



(a) Euler method (b) Midpoint method (c) Trapezoidal rule

Figure 6.1: Three low-order integration methods [Wikipedia, 2004].

6.2.2. Midpoint method

Instead of proceeding along the slope of $\mathbf{y}$ at $\mathbf{y}_0$, one can also move in the direction of the tangent at $t_0 + h/2$, see Figure 6.1(b). This integration method is called the

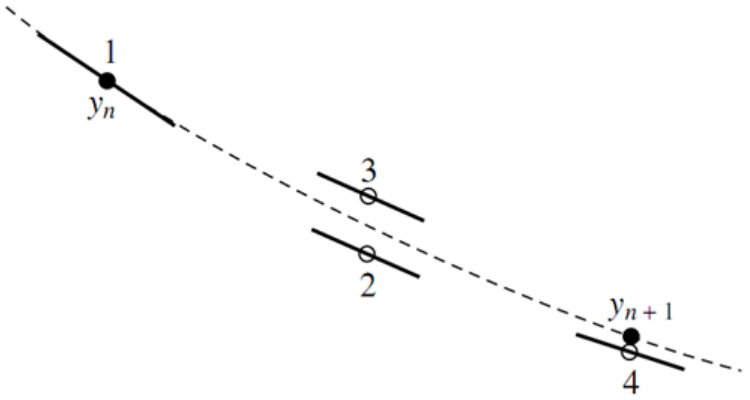


Figure 6.2: Fourth-order Runge-Kutta method. The slope is evaluated four times: once at the initial point, twice at trail midpoints and once at a trail endpoint. From the slopes the final function value is calculated [Noomen, 2010a].

midpoint method and can be expressed as follows:

$$\mathbf{y}(t_0 + h) \approx \mathbf{y}_0 + h\mathbf{f}\left(t_0 + h/2, \mathbf{y}_0 + h\mathbf{f}(t_0, \mathbf{y}_0)/2\right) \tag{6.3}$$

This method is also known as the second-order Runge-Kutta method (RK2).

6.2.3. Trapezoidal rule

In the trapezoidal rule the (approximate) slopes at the initial point and next point are taken and averaged to form the incremental function. The trapezoidal rule can therefore be expressed as follows:

$$\mathbf{y}(t_0 + h) \approx \mathbf{y}_0 + h\frac{\mathbf{f}(t_0, \mathbf{y}_0) + \mathbf{f}\left(t_0 + h, \mathbf{y}_0 + h\mathbf{f}(t_0, \mathbf{y}_0)\right)}{2} \tag{6.4}$$

The name trapezoidal rule stems from the fact that actually a trapezoid is used to approximate the integral of a function, see Figure 6.1(c).

6.2.4. Fourth-order Runge-Kutta method

The fourth-order Runge-Kutta method (RK4) uses the slopes at four different points on the function in order to increase the accuracy of the integration step, see Figure 6.2. The four slopes are weighted to form the increment function $\mathbf{\Phi}$ as follows [Montenbruck and Gill, 2000]:

$$\mathbf{\Phi}_{\mathrm{RK4}} = \frac{1}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4) \tag{6.5}$$

where the four slopes are given by:

$$\begin{aligned}
\mathbf{k}_1 &= \mathbf{f}(t_0, \mathbf{y}_0) \\
\mathbf{k}_2 &= \mathbf{f}(t_0 + h/2, \mathbf{y}_0 + h\mathbf{k}_1/2) \\
\mathbf{k}_3 &= \mathbf{f}(t_0 + h/2, \mathbf{y}_0 + h\mathbf{k}_2/2) \\
\mathbf{k}_4 &= \mathbf{f}(t_0 + h, \mathbf{y}_0 + h\mathbf{k}_3)
\end{aligned} \tag{6.6}$$

RK4 is called a fourth-order method since it can approximate the exact solution up to terms of order h^4. Actually, RK4 is the weighted sum of the midpoint method and the trapezoidal rule such that their errors cancel and only order-h^5 errors per integration step remain.

Since the derivatives of the ΔV and polar angle are known as a function of time (or polar angle), one can write $\mathbf{f}(t_0, \mathbf{y}_0) = \mathbf{f}(t_0)$ and as a consequence the expressions for the integration methods, Eqs. (6.3) to (6.6), simplify significantly. In addition, the values of the derivatives at the end point of an integration step are the same as the derivative values at the start point of the next integration step, i.e. $\mathbf{f}(t_i + h) = \mathbf{f}(t_{i+1})$. Furthermore, since the derivative is known as a function of time, the derivative values k_2 and k_3 for RK4 integration are the same and therefore the RK4 method simplifies to Simpson's rule [Süli and Mayers, 2003]:

$$\mathbf{y}(t_0 + h) \approx \mathbf{y}_0 + h\frac{\mathbf{f}(t_0) + 4\mathbf{f}(t_0 + h/2) + \mathbf{f}(t_0 + h)}{6} \tag{6.7}$$

As a result, one only has to calculate two derivative values per integration step instead of four for the RK4 method. Likewise, for the Euler method, midpoint method and trapezoidal rule, only one derivative value has to be computed per integration step.

So, to speed up the integration process, the derivative values are first computed at the required points in time (or polar angle), i.e. at the start and/or the half-way point of each integration step. After that the integration is performed using these derivative values. This is faster than computing the derivatives for each integration step separately.

Next to the methods discussed above also higher-order or variable step-size integration methods can be used. In case a variable step-size integrator is used instead of a fixed step-size integrator, the benefit of a priori computing derivative values is lost. In addition, variable step-size integrators are more complex to implement, and therefore it is chosen not to use these type of integrators. On the other hand, higher than 4^{th}-order integration methods are also more complex and, again, extremely high accuracies are not required.

6.3. Comparison and Trade-off

In order to compare the four integration methods mentioned above, they are tested on two different problems. The first problem is propagating the function e^{-t^2}. The second problem is calculating the required ΔV for the example of the hodographic-shaping method given in Section 5.1. The computation of e^{-t^2} is used to compare the accuracy of the methods, and the hodographic-shaping method example is used to compare the speed of the methods. The thrust acceleration and the derivative of e^{-t^2}, which need to be integrated to obtain the required ΔV and e^{-t^2}, respectively, can both be determined analytically as a function of time.

Hodographic-shaping method example is used as test case, since in the end the integrator will be applied to this method. However, since the verification of the hodographic-shaping methods is done after an integration method is selected, the propagation of e^{-t^2} is used the test the accuracy of the integrators.

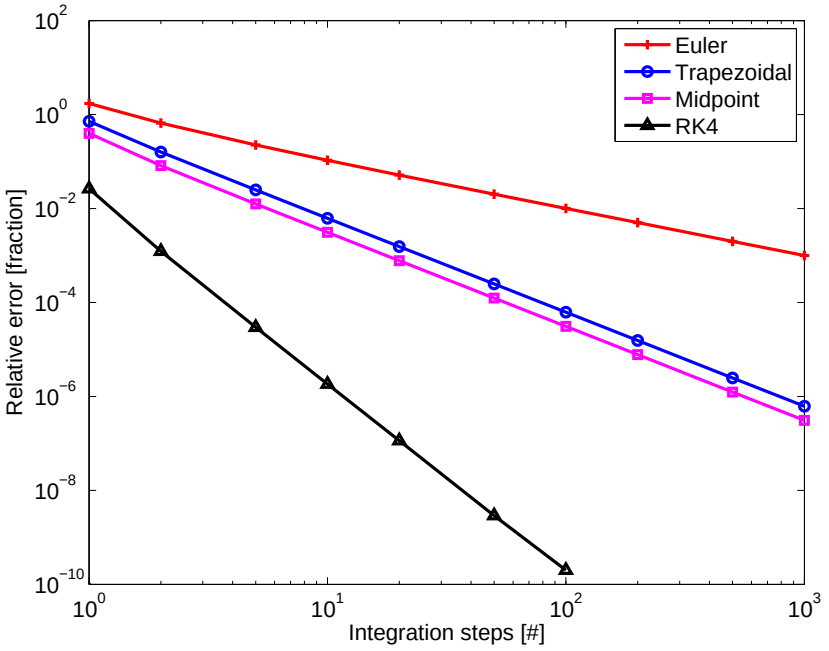


Figure 6.3: Error relative to exact solution versus number of integration steps for four different integration method applied to the propagation of e^{-t^2}.

The results of the integration-error test are shown in Figure 6.3. This figure shows the integration error relative to the exact solution of e^{-t^2} for integrating the derivative from $t = 0$ to $t = 1$. From this figure it is clear that the RK4 method is much more accurate than the other methods, which was to be expected. More specifically, RK4 is twice as accurate as the other methods regarding the order of magnitude of the relative error.

From the test of calculating the required ΔV it was found that the Euler method, midpoint method and trapezoidal rule are equally fast and that the RK4 method is two times slower for a given number of integration steps, see Figure 6.4. This result agrees with the fact that the RK4 method requires two derivative evaluations per

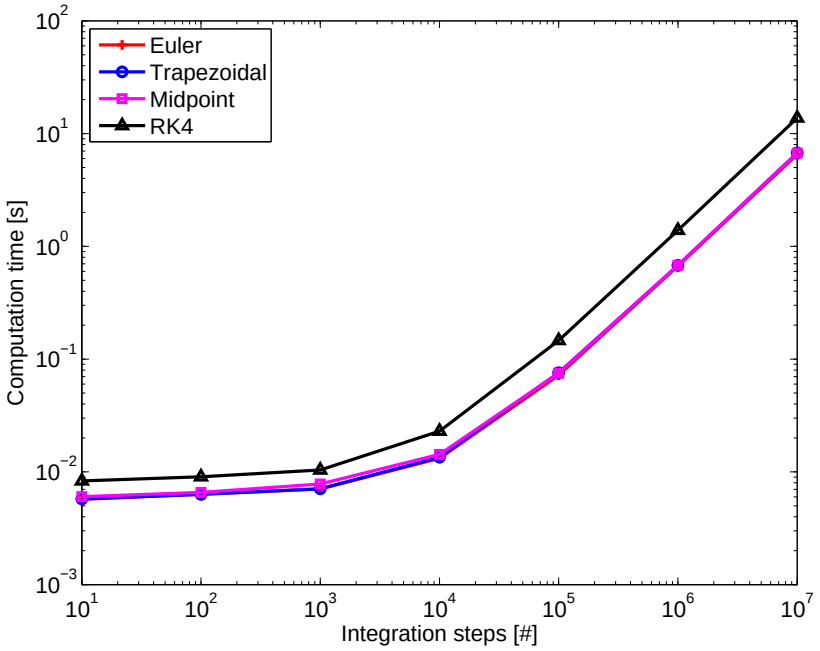


Figure 6.4: Computation time versus number of integration steps for four different integration method applied to the integration of a thrust acceleration.

integration step and the Euler method, midpoint method and the trapezoidal rule only one. In Figure 6.4 the line corresponding to Euler integration is hard to see, since it is overlapped by the trapezoidal line. Furthermore, the lines are not linear, but are flattened at low number of integration steps, since a minimum amount of computation time is required independent of the number of steps.

For both problems, the RK4 method required only 20 integration steps to determine the solution up to six digits accurately. This is a very low number of integration steps, even when taking into consideration the fact that the thrust acceleration and the derivative of e^{-t^2} are smooth functions. Moreover, six-digit-accurate integration results fulfill the requirement on ΔV and polar angle accuracy.

Concluding, RK4 is at least twice as accurate in orders of magnitude(!) and only twice as slow than the other integration methods. Therefore RK4, implemented as Simpson's rule (Eq. (6.7)), is chosen as numerical integrator for calculating the ΔV and polar angle in the hodographic-shaping method.

7. Verification

In order to be sure that trajectories computed by the hodographic-shaping methods are correct, the methods have to be verified. Therefore one has to check whether the boundary conditions are satisfied and whether the computed ΔV and final polar angle are correct.

In the following first the verification method is explained. After that the verification results are discussed both quantitatively and qualitatively.

7.1. Method

The way in which the hodographic-shaping methods are verified is as follows. First, for a trajectory computed by the hodographic-shaping method the required thrust acceleration components are taken for many points in time during the transfer. This thrust acceleration data is then used to propagate the spacecraft by numerically integrating the equations of motion. For the integration the initial position and velocity of the spacecraft for the trajectory are used as initial state. Finally, the trajectory which results from the numerical propagation is compared with the trajectory resulting from the hodographic-shaping method.

The propagation of the spacecraft is done using Cowell's method. For this method the equations of motion which are numerically integrated are written as [Cornelisse et al., 1979]:

$$\frac{d^2\mathbf{r}}{dt^2} = \mathbf{f}_{tot} \tag{7.1}$$

where the total acceleration $\mathbf{f}_{tot}$ is given by [Cornelisse et al., 1979]:

$$\mathbf{f}_{tot} = -\frac{\mu}{r^3}\mathbf{r} + \mathbf{f} \tag{7.2}$$

with $\mathbf{f}$ being the thrust acceleration in our case.

For numerically integrating the equations of motion cylindrical coordinates are used. Cylindrical states namely vary much less than Cartesian states during a transfer and therefore the integration results are much more accurate for the same number of integration steps. Furthermore, the thrust-acceleration components are also expressed in cylindrical coordinates, which simplifies the usage of these. However, the reference frame corresponding to cylindrical coordinates is a rotating one, see Section 2.2.2. Therefore, in addition to the gravitational and thrust accelerations, also centrifugal and Coriolis accelerations have to be taken into account. As a consequence the time derivatives of the cylindrical velocity components, and thus the components of the total acceleration $\mathbf{f}_{tot}$, become:

$$\dot{V}_r = -\frac{\mu}{s^3} r + \frac{V_\theta^2}{r} + f_r \tag{7.3}$$

$$\dot{V}_\theta = -\frac{V_\theta V_r}{r} + f_\theta \tag{7.4}$$

$$\dot{V}_z = -\frac{\mu}{s^3} z + f_z \tag{7.5}$$

The trajectory is then computed by propagating the spacecraft's state by numerically integrating V_r, V_θ, V_z, $\dot{V}_r$, $\dot{V}_\theta$ and $\dot{V}_z$ from $t = 0$ to $t = \text{TOF}$ using an RK4 integrator. In addition, the polar angle θ is computed by additionally integrating $V_\theta / r (= \dot{\theta})$. The verification is completed by checking whether the shaped trajectory and the propagated trajectory are the same by comparing their final position and velocity components and required ΔV's.

The verification of the polar-angle-driven method (option 2) is done in a similar way, but a little bit differently. For the verification of the time-driven method one uses thrust-acceleration components at equally-spaced time intervals. However, for the polar-angle-driven method it is not possible to output the thrust accelerations at equally-spaced moments in time, since time is not an independent variable. Therefore, the thrust acceleration components are put out at equally-spaced polar-angle intervals. Consequently, the spacecraft is propagated from $\theta = 0$ to $\theta = \theta_f$ and the integration step-size for the propagation is a polar-angle step, h_θ. The integration of the state derivatives, which are derivatives with respect to time, still needs to be done by integrating over time. For that reason, the polar-angle step h_θ is divided by the angular velocity to obtain the corresponding time step h: $h = \frac{h_\theta}{d\theta/dt}$.

7.2. Results

Verification has been done for rendezvous missions from Earth to Mars, comet Tempel 1 and Mercury. Earth and Mars were assumed to have circular orbits and Mercury and Tempel 1 elliptical orbits. Moreover, the rendezvous with Mercury and Tempel 1 have been assumed to take place at their pericenter and apocenter, respectively. The verification results of the time-driven method shown in this section were generated using a two-dimensional version of the hodographic-shaping method. Nevertheless these results are similar to verification results obtained for the three-dimensional hodographic-shaping method.

7.2.1. Verification of trajectory

To check whether the thrust profiles computed by the hodographic-shaping method are correct, two random trajectories to the apocenter of Tempel 1 and the pericenter of Mercury have been verified. These two trajectories were randomly generated

during Monte Carlo searches. The results of the hodographic shaping and the numerical propagation using the thrust profile are shown in Tables 7.1 and 7.2 for a trajectory to Tempel 1 and Mercury, respectively. The corresponding trajectories and velocity hodographs are shown in Figures 7.1 and 7.2, respectively.

Table 7.1: Results generated by hodographic shaping as function of time (with 10^6 integration steps) and by numerical propagation (with 1.4×10^6 integration steps) for a test mission from Earth to the apocenter of Tempel 1 with a TOF of 4 years, V_r=V_θ=CPowSinSin2CosCos2Pow2Pow3PSinPCos.

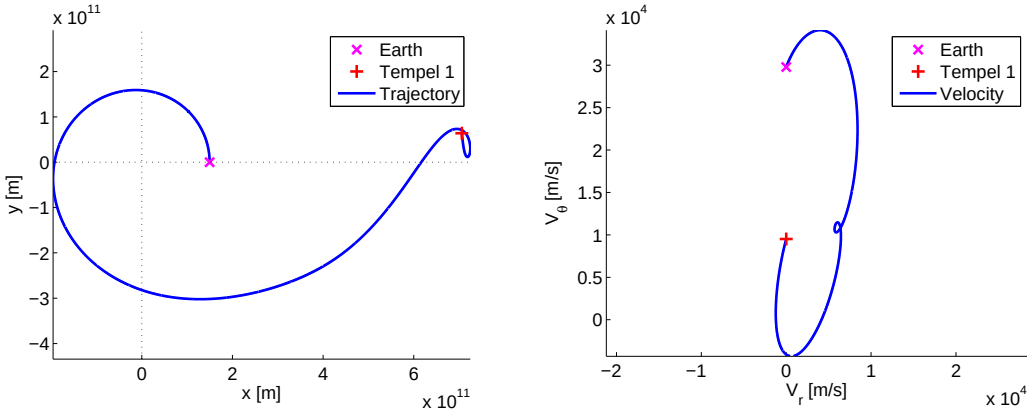
Result	Hodographic shaping	Numerical propagation	Difference	Relative difference
$r(t_f)$ [km]	7.089240×10^8	7.089240×10^8	9.96×10^{-4}	1.40×10^{-12}
$\theta(t_f)$ [rad]	6.3736486527	6.3736486527	3.29×10^{-11}	5.17×10^{-12}
$V_r(t_f)$ [km/s]	2.109×10^{-15}	2.595×10^{-11}	2.595×10^{-11}	1.00
$V_\theta(t_f)$ [km/s]	9.5099	9.5099	3.76×10^{-12}	3.96×10^{-13}
ΔV [km/s]	84.2013	84.2013	1.19×10^{-9}	1.41×10^{-11}

Table 7.2: Results generated by hodographic shaping as function of time (with 2×10^6 integration steps) and by numerical propagation (with 7×10^5 integration steps) for a test mission from Earth to the pericenter of Mercury with a TOF of 2 years, V_r=V_θ=CPowCosSinSin2Sin3Cos2Cos3Pow2Pow3PSinPCos.

Result	Hodographic shaping	Numerical propagation	Difference	Relative difference
$r(t_f)$ [km]	4.718040×10^7	4.718040×10^7	1.70×10^{-4}	3.59×10^{-12}
$\theta(t_f)$ [rad]	6.15519459	6.15519459	1.20×10^{-9}	1.94×10^{-10}
$V_r(t_f)$ [km/s]	-5.9175×10^{-16}	-8.9196×10^{-11}	8.92×10^{-11}	1.00
$V_\theta(t_f)$ [km/s]	58.2340	58.2340	1.94×10^{-10}	3.34×10^{-12}
ΔV [km/s]	215.9514	215.9514	4.91×10^{-9}	2.28×10^{-11}

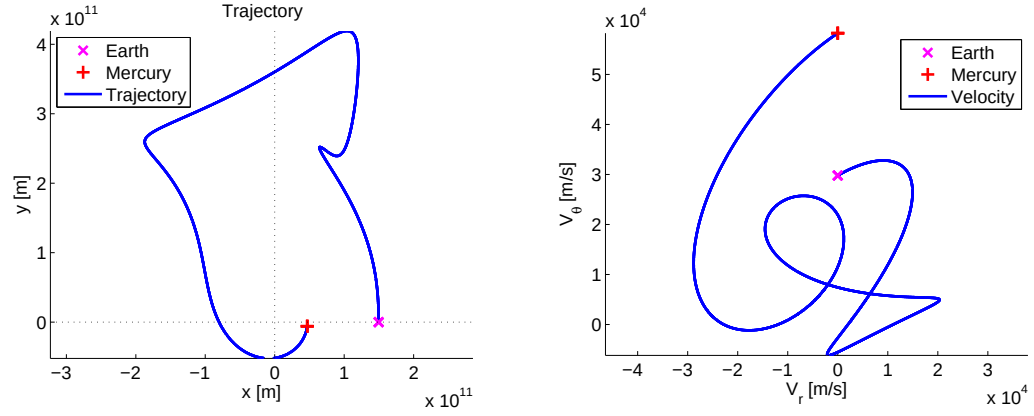
In Tables 7.1 and 7.2 the final position and velocity and the required ΔV computed by the shaping method and the numerical propagation are shown. For the numerical propagation the same initial position and velocity as for the shaping method have been used, and therefore these are the same. For the computation of the ΔV and final polar angle in the shaping method 1.0×10^6 and 2.0×10^6 integration steps were used for the trajectory to Tempel 1 and Mercury, respectively. These numbers of integration steps are very large, but make sure that the results of the hodographic-shaping method are not erroneous due to integration errors.

From the two tables it is clear that the shaping method and the numerical propagation give the same final results. Therefore one can conclude that trajectories and corresponding thrust profiles computed by the hodographic-shaping method correspond and match with each other. The differences between the results are about 11 orders of magnitudes smaller than the results itself and can therefore be neglected. These marginal differences are probably due to integration errors of the hodographic-shaping method and the numerical integration.



(a) Trajectory (b) Velocity hodograph

Figure 7.1: Trajectory and velocity hodograph corresponding to the test trajectory to comet Tempel 1 in Table 7.1.



(a) Trajectory (b) Velocity hodograph

Figure 7.2: Trajectory and velocity hodograph corresponding to the test trajectory to Mercury in Table 7.2.

In Figures 7.1 and 7.2 the trajectories and velocity hodographs corresponding to Tables 7.1 and 7.2 are shown. The trajectories seems to be far from optimal, especially the one to Mercury seems to make no sense at all. However, it was found that the trajectories fulfill the initial and final boundary condition and therefore the trajectories are correct (in spite of the large ΔV, see Table 7.2). This shows that independent of the quality of result the hodographic-shaping method gives correct results.

In Tables 7.3 and 7.4 the verification results of the polar-angle-driven method for two trajectories to Mercury are shown. Both trajectories have the same departure date and TOF, but different velocity functions and number of revolutions. The final states generated by the hodographic-shaping method for both trajectories are the same, except of course for the final polar angle and ΔV. In contrast to the time-driven results the final states generated by the numerical propagation differ from the states generated by the shaping method. These differences are very probably caused by integration errors of the numerical propagation. As discussed

Table 7.3: Results generated by hodographic shaping as function of polar angle (with 10^6 integration steps) and by numerical propagation (with 2×10^6 integration steps) for a test mission from Earth to the pericenter of Mercury with a TOF of 800 days and $N = 0$, R=CPowPow2Pow3Pow4Pow5Pow6, T=CPowPow2Pow3Pow4Pow5Pow6, Z=CosPCosPSinP2CosP2Sin.

Result	Hodographic shaping	Numerical propagation	Difference	Relative difference
$r(t_f)$ [km]	6.5722524×10^{7}	6.5722546×10^{7}	2.23×10^{1}	3.39×10^{-7}
$\theta(t_f)$ [rad]	4.3413511	4.3413489	2.17×10^{-6}	5.00×10^{-7}
$z(t_f)$ [km]	-7.4467420×10^{6}	-7.4467377×10^{6}	4.25	5.71×10^{-7}
$V_r(t_f)$ [km/s]	-6.40481	-6.40492	1.04×10^{-4}	1.62×10^{-5}
$V_\theta(t_f)$ [km/s]	40.97121	40.97128	7.33×10^{-5}	1.79×10^{-6}
$V_z(t_f)$ [km/s]	-1.22155	-1.22155	2.25×10^{-6}	1.84×10^{-6}
ΔV [km/s]	73.24160	73.24145	1.49×10^{-4}	2.03×10^{-6}
t_f [days]	800.0	800.0	1.02×10^{-12}	1.28×10^{-15}

in Section 7.1 the numerical propagation used to verify the polar-angle-driven results requires that the equations of motion are integrated over the polar angle, and therefore the state derivatives have to be multiplied by the angular velocity. However, since the angular velocity is computed by numerical integration itself, the integration error is larger than for the time-driven method. As a result a difference occurs between the results of hodographic shaping and numerical propagating. Nevertheless, for both trajectories the differences are less than 1000 km for the position and less than 1 m/s for the velocity.

Now that it is clear that the trajectories and thrust profile computed by the hodographic-shaping methods are correct, the accuracy of computed ΔV and final polar angle are considered.

7.2.2. Accuracy of computed ΔV and final polar angle

In the hodographic-shaping method the required ΔV and final polar angle θ_f are computed using numerical integration. As a consequence the accuracy of the estimated ΔV and θ_f values depends on the number of integration steps used for the integration. In order to find out how many integration steps are required to obtain sufficiently accurate results, computations have been done using different numbers of integration steps. The results are shown in Tables 7.5 to 7.8. The absolute errors shown in these tables are errors with respect the corresponding results computed using 10^6 integration steps. In Table 7.8 θ_f is not shown, since in the polar-angle-driven method θ_f is not computed using numerical integration.

From these tables it is clear that, as expected, the accuracy of the results increases with increasing number of integration steps. Furthermore, one can see that trajectories which require a low ΔV have smaller integration errors than those which

Table 7.4: Results generated by hodographic shaping as function of polar angle (with 10^6 integration steps) and by numerical propagation (with 2×10^6 integration steps) for a test mission from Earth to the pericenter of Mercury with a TOF of 800 days and $N = 3$, R=CPowPow2Pow3Pow4Pow5Pow6, T=CPowPow2Pow3Pow4Pow5Pow6, Z=CosPCosPSinP2CosP2Sin.

Result	Hodographic shaping	Numerical propagation	Difference	Relative difference
$r(t_f)$ [km]	6.5722524×10^7	6.5722643×10^7	1.19×10^2	1.81×10^{-6}
$\theta(t_f)$ [rad]	23.1909070	23.1908954	1.16×10^{-5}	5.00×10^{-7}
$z(t_f)$ [km]	-7.4467420×10^6	-7.4467192×10^6	2.27×10^1	3.05×10^{-6}
$V_r(t_f)$ [km/s]	-6.40481	-6.40475	5.91×10^{-5}	9.23×10^{-6}
$V_\theta(t_f)$ [km/s]	40.97121	40.97120	4.54×10^{-6}	1.11×10^{-7}
$V_z(t_f)$ [km/s]	-1.22155	-1.22160	4.82×10^{-5}	3.94×10^{-5}
ΔV [km/s]	20.54232	20.54224	7.61×10^{-5}	3.71×10^{-6}
t_f [days]	800.0	800.0	0.0	0.0

require a large ΔV. Actually, it was found that the integration error decreases when the trajectories are more optimal, see Section 7.2.3. In addition, the complexity of the mission geometry seems to have an influence on the error of the estimated ΔV, since the accuracy of the computed ΔV for a mission to Mercury is larger than that for a mission to Tempel 1 or Mars. For complex mission geometries the thrust acceleration may vary strongly over time, and therefore the integration error may increase. The angular velocity, on the other hand, in general varies more gradually and therefore the accuracy is higher.

Table 7.7 show that even for a very bad trajectory with a TOF of 8 years, only 50 integration steps are required to obtain a position error which is less than 1000 km. On the other hand, Tables 7.5 and 7.6 indicate that for more optimal trajectories only 20 or 15 integration steps are sufficient for a 1000 km-accurate final position estimate. In addition, only 20 integration steps are required to estimate the ΔV within 0.1%. Based on these results it was decided to apply a fixed number of 25 integration steps for numerical integration in the hodographic-shaping methods. Using 25 integration steps, the integration errors are sufficiently low for both the ΔV and final polar angle, except for very bad trajectories which are not of interest (they will be superseded by better ones in the optimization process), and the required computational effort is low as well. A different number of integration steps can be applied easily if one wants to increase the accuracy of the estimated results or to reduce the required computational effort.

In the following two paragraphs two phenomena are discussed which were found during verification and which should be kept in mind when working with the hodographic-shaping method.

Table 7.5: Estimated ΔV and final polar angle and corresponding absolute errors for different number of integration steps for the time-driven method, for a test mission from Earth to Mars with a TOF of 3 years and V_r=CCosPowSin and V_θ=CPowSin.

Number of integration steps	ΔV [km/s]	ΔV error [m/s]	Final polar angle [rad]	Position error [km]
10	5.860628	2.298	14.2972879	2483.5
15	5.858385	0.056	14.2972792	489.0
20	5.858355	0.025	14.2972777	153.9
25	5.858365	0.035	14.2972773	62.7
50	5.858331	0.001	14.2972770	3.9
100	5.858330	0.000	14.2972770	0.3
1000	5.858330	0.000	14.2972770	0.0

Table 7.6: Estimated ΔV and final polar angle and corresponding absolute errors for different number of integration steps for the time-driven method, for a test mission from Earth to Mercury's pericenter with a TOF of 2 years and V_r=CCosPowSin and V_θ=CPowSin.

Number of integration steps	ΔV [km/s]	ΔV error [km/s]	Final polar angle [rad]	Position error [km]
10	152.975	1.906	45.0307430	11471.2
15	155.009	0.129	45.0305492	2331.4
20	154.938	0.058	45.0305156	745.1
25	154.989	0.109	45.0305063	306.6
50	154.838	0.043	45.0305002	19.3
100	154.885	0.004	45.0304999	1.2
1000	154.880	0.000	45.0304998	0.0

Exponential term In Section 5.4 it was discussed that the independent variables time and polar angle are scaled in the base functions by their final value. During verification it was found that scaling of the independent variable is especially important when exponential base functions are used. First of all, e^t-terms can become very large when the independent variable is not scaled. Furthermore, the value of e^{-t}-functions decreases extremely quickly from 1 to 0 when an unscaled independent variable is used. The e^{-t}-term will therefore act like an impulse function at $t = 0$, which results in large integration errors.

Negative radial distance For trajectories to Mercury sometimes a remarkable thing occurred. When flying from Earth to Mercury one has to decrease the radial distance from the Sun. As a consequence the average radial velocity has to be negative in order to reach Mercury. However, since the orbit of Mercury is eccentric it may occur that the final radial velocity required to rendezvous with Mercury is

Table 7.7: Estimated ΔV and final polar angle and corresponding absolute errors for different number of integration steps for the time-driven method, for a test mission from Earth to Tempel 1's apocenter with a TOF of 8 years and V_r=V_θ=CPowCosSinSin2Cos2Pow2Pow3PSinPCos.

Number of integration steps	ΔV [km/s]	ΔV error [m/s]	Final polar angle [rad]	Position error [km]
10	442.343	642.806	13.4851913	7564819.2
15	442.517	468.527	13.4956401	157418.7
20	442.912	73.236	13.4959232	43285.3
25	443.009	23.757	13.4958870	17632.0
50	442.988	2.338	13.4958632	743.8
100	442.986	0.188	13.4958622	42.7
1000	442.985	0.000	13.4958622	0.0

Table 7.8: Estimated ΔV and corresponding absolute errors for different number of integration steps for the polar-angle-driven method, for a test mission from Earth to Mercury with N=4, departure date is 5045 MJD2000, TOF is 540 days and R=T=CPowPow2Pow3Pow4Pow5 and Z=CosPCosPSinP2CosP2SinP3Cos.

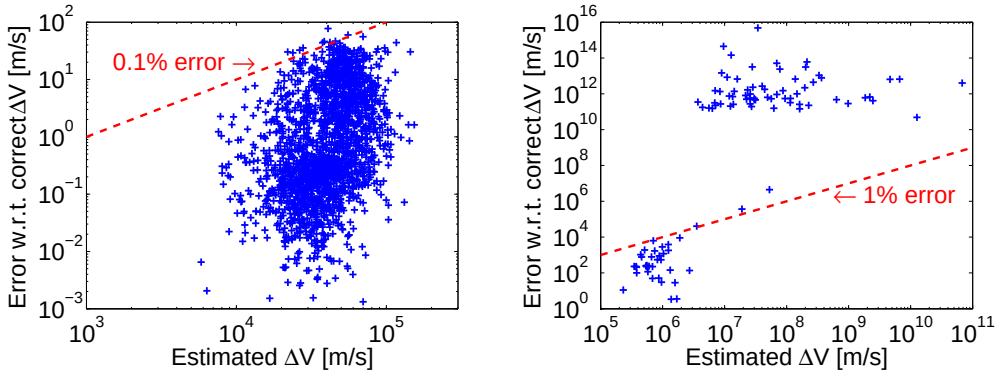
Number of integration steps	ΔV [km/s]	ΔV error [m/s]
15	39.2655	117.95
20	39.1595	11.99
25	39.1383	9.25
50	39.1476	0.08
100	39.1475	0.01
1000	39.1475	0.00

positive. In such a situation, one will approach Mercury from the inside, i.e. from the Sun side. If the final positive radial velocity is large one needs to fly far inside the orbit of Mercury to compensate for the radial distance which will the covered in the final approach. Consequently, it may occur that the hodographic-shaping method obtains a trajectory with a minimum radial distance which is negative. This is of course physically impossible. Fortunately, the ΔV required to fly such trajectories is very large, as is shown in the following section. Therefore these trajectories will never be found to be (near-)optimal, however, they may be present as solution candidate during an optimization process.

Next the error of the estimated ΔV with respect to the value of the estimated ΔV is discussed. This topic is of interest since trajectories are usually optimized with respect to ΔV.

7.2.3. Error versus estimated ΔV

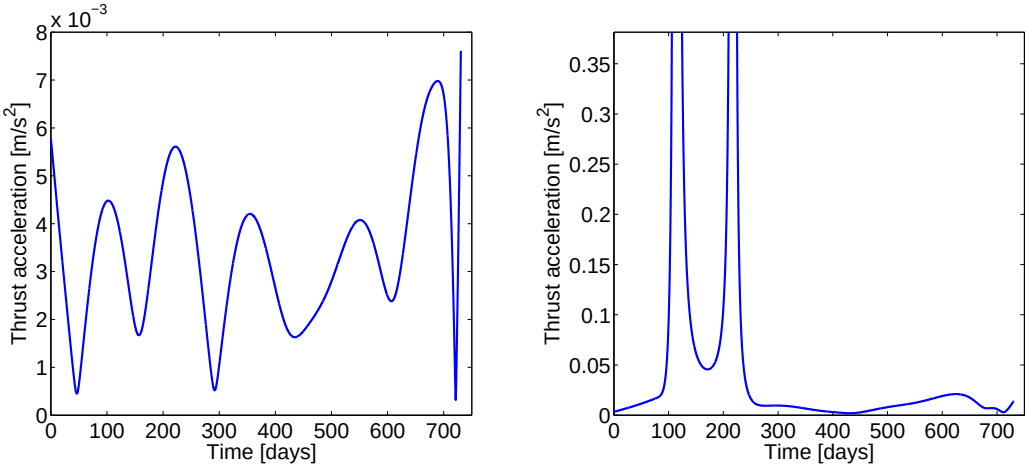
During the optimization of trajectories, trajectories are ranked according to the required ΔV. In order to be sure that the best trajectories are ranked correctly, it is required that low ΔV are estimated accurately and that high ΔV are not underestimated largely. On the other hand, if high ΔV are overestimated then this is not a problem, since only trajectories which require a low ΔV are of interest.



(a) Mars (b) Mercury

Figure 7.3: Error of estimated ΔV for trajectories to a) Mars and b) Mercury.

To be sure that ranking is done correctly, it has been investigated whether the error in the estimated ΔV is significant and whether it depends on the magnitude of the estimated ΔV. The results are shown in Figure 7.3, for trajectories to Mars and Mercury. From these figures it is clear that only for very bad trajectories the error in estimated ΔV is large ($> 1\%$). This means that ranking of trajectories using estimated ΔV is done correctly for high-ranked (good) trajectories, since only for the low-ranked, extremely bad trajectories the estimated ΔV is very inaccurate.



(a) Feasible (b) Unfeasible

Figure 7.4: Thrust acceleration profiles for a) a feasible and b) an unfeasible (requires an infinite thrust acceleration) trajectory to Mercury.

Integration errors usually increase when the derivative function changes less gradually and when the derivative function is non-smooth. In Figure 7.4 the thrust-acceleration profiles of a good (i.e. low ΔV) and a bad (i.e. high ΔV) trajectory to Mercury are shown. The thrust acceleration in Figure 7.4a changes not very gradually, but is nevertheless smooth, whereas the thrust profile in Figure 7.4b contains two asymptotic spikes and is therefore not smooth. The two spikes are singularities caused by the fact that the spacecraft flies 'through' the Sun, first to a negative radial distance and subsequently back to a positive radial distance. Since such a flight is physically impossible, the trajectory is unfeasible. An integrator with a relatively large step-size can easily misjudge the height of these spikes and as a consequence the required ΔV is estimated wrongly. However, the estimated ΔV will still be very large. Since good trajectories do not have thrust profiles with singularities or high spikes, the integration errors for these trajectories are small. This explains why the error in estimated ΔV is large for bad trajectories and small for good trajectories to Mercury.

Next to the small estimated ΔV with small errors and large estimated ΔV with huge errors, Figure 7.3b also shows that there are some situations with a large estimated ΔV with relatively small errors (the two points in the middle). It was found that these large estimated ΔV correspond to trajectories which do not reach negative radial distances, but come very close to the Sun. Therefore, their thrust acceleration profile also contains a spike, which results in a high ΔV. Such spikes, however, don't behave asymptotic and therefore the required ΔV and the integration error are limited.

This discussion on the error of the estimated ΔV concludes this chapter on verification. Now, that it has been verified that the hodographic-shaping methods function well, the optimization of velocity functions is considered, starting with a brief discussion on the theory of optimization, in the next chapter.

8. Optimization

In Chapter 5 two hodographic-shaping methods have been presented. As discussed, higher-order solutions of these methods contain free parameters which can to be optimized to obtain optimal trajectories. In this chapter the corresponding optimization problem is discussed and, in addition, a few optimizers are considered which can possibly solve this problem.

8.1. Optimization Problem

The goal of preliminary design of low-thrust trajectories is to find feasible and near-optimal trajectories. In general, a trajectory is feasible if the thrust-level limit is not exceeded, the required propellant mass fraction is smaller than 1 and the boundary conditions are satisfied. In addition, a trajectory is (near-)optimal when it is better than (most) other ones. Whether or not a trajectory is better depends on the objective, or on multiple objectives.

For low-thrust transfer trajectories the objective is, most commonly, to:

- minimize required ΔV
- maximize final mass
- minimize time of flight or
- a variation or combination of these

Other objectives may of course also be possible. An objective can usually be expressed in mathematical form as an objective function J, which is used for the optimization. In an objective function extra terms can be included called penalty functions. These penalty functions serve as weak constraints by penalizing violation of constraints, e.g. on maximum thrust level.

It should be noticed that in case no penalty mechanism is applied, minimization of the required ΔV gives the same result as maximizing the final mass. For the hodographic-shaping method, the objective of minimum required ΔV will be used to optimize low-thrust trajectories. If necessary, the obtained ΔV can be used to compute the final mass fraction or propellant mass fraction, using Tsiolkovsky's rocket equation Eq. (3.6).

The objective function for minimizing the required ΔV is as follows:

$$J = \Delta V = \int_0^{t_f} f\,dt = \int_0^{t_f} \sqrt{f_r^2 + f_\theta^2 + f_z^2}\,dt \tag{8.1}$$

where f_r, f_θ and f_z are the thrust acceleration components in radial, normal and axial direction, respectively. It can be noticed that this objective function for

minimizing ΔV is thus a function of the thrust acceleration profile. In addition, the objection function is clearly nonlinear, which makes the optimization complex.

Since in the hodographic-shaping method the thrust acceleration is determined by the velocity functions, the velocity functions need to be optimized in order to minimize the required ΔV. In addition to optimizing the velocities functions, it is essential that the velocity functions satisfy the boundary conditions on position and velocity. This can be done by constraining some coefficients. These coefficients which solve the boundary value problem are the 'fixed' coefficients. The free coefficients, on the other hand, are used as optimization parameters. The way in which the boundary conditions are satisfied was already explained in Sections 5.2 and 5.3 for the time-driven and polar-angle-driven method, respectively.

So, the purpose of the optimization is to minimize the required ΔV. This is done by optimizing the free coefficients of the velocity functions, which, together with the boundary conditions, determine the fixed coefficients. All coefficients together finally determine the required ΔV which is the objective-function value.

For a transfer between two celestial bodies, the trajectory largely depends on the launch date and time of flight. Therefore, if the objective is to minimize the required ΔV, then next to optimizing the velocity functions, also the launch date and the time of flight need to be optimized.

Furthermore, the amount of thrust which can be delivered by the rocket is limited, e.g. by the achievable exhaust velocity or the available input power, see Section 3.1. This limit can be taken into consideration when optimizing the velocity functions in two ways. One way is to simply reject trajectory solutions during optimization that do not respect the thrust limit. Another possibility is to implement a penalty function into the objective function, which penalizes violation of the thrust limit. An example of an objective function including such a penalty is as follows:

$$J = M_{prop}(t_f) + P \cdot \delta(T - T_{max}) \tag{8.2}$$

where P is the penalty factor and $\delta(T - T_{max})$ is 0 if $T \leq T_{max}$ and 1 if $T > T_{max}$.

Summarizing, the optimization of trajectories in the hodographic-shaping methods will be done by optimizing the velocity functions, departure date and time of flight for minimum ΔV. In addition, a thrust-level limit may be considered during the optimization. Such thrust limit has not been applied in this research, but may be implemented in the future.

8.2. Optimization Methods

In general, one is interested in the overall best trajectory, i.e. in the global optimum. Finding the global optimum can be very complicated since the search space might be very large and/or multiple local minima might be present. Today, many

optimization methods are available which can find global optima. All these methods have their own characteristics regarding robustness (does the method find the global optimum) and convergence speed (how fast does the method find the optimum). In the study preceding this research several optimization methods have been investigated and compared. Based on that study four methods have been selected to be used to assess and solve the optimization problem corresponding to the hodographic-shaping method. These methods are grid search, the Monte Carlo method, the Nelder-Mead simplex method and Differential Evolution. These four methods are discussed here. In the next chapter the optimization problem will be assessed and the most suitable optimization method will be chosen.

8.2.1. Grid search

In a grid search the problem space is first discretized using a grid and then the objective function is evaluated at each grid point. The grid point with the lowest objective function is assumed to be the global minimum. The main disadvantage of grid search is that the search space is often so large that either the computation time becomes immense or the grid spacing becomes very large (which results in limited resolution) [Noomen, 2010b].

The issue of the limited resolution can be tackled by performing subsequent searches with more dense grids in possible-minimum regions.

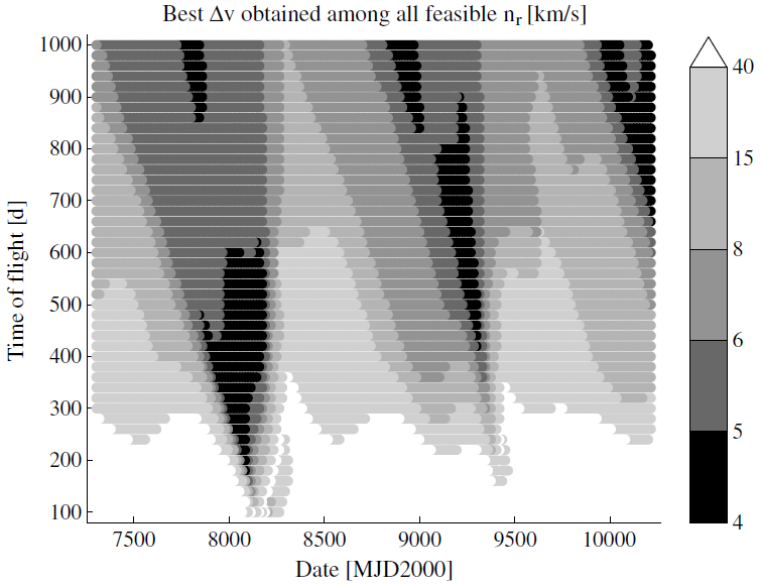


Figure 8.1: Best ΔV's found by spherical shaping method for a rendezvous with asteroid 1989ML using a grid search for launch date and time of flight [Novak and Vasile, 2011].

Grid searches can very well be applied for optimization of the launch date and time of flight by simply stepping through the possible launch dates and transfer times in discrete steps. An example of a grid search optimization for launch date and time of flight is shown in Figure 8.1.

8.2.2. Monte Carlo method

The Monte Carlo method tries to find the global optimum by evaluating the objective function at many points in the search space, just like the grid search. However, in the Monte Carlo method the trail points are chosen randomly and not using a grid [Noomen, 2010b]. Usually a uniform distribution is used to randomly select the points.

On the one hand, since the Monte Carlo method chooses the trail points randomly and not using a regular grid, points can be assessed which would not be assessed using a grid. On the other hand, since the points are picked randomly, it can occur that in some areas no points are assessed at all, whereas in other areas many points are assessed. The major drawback of the Monte Carlo is that, just like the grid search, the computation time becomes immense when the search space is large.

8.2.3. Nelder-Mead simplex method

The Nelder-Mead (NM) simplex method is a single-objective optimization method for an n-dimensional search space [Weise, 2009].

Nelder-Mead simplex optimization uses $n + 1$ points in the n-dimensional search space. These points form a polytope (i.e. a simplex), which is the generalization of a polygone, i.e. a line segment in 1D, a triangle in 2D, a tetrahedron in 3D, etc. Of interest are simplexes that are nondegenerate, i.e. that enclose a finite inner n-dimensional volume, since these simplexes have the property that their reflections through the opposite face are again nondegenerate simplexes, see Figure 8.2a.

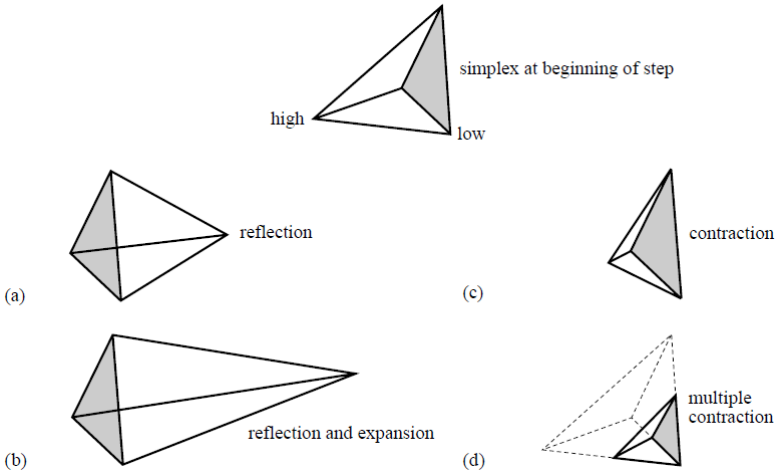


Figure 8.2: Possible outcomes for a step in the Nelder-Mead simplex method. The simplex at the beginning of the step, here a tetrahedron, is shown, top. The simplex at the end of the step can be any one of (a) a reflection away from the high point, (b) a reflection and expansion away from the high point, (c) a contraction along one dimension from the high point, or (d) a contraction along all dimensions towards the low point. An appropriate sequence of such steps will always converge to a minimum of the function [Press, 2007].

The idea of the Nelder-Mead simplex method is that the $n+1$ dimensional simplex

changes its shape depending on the objective function values at the $n+1$ points. By doing this in a clever way, the simplex will finally contract around the minimum in the search space.

At the start, an initial simplex needs to be defined by $n+1$ points. If one point is taken as initial starting point $\mathbf{P}_0$, then the other n points can be taken as follows [Press, 2007]:

$$\mathbf{P}_i = \mathbf{P}_0 + \lambda \mathbf{e}_i \tag{8.3}$$

where the $\mathbf{e}_i$'s are n unit vectors and λ is a constant which is the guess of the problem's characteristic length scale. It is also possible to have different λ_i's for each vector direction.

After initialization the method changes the simplex step by step, mostly by moving the point of the simplex with the largest function value through the opposite face of the simplex to a lower point. These steps are called reflections, and they maintain the simplex' nondegeneracy. Another possibility is to expand the simplex to take larger steps. Once the simplex reaches a "valley floor", it moves down into the valley, and finally the simplex will contract around the lowest point. The basic moves are shown in Figure 8.2.

The NM method has converged when the simplex changes shape only very little. In case convergence does not occur, the method should be restarted by taking the minimum of the non-converged simplex as the new initial start point $\mathbf{P}_0$.

The main advantage of the NM method is its simplicity and its ability to accurately find local minima. The drawbacks are that it can get stuck at local minima and that it can be slow because many function evaluations may be required [Press, 2007].

8.2.4. Differential Evolution

Differential Evolution (DE) was developed by Storn and Price [Storn and Price, 1997] to obtain a method which is both robust (finds the global optimum) and efficient (converges fast). DE makes use of a population of N solution candidates ($\mathbf{x}_1$,$\mathbf{x}_2$,...,$\mathbf{x}_N$), where each solution candidate is represented by a vector of the n optimization parameters.

First an initial population is generated, consisting of solution candidates which cover the entire solution space. After that, for each solution vector $\mathbf{x}_i$ a trail vector $\mathbf{y}_i$ is created. The candidate $\mathbf{x}_i$ is then replaced by the trail $\mathbf{y}_i$ if the objective function value of $\mathbf{y}_i$ is lower than the one of $\mathbf{x}_i$, i.e. if $J(\mathbf{y}_i) < J(\mathbf{x}_i)$. This process of creating trails and replacing solution candidates is repeated until the global minimum is found.

By comparing each solution vector only to one trail vector and vice versa, the DE algorithm prevents the search for the minimum to converge too fast.

In the original DE algorithm by Storn and Price [Storn and Price, 1997] the generation of trail vectors is done as follows. For each solution vector $\mathbf{x}_i$ three other solution vectors are randomly selected, $\mathbf{x}_{p(1)}$, $\mathbf{x}_{p(2)}$ and $\mathbf{x}_{p(3)}$. The weighted difference of two of these points is then added to the third point, which can be expressed as:

$$\hat{\mathbf{x}}_i = \mathbf{x}_{p(1)} + F * (\mathbf{x}_{p(2)} - \mathbf{x}_{p(3)}) \tag{8.4}$$

where F is a scaling factor. The trail point is then found by applying crossover to $\mathbf{x}_i$ and $\hat{\mathbf{x}}_i$ as follows:

$$y_i^j = \begin{cases} \hat{x}_i^j & \text{if } R^j \leq C_R \text{ or } j = I_i \\ x_i^j & \text{if } R^j > C_R \text{ and } j = I_i \end{cases} \tag{8.5}$$

where the superscript j indicates the j^{th} component of the corresponding vector, $R^j \in (0, 1)$ is drawn randomly for each j, C_R is a constant (e.g. 0.5) and I_i is a randomly chosen integer between $\in 1, 2, .., n$ which ensures that $\mathbf{y}_i$ gets at least one parameter from $\hat{\mathbf{x}}_i$.

The DE algorithm is very simple and straightforward and has proven to be very robust and efficient for complex optimization problems [Ali and Törn, 2004] [Vinkó et al., 2007].

Now that the optimization methods have been discussed, the optimization problem is considered regarding the way it can be solved best. This is investigated in the following chapter by assessing the optimization problem, tuning the optimization methods and finding the best way to apply them.

9. Optimization Problem Assessment

In this chapter the problem of finding the best velocity functions is assessed by analyzing the solution space, finding the right settings for the optimizers and picking the most suitable optimizer.

9.1. Solution Space

In order to assess the complexity of the optimization problem the solution space has been investigated first. This has been done by performing Monte Carlo evaluations in the search space, i.e. by picking random values for the free coefficients and subsequently computing the corresponding required ΔV's. Such evaluations have been performed for various (combinations of) velocity functions and for missions to Mars, Tempel 1 and Mercury (these test cases are discussed in Section 10.2).

For the evaluations the values of the free coefficients were picked randomly in a range which was fixed per target. In addition, also local searches has been performed by reducing the range of free coefficient values, to investigate the solution space around global optima.

The required ΔV's can be plotted against the coefficient values, which allows one to see what the solution space looks like. It is especially interesting to find out if there exist regions in the solution space where the ΔV values are low.

In Figure 9.1 the required ΔV is plotted against the value of the coefficient C_8. This plot is very noisy, since the plotted ΔV values also contain the contributions of all other free coefficients. Actually, since the plot only shows the variation of the required ΔV over C_8 it can be considered as a side-view of the solution space. The nice thing about this plot is that it clearly shows that there exists a valley regarding required ΔV at the lower side of the graph as well as at the upper side. This indicates that the required ΔV depends on the value of C_8. The valley at the lower side is most important since the objective is to minimize the required ΔV. Despite the presence of the 'noise' it is rather easy the find the optimal value of C_8 for minimal ΔV, which is about 400.

It was found that in general the valley at the upper side is more clear than the valley at the lower side, so the negative effect of a coefficient is more clear than the positive effect. Unfortunately, the locations of the valley minima on the upper and the lower side do not coincide, so the upper side valley can not be used to find the global minimum ΔV.

The values of the ΔV at the upper and lower side of the graph can differ largely. This is due to the fact that the search space is large. When the search space is reduced around the global optimum the difference between the top and bottom of the graph reduces. This indicates that the global minimum is located in an area where the required ΔV values are low. This observation is confirmed by Figure 9.2a.

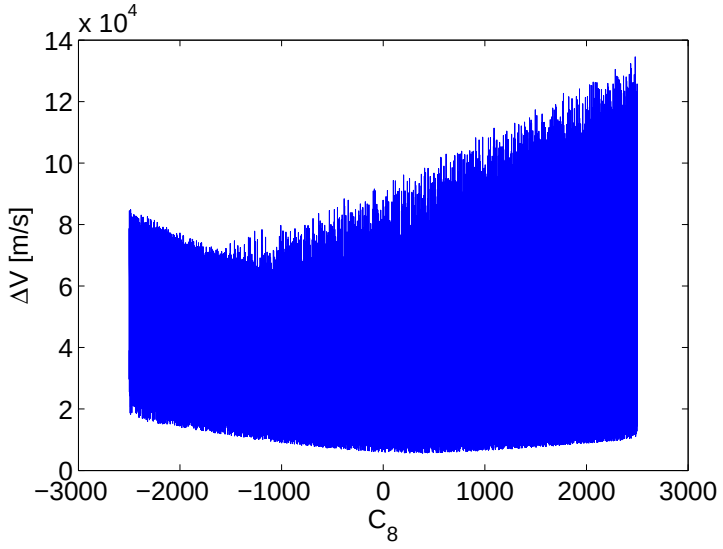


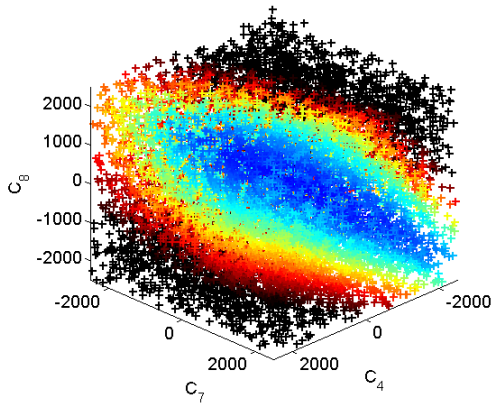
Figure 9.1: ΔV as function of coefficient C_8 for a mission to Mars with a TOF of 2 years, and $V_r = C_1 + C_2 \cos t + C_3 t + C_4 \sin t$ and $V_\theta = C_5 + C_6 t + C_7 \sin t + C_8 \cos t$.

In Figure 9.2 the required ΔV is plotted against three free coefficients; coefficient C_4 belongs to the sine base function in the radial velocity function and the other two coefficients belong to the sine and cosine base function in the normal velocity function. Figures 9.2a to c show the solution space in case of 3, 7 and 19 free coefficients, i.e. for 3, 7 and 19 DoF, respectively. These figures show that in case of only a few DoF there clearly exists a region is where ΔV is lowest, see Figure 9.2a. However, when the number of DoF is increased, such a region of low objective function values becomes less clear, see Figure 9.2b, or seems to be absent, see Figure 9.2c. As the region of low ΔV values becomes less clear, it is harder to find the global optimum, or maybe even practically impossible.

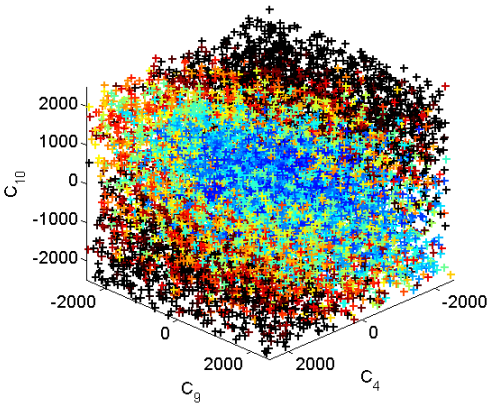
In addition to many Monte Carlo searches also a grid search has been performed. Using the results of the grid search it is possible to plot the ΔV as function of three coefficients while keeping the values of the other coefficients constant. From such solution-space plots it was found that when all coefficients but three are fixed, there exists a region of low ΔV values. This means that for each set of three coefficients there probably exists only one local optimum.

So, for a set of three free coefficients the global optimum can be found very simply in a smooth solution space. It is suspected that there also is a region of attraction when the number of free coefficients is increased. However, for a large number of free coefficients such a region cannot be found by Monte Carlo search with a limited number of trails, and the region is not visible anymore in a three-dimensional plot of the solution space (Figure 9.2c).

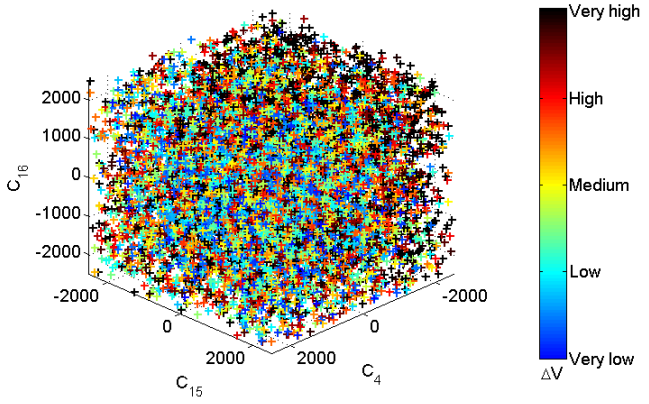
Coefficient value ranges In Appendix B the ranges of the optimal coefficient values are shown for different missions and different number of revolutions for the best results found using optimization. From the tables in this appendix it is clear that the range of optimal coefficient values depends on the mission and number of



(a) 3 DoF; $V_r = C_1 + C_2 \cos t + C_3 t + C_4 \sin t$, $V_\theta = C_5 + C_6 t + C_7 \sin t + C_8 \cos t$



(b) 7 DoF; $V_r = C_1 + C_2 \cos t + C_3 t + C_4 \sin t + C_5 t \sin t + C_6 t \cos t$, $V_\theta = C_7 + C_8 t + C_9 \sin t + C_{10} \cos t + C_{11} t \sin t + C_{12} t \cos t$



(c) 19 DoF; $V_r = C_1 + C_2 \cos t + C_3 t + C_4 \sin t + C_5 \sin 2t + C_6 \sin 3t + C_7 \cos 2t + C_8 \cos 3t + C_9 t^2 + C_{10} t^3 + C_{11} t \sin t + C_{12} t \cos t$, $V_\theta = C_{13} + C_{14} t + C_{15} \sin t + C_{16} \sin 2t + C_{17} \sin 3t + C_{18} \cos t + C_{19} \cos 2t + C_{20} \cos 3t + C_{21} t^2 + C_{22} t^3 + C_{23} t \sin t + C_{24} t \cos t$

Figure 9.2: Required ΔV as function of coefficients. Mars 2years.

revolutions. In addition, it can be noticed that the ranges of optimal coefficient values are small for good trajectories (with low ΔV) and they increase when bad trajectories (with high ΔV) also taken into consideration.

For the time-driven method the ranges of the optimal coefficient values have about the same magnitude, independent on the corresponding velocity function and base function. Only when bad trajectories are included, the range of the optimal coefficient values sometimes becomes large for specific velocity functions.

For the polar-angle-driven method, on the other hand, the ranges of the optimal coefficient values are very dependent on the corresponding velocity function. The range of optimal coefficient values for the time-evolution function are about a factor 1000 to 10000 smaller than the range of optimal radial coefficient values. Furthermore, the range of optimal coefficient values for the axial-velocity function

are about 2 to 10 times smaller than the range of the optimal radial coefficient values.

Furthermore, for both the time-driven and polar-angle-driven method, there seem to be no correlations between the optimal coefficient values for the different missions.

Finally, for the time-driven method the bounds of the optimal coefficient ranges are approximately three times the magnitude of the ΔV required for a Hohmann transfer to the corresponding target. This rule of thumb may be used as first guess to set the optimization parameter ranges in an optimizer. In addition, also the coefficient value ranges in Appendix B may be used to set the search space for an optimizer.

9.2. Optimizer Settings

The settings of an optimization algorithm have a large influence on the robustness and convergence speed of the optimization. For that reason, optimization using DE and NM have been investigated to obtain the best settings for the hodographic-shaping problem.

9.2.1. Differential Evolution settings

The DE optimizer which has been applied is taken from [Feoktistov, 2006] and is based on the algorithm that was first proposed by Price and Storn [Storn and Price, 1997]. This DE algorithm is an implementation of sequential DE, which means that the individuals immediately return in the population when they have evolved.

The parameters in this DE algorithm which can be set are the following:

- population size
- differentiation constant F
- crossover constant C_R
- number of generations
- lower bound
- upper bound

The population size together with the number of generations determine the number of objective-function evaluations which are carried out. The product of these parameters therefore needs to be as small as possible to minimize the required computation time. It was found that a small population with many generations converges faster than a large population with few generations. For six optimization parameters a population size of 25 together with 100 generations was found to perform best, for finding the optimal rendezvous-trajectory to Mars using the time-driven method. In case of more and less optimization parameters the number of generations and individuals can be increased of decreased, respectively.

The differentiation constant F is a scaling factor used for creating a solution candidate. The best value for F was found to be between 0.5 and 0.6, and therefore a value of 0.55 is applied. The crossover constant C_R determines the chance of crossover during the creation of a trail vector. When the chance of crossover is low, a large part of the trail vector is created using an already-known solution vector. When, on the other hand, the chance of crossover is high, then a newly-created solution candidate will mostly be used to create the trail vector. This causes the trail vectors to differ significantly from known solution vectors, and therefore the search for the optimum is done more globally. The chance of finding a solution which is close to the optimum is then larger, however, the results converge much slower. It was found that a large chance on crossover gives better results when only a few function evaluations are allowed. Therefore a value of 0.95 is used for the crossover constant C_R.

The lower and upper bound in the DE algorithm determine the search space, i.e. the allowed range of the free coefficient values. This range should be set large enough to be able to find all optimal values. However, it is desired to set the search space as small as possible, since otherwise a large number of function evaluations may be required to find the optimum. The results in Appendix B can be used to set the search space. Otherwise it is best to first quickly assess the objective function behavior for a specific mission, and then to choose lower and upper bounds such that all optimal free coefficient values are well inside the range.

Finally, next to parameters of the DE algorithm also the seed number of the applied random number generator can be set. However, it was found that the seed number has little influence on the outcome of the optimization, and therefore an arbitrary seed number can be used.

9.2.2. Nelder-Mead settings

The NM algorithm which has been applied was taken from [Burkardt, 2008]. This algorithm is a C++ version of the NM algorithm implemented by O'Neill [O'Neill, 1971] with comments by [Chambers and Ertel, 1974], [Benyon, 1976] and [Hill, 1978].

The parameters in this NM algorithm which can be tuned are:

- starting point for the iteration, i.e. initial guess
- step-size used to vary optimization parameters
- terminating limit for the variance of objective function values
- maximum allowed number of function evaluations
- number of iterations after which the convergence is checked
- factor to reduce step-size when solution has converged, but is not the optimum

For the starting point of the optimization the null vector is used. This means that the initial guess actually is the lowest-order solution. However, if a good

initial guess is available then that guess can be used. The step-size used to vary optimization parameters determines whether the search is (very) local or more global. A small step-size results in a very local search whereas a large step-size results in a more global search and usually faster convergence. However, when the step-size is chosen too large, one runs the risk that the optimizer steps over the global optimum and will find another local optimum. The step-size is therefore chosen to be an order of magnitude smaller than the values of the optimal free coefficients of the best solutions.

The optimization is terminated when either the maximum allowed number of function evaluations is exceeded, or when the variance of the found objective-function values is lower than required, i.e. when the solutions have converged. The limit for the variance of objective-function values is set very low (10^{-9}) to be sure that solutions are fully converged. The maximum number of function evaluations, on the other hand, is set such that at least all good solutions have converged; 5000 function evaluations usually suffices.

After a specified number of iterations the solutions are checked on convergence. If the solutions have converged then it is checked whether the found solution is a local optimum or not. If the solution is an optimum the optimization is terminated, otherwise the search is restarted with a reduced step-size and with the best found solution as initial guess. The step-size reduction factor is set to 0.1 and the converge check is done every 10 iterations. The advantage of checking the convergence every 10 iterations is that it can be noticed quickly when the optimizer gets stuck.

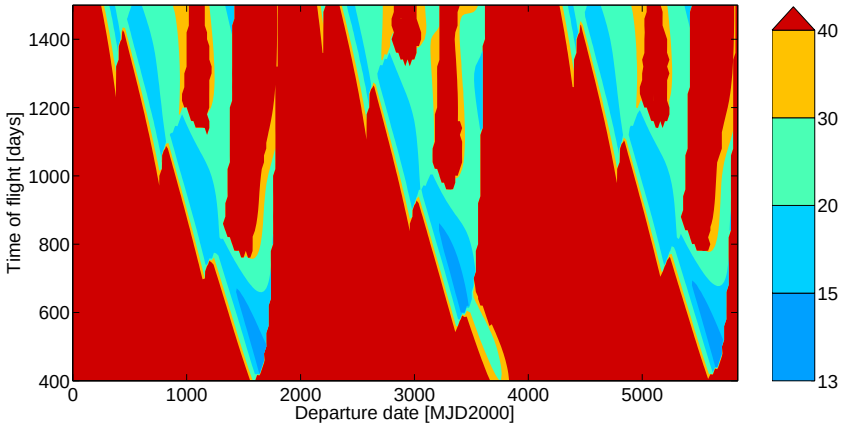
9.3. Differential Evolution versus Nelder-Mead

A global optimizer like DE is in general very robust, i.e. the global optimum is found very often. A local optimizer like NM, on the other hand, can get stuck at a local minimum and is therefore in general less robust. However, when the optimization problem is simple a local optimizer is much faster and accurate in finding the optimum than a global optimizer. In order to find out whether it is better to use DE or NM, both methods have been applied to the problem of optimizing velocity functions. The obtained results and number of required function evaluations of the DE and NM optimizers are compared and discussed here. For these tests the lowest-order solution was used as initial guess for the NM optimizer and the lowest-order solution was also included in the initial population for the DE optimizer.

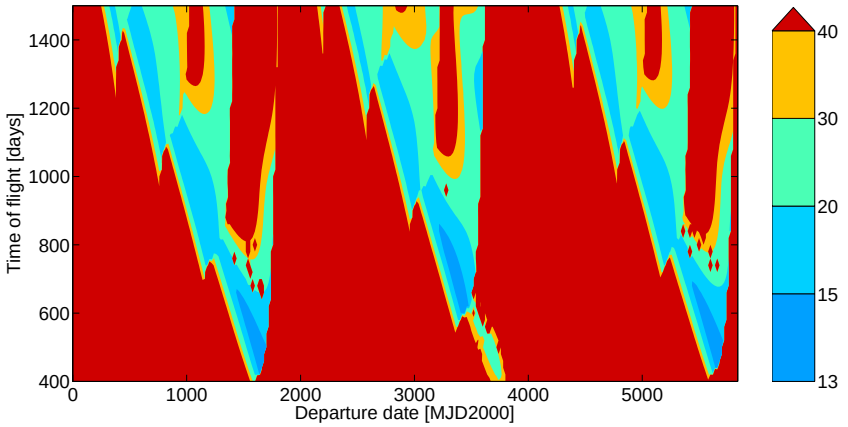
Two test cases have been used to compare the DE and NM optimizers. The first is optimizing trajectories to Tempel 1 for the polar-angle-driven method. Optimizing trajectories for this case was namely found to be the most difficult. The geometry of Earth-Tempel 1 missions is rather complex, and the optimization of free coefficients for the polar-angle-driven method is harder than for the time-driven method, since the search space is usually larger and the optimal values

differ more. In addition, both optimizers have also been compared for an easier case, namely a mission to asteroid 1989ML for the time-driven method.

First of all, it was found that both DE and NM are not always able to find the global optimum when the number of allowed functions evaluations is limited. This is especially the case when the search space is set too large for the DE optimizer and when the step-size set in the NM optimizer is not suitable. Secondly, NM was found to be much faster in obtaining optima than DE, especially when the required ΔV is low.



(a) Nelder-Mead



(b) Differential Evolution

Figure 9.3: ΔV plots for a mission to Tempel 1 with N=1 computed using a) Nelder-Mead and b) Differential Evolution optimizers for 6 DoF (R=CPowPow2Pow3Pow4, T=CPowPow2Pow3Pow4 and Z=CosP3CosP3SinP4CosP4Sin).

For the trajectories to Tempel 1 computed by the polar-angle-driven method, NM was found to be more robust in finding the optima in regions of low ΔV than DE. This can be seen in Figure 9.3 where the ΔV plots obtained using DE and NM are shown for $N = 1$. Clearly, there are some spots in both the NM and DE plot where the optimum is not found. For NM these spots occurs near regions of high ΔV. NM was found to be unable to find the optimum trajectory when the

corresponding optimal required ΔV is very large. For DE the spots do not always occur close to regions of high ΔV, however, not close to regions of low ΔV either. So, regarding robustness NM and DE perform equally well/badly. In addition to that, it was found that NM requires a much lower number of evaluations than DE, namely 2048 (on average) for NM against 20,000 for DE.

For a mission to asteroid 1989ML using the time-driven method, NM was found to perform much better than DE. First of all, the number of evaluations required by NM are lower; 1262 (on average) for NM against 5000 for DE. In addition, NM is more robust; overall 35% of the optimum ΔV values found by NM are more than 1% better than the ones found by DE whereas only 1% of the optimum ΔV values are more than 1% worse. Furthermore, for the best 10% of the ΔV values found by NM the maximum difference with DE is 9 m/s whereas for the best 10% of the ΔV values found by DE the maximum difference is 2401 m/s. So, NM is most robust when the required ΔV is low.

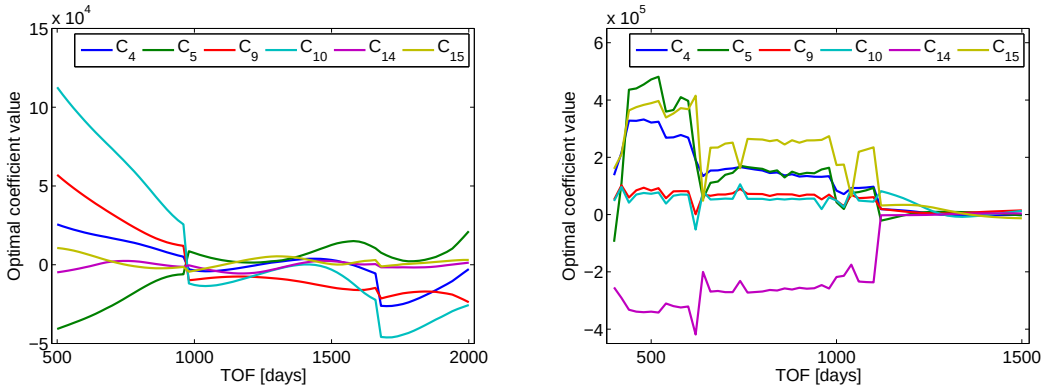
Concluding, it was found that NM almost always finds the optimal trajectory in regions of low ΔV. Only for more complex optimization problems NM does not find to global optima in regions of high ΔV. DE, on the other hand, is only robust when the search space is small and/or the allowed number of function evaluations is large. However, when a (too) small search space is used, it can occur that the optimal coefficient values are not within the search space. Furthermore, allowing for more function evaluations reduces the optimization speed. All together, since it is most robust (in regions of low ΔV) and requires less function evaluations, NM is chosen as optimizer for the hodographic-shaping method.

9.4. Initial Guesses

After optimizing many trajectories for various missions it was found that the optimal values of the free coefficients vary gradually over TOF and departure date. This actually seems logical since when the mission geometry changes little the optimal trajectories will be similar and thus the optimal free coefficient values will not differ much.

Two examples of the variation of the optimal coefficient values over TOF are shown in Figure 9.4. Figure 9.4a shows the optimal coefficient values against TOF for a fixed departure date and a fixed number of revolutions for a mission to Mars. Clearly the optimal coefficient values vary gradually and smoothly except for two locations. These discontinuities are the result of a sudden change in the geometry of the problem. Between a TOF of 960 and 980 days and 1660 and 1680 days the transfer angle ψ crosses the value 2π and therefore its value drops to zero. As a consequence, the final polar angle condition changes discontinuously[1].

[1]The range of the transfer angle ψ is $[0, 2\pi)$ and therefore ψ varies discontinuously when it crosses the value 2π. As a consequence, the final polar angle θ_f $(= \psi + 2\pi N)$ also varies discontinuously when the number of revolutions N is fixed.



(a) Mars with $N = 2$ and departure date is 8705 MJD2000 with V_r=CPowPow2PSin05PCos05, V_θ=CPowPow2PSin05PCos05 and V_z=CosR5P3CosR5P3SinR5P4CosR5P4SinR5

(b) Tempel1 with $N = 1$ and departure date is 4839 MJD2000 with V_r=CPowPow2PSin05PCos05, V_θ=CPowPow2PSin05PCos05 and V_z=Cos05PCos05PSin05P2Cos05P2Sin05

Figure 9.4: Optimal free coefficient values for different TOF for two different missions.

In Figure 9.4b the optimal coefficient values are shown for a mission to Tempel 1. Here the transfer-angle discontinuity occurs between a TOF of 1100 and 1120 days. However, before this discontinuity the optimal coefficient values do not change very smoothly. It was found that the ΔVs which correspond to all these trajectories are very large. After the transfer-angle discontinuity (i.e. when TOF>1120 days) the optimal coefficient values vary smoothly and the corresponding ΔVs have low values. Despite the fact that the coefficient values do not always vary smoothly, clearly a trend can be observed in the optimal coefficient values over TOF.

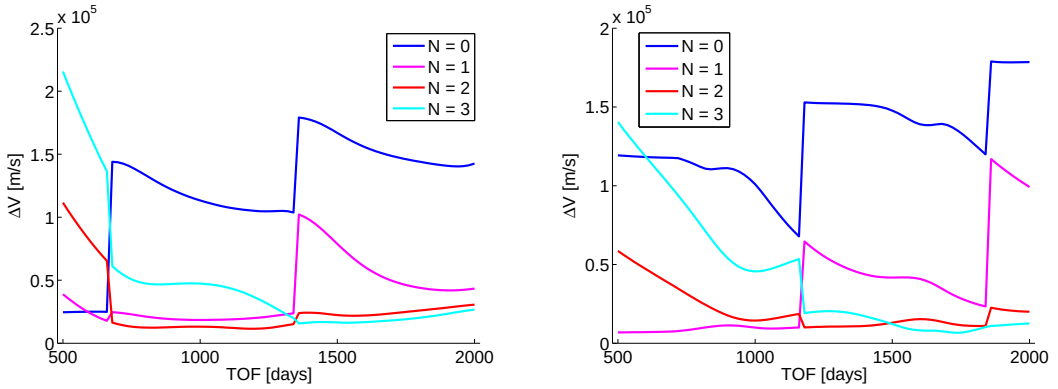
Since the optimal coefficient values vary smoothly or at least follow some trend, one can use previously obtained optimal coefficient values as initial guess for optimization. In this way the optimization of free coefficients can be speeded up. Such initial guesses have been implemented for NM optimization: simply the optimal coefficients of the velocity functions of the previously applied TOF are used as initial guess. Only when a transfer-angle discontinuity occurs, or when a new loop over the TOF is started, an initial guess is not used. The application of initial guesses was found to speed up the optimization significantly; optima are found 2 to 3 times faster.

Initial guesses have been used both for the time-driven and polar-angle-driven method. However, when using initial guesses for polar-angle-driven method sometimes non-optimal results were found for (difficult) missions to Tempel 1 and Mercury. Fortunately, one can easily observe when non-optimal results are found, since in such a case the ΔV plots look discontinuous. So, for the polar-angle-driven method initial guesses have not always been used.

Since optimal coefficient values vary gradually over both TOF and departure date, one may also loop over departure date instead of TOF when using initial guesses. However, the optimal coefficient values usually change more gradually over TOF than over departure date, since over TOF the geometry changes less. Therefore it

is better to loop over TOF.

In addition to the application of initial guesses, one can also choose to increase the number of revolutions when a transfer-angle discontinuity occurs. By doing this the final polar angle condition will change continuously over TOF. As a result the previously found optimal coefficients values can be used as initial guess even after a transfer-angle discontinuity.



(a) 7545 MJD2000 (b) 7345 MJD2000

Figure 9.5: Optimal ΔV for different TOF for two departure dates for a mission to Mars with V_r=CPowCosPCos, V_θ=CPowCosPSinPCos and V_z=CosR5Pow6P6CosR5.

In Figure 9.5a ΔV is plotted against TOF for different number of revolutions. It is clear that when one changes the number of revolutions at a transfer-angle discontinuity, one continues to follow the ΔV trend which one is on. If one does, however, not change the number of revolutions, the ΔV will make a discontinuous jump to another trend. By changing the number of revolutions at a transfer-angle discontinuity one can follow the optimal ΔV trend, and as a result one only needs to search the flight window once and not multiple times for different number of revolutions. This approach can therefore speed up the optimization significantly.

In general, this approach works well if one starts at the proper number of revolutions at the beginning of the TOF loop. However, the ΔV trend which is optimal regarding ΔV can differ over TOF. Figure 9.5b shows that between a TOF of 1500 and 1800 days another ΔV trend ($N = 2, 3$) is optimal than elsewhere ($N = 1, 2$). As a consequence, it is possible that when one follows a particular trend line one will not find the optimal ΔV for all TOF. For that reason, the decision was made to search the flight window multiple times for different number of revolutions instead of changing the number of revolutions during the loop over TOF.

In conclusion, the optimization of the free coefficients is done fast and robust by applying an NM optimizer and using initial guesses for the free coefficients when looping over TOF.

10. Test Cases

The goal of the hodographic-shaping methods is to generate feasible and near-optimal solutions for low-thrust trajectories in an efficient way. Whether the method is able to do this and to which extent, needs to be evaluated. This is done by applying the method to several test cases and comparing the results to solutions obtained by other methods.

First of all, it is interesting to find out whether the hodographic-shaping methods perform better or worse than other shape-based methods. These shape-based methods have already been discussed in Chapter 4 and include the exponential sinusoid, inverse polynomial, and spherical, pseudo-equinoctial and pseudo-spectral shaping.

Secondly, by comparing the results of the hodographic-shaping method to fully optimized trajectories, the optimality of the solutions can be investigated. Various design and optimization programs exist for low-thrust trajectories, such as STOUR-LTGA, GALLOP and DITAN. The results of these programs have already been used frequently to compare and analyze shape-based low-thrust methods.

10.1. Low-Thrust Trajectory Comparison

In literature results of low-thrust shape-based methods and low-thrust optimization algorithms can be found which can be used to assess the performance of the hodographic-shaping method. However, comparison with results from literature can only be done properly if all parameters and conditions which were assumed are known. The parameters and conditions that need to be known in case of low-thrust trajectories are the following:

- departure time, t_0
- departure position, usually the position of the departure celestial body
- (hyperbolic excess) velocity at departure, both magnitude and direction, $\mathbf{V}_{\infty,0}$ or $\mathbf{V}_0$
- arrival time or time of flight, t_f or TOF
- arrival position, usually the position of the target celestial body
- (hyperbolic excess) velocity at arrival, both magnitude and direction, $\mathbf{V}_{\infty,f}$ or $\mathbf{V}_f$
- number of complete revolutions, N
- initial mass, M_0
- performance of the propulsion system: 2 out of 3 of the following parameters are required and the third can be computed using Eqs. (3.2) and (3.3):

- specific impulse, I_{sp}
- maximum thrust, T_{max}
- input power, W

In case the required ΔV is optimized, the performance of the propulsion system and the initial spacecraft mass does not need to be known. Furthermore, when rendezvous is considered the initial/final velocity of the spacecraft equals the velocity of the departure/target celestial body.

A brief overview of low-thrust design and optimization methods for which results are available in literature is given next. For each method the most important assumptions are stated as well as the test cases to which they were applied.

- The exponential sinusoid has been implemented by Petropoulos in a trajectory design program called STOUR-LTGA (Satellite Tour Design Program including Low-Thrust and Gravity Assists) [Petropoulos and Longuski, 2004]. The test cases used for this program, however, include gravity assists and are therefore not of interest. Unfortunately, Petropoulos does not provide clear results of the exponential sinusoid for direct transfers. Therefore the exponential sinusoid cannot be used for comparing direct transfer trajectories.
- The inverse polynomial has only been tested for a direct transfer to Mars with a rendezvous with Mars ($V_{\infty,0} = 1.6$ km/s) [Wall, 2008]. No maximum thrust level was implemented and the trajectories were optimized for minimum propellant mass fraction.
- Spherical shaping has been tested for rendezvous missions with Mars, 1989ML, Tempel 1 and Neptune (for Neptune $V_{\infty,0} = 3.0$ km/s) [Novak and Vasile, 2011]. No maximum thrust level was implemented and the trajectories were optimized for minimum required ΔV.
- Pseudo-equinoctial shaping has been tested by Vasile and De Pascale by implementing it into IMAGO (Interplanetary Mission Analysis Global Optimization) [De Pascale and Vasile, 2006]. IMAGO was tested for a rendezvous with Mars and Mercury. The trajectories were optimized for minimum propellant mass fraction including penalty functions for violating a maximum thrust level and time of flight constraint. In addition, Novak tested the pseudo-equinoctial shaping method for rendezvous with Mars, 1989ML, Tempel 1 and Neptune (for Neptune $V_{\infty,0} = 3.0$ km/s), without thrust level constraints and optimizing for minimum ΔV [Novak and Vasile, 2011].
- The pseudo-spectral method was tested by De Vogeleer for a rendezvous with Mars ($V_{\infty,0} = 0.86$ km/s) and Tempel 1 ($V_{\infty,0} = 0.42$ km/s) [De Vogeleer, 2008]. The trajectories were optimized for both minimum ΔV and minimum propellant mass and with or without penalty functions for extending the maximum thrust level.
- DITAN (Direct Interplanetary Trajectory Analysis) [Vasile, 2006] is a local

optimizer which has been used to improve the results of pseudo-equinoctial and spherical shaping. Vasile and De Pascale did this for a rendezvous with Mars, asteroid Dionysus [Vasile et al., 2007] and Mercury [De Pascale and Vasile, 2006]. In addition, Novak used DITAN for optimizing trajectories for rendezvous with Mars, 1989ML, Tempel 1 and Neptune (for Neptune $V_{\infty,0} = 3.0$ km/s) [Novak and Vasile, 2011]. The optimizations all included a thrust-level constraint and the objective was maximizing the final mass.

- GALLOP (Gravity-Assist Low-thrust Local Optimization Program) [McConaghy et al., 2002] was used to improve near-optimal low-thrust trajectories. Especially trajectories including gravity-assists have been tested. However, also a rendezvous trajectory with Tempel 1 ($V_{\infty,0} = 0.8$ km/s) was optimized for minimum propellant mass fraction [McConaghy et al., 2003].

For missions where no $V_{\infty,0}$ was specified, $V_{\infty,0}$ equals zero. Furthermore, for rendezvous trajectories $V_{\infty,f} = 0$. Finally, for all test cases mentioned above the three-dimensional case is considered, so no planar transfers were assumed.

10.2. Test Cases

Four mission scenarios have been selected to test the shaping method, namely rendezvous missions from Earth to Mars, to the near-Earth asteroid 1989ML, to comet Tempel 1 and to Mercury. These missions have been selected since they include both inner and outer targets and because they have different orbit characteristics. In addition, all these missions have already been used to test other shaping methods and therefore the results for these missions can be compared. Table 10.1 shows the methods which have already been tested on these missions. All test cases in Table 10.1 have been used for at least two different methods, so they are all suitable for comparison.

Table 10.1: Test cases for direct low-thrust transfer trajectories; † optimized including maximum thrust level constraint.

Method	Mars	Mercury	Tempel 1	1989ML
Inverse polynomial	✓			
Spherical	✓		✓	✓
Pseudo-equinoctial	✓	✓†	✓	✓
Pseudo-spectral†	✓		✓	
DITAN†	✓	✓	✓	✓
GALLOP			✓	

In principle, the solutions found by DITAN are fully optimized ones and can therefore be used as benchmark. So, for all test cases a proper benchmark is available.

The orbital elements of Earth, Mars and Mercury can be found in Table 10.2 and those of 1989ML and Tempel 1 in Table 10.3. These orbital elements have been taken from NASA JPL's Solar System Dynamics website; [NASA - JPL, 2012a] and [NASA - JPL, 2012b]. Unfortunately, it is unknown whether these orbital elements (fully) agree with the ones used to test the methods in Table 10.1, since those were not (fully) provided in the corresponding references.

Table 10.2: Keplerian elements and their rates of selected planets, with respect to the mean ecliptic, equinox and epoch of J2000: semi-major axis, eccentricity, inclination, mean longitude, longitude of perihelion and longitude of the ascending node [NASA - JPL, 2012a].

	a [AU], [AU/Cy]	e [-], [/Cy]	i [deg], [deg/Cy]	L [deg], [deg/Cy]	$\bar{\omega}$ [deg], [deg/Cy]	Ω [deg], [deg/Cy]
Mercury	0.38710	0.20564	7.00559	252.25167	77.45772	48.33962
	0.00000	0.00002	-0.00590	149472.67487	0.15940	-0.12214
EM Bary	1.00000	0.01673	-0.00054	100.46692	102.93006	-5.11260
	0.00000	-0.00004	-0.01337	35999.37306	0.31795	-0.24124
Mars	1.52371	0.09337	1.85182	-4.56813	-23.91745	49.71321
	0.00000	0.00009	-0.00725	19140.29934	0.45224	-0.26852

Table 10.3: Keplerian elements of asteroid 1989ML and comet Tempel-1, with respect to the mean ecliptic and equinox of J2000: semi-major axis, eccentricity, inclination, mean longitude, longitude of perihelion, longitude of the ascending node and epoch [NASA - JPL, 2012b].

	a [AU]	e [-]	i [deg]	Ω [deg]	ω [deg]	M [deg]	Epoch [MJD]
1989ML	1.27254	0.13671	4.37800	104.38253	183.23998	287.92582	56000
Tempel-1	3.12338	0.51734	10.52975	68.93384	178.91137	162.40622	54466

For each mission the search for optimal trajectories is done in a limited flight window. The chosen flight windows are shown in Table 10.4. For the missions to Mars, asteroid 1989ML and comet Tempel 1, the flight windows are applied which were used by Novak to test the spherical shaping method [Novak and Vasile, 2011]. The flight window for Mercury equals the flight windows used by De Pascale and Vasile to test the pseudo-equinoctial shaping method [De Pascale and Vasile, 2006].

For the search through the flight windows, the departure date and TOF ranges are discretized with a 20-day step-size for higher-order solutions and with a 10-day step-size for lowest-order solutions. A smaller step-size is selected for lowest-order solutions, since they are more dependent on the departure date and TOF than higher-order solutions, and computation times are not an issue for the lowest-order case.

Table 10.4 also shows the orbital periods and synodic periods of the target bodies and Earth. These periods were computed using Eqs. (2.19) and (2.20). The orbital

Table 10.4: Departure date and TOF ranges of flight windows for different test cases and orbital and synodic periods of target bodies (with respect to Earth).

	Departure date [MJD2000]	TOF [days]	Orbital period [days]	Synodic period [days]
Mars	7304 - 10225	500 - 2000	687	780
1989ML	7304 - 10225	100 - 1000	524	1204
Tempel-1	0 - 5845	400 - 1500	2016	446
Mercury	3285 - 5475	100 - 1400	88	116

and synodic periods are of interest since they determine the periodicity at which the mission geometry changes.

Next the orbits of the four target bodies are briefly discussed.

Mars The orbit of Mars has a low eccentricity and is almost in the same plane as Earth's orbit. The semi-major axis, on the other hand, is 1.5 times Earth's. Therefore for a mission to Mars, the orbital plane does not have to be changed much, however, the pericenter and apocenter distances have to be raised significantly.

1989ML Asteroid 1989ML is a near-Earth asteroid with its pericenter close to Earth's orbit. Compared to Mars' orbit, the orbit of 1989ML is more inclined, more eccentric and closer to Earth. The higher eccentricity and inclination of 1989ML make the asteroid a more challenging target than Mars, however, the required semi-major axis change is rather small.

Tempel 1 Comet Tempel 1 has a highly eccentric and relatively highly inclined orbit. Its pericenter is approximately at Mars' orbit and its apocenter at Jupiter's. Especially the high eccentricity of Tempel 1's orbit makes a mission to the comet more challenging than a mission to Mars or 1989ML. Furthermore, of the four target bodies Tempel 1 has the highest inclination. So, regarding required orbital-plane change this mission is the most interesting.

Mercury Mercury is the only inner body considered in the test cases. A mission to an inner body is especially of interest since instead of an increase in orbital period, a decrease in orbital period is required. Therefore it seems logical that for a mission to an inner body the number of completed revolutions during the transfer is relatively high. As a result, the mission to Mercury is suited to test the multi-revolution abilities of the hodographic-shaping method.

Now that all test cases have been defined and discussed, the results of hodographic-shaping method for all tests will be discussed in the next chapter.

11. Results

In this section the results obtained by the hodographic-shaping methods are discussed. First a brief introduction is given, after which the results are discussed and finally conclusions are drawn.

11.1. Introduction

In this chapter it is investigated whether the hodographic-shaping methods are capable of providing good results. First of all, it is interesting to find out whether trajectories with near-optimal required ΔV and low f_{max} can be obtained, and which velocity functions should be used for this. In addition, the optimal departure dates and TOF are considered as well as the best trajectories and the corresponding thrust profiles. Finally, also the computational speed of the hodographic-shaping methods is assessed.

The results of the hodographic-shaping methods are presented as follows. The methods have been tested for all four different cases which were treated in the previous chapter. For each test case both the time-driven and polar-angle-driven method are considered. In addition, the lowest-order and higher-order solutions of both methods are discussed and compared. Finally, for each test case the performance of hodographic-shaping methods is assessed by comparing the results with results of other low-thrust (shaping) methods at the end of every test-case section.

For the assessment and comparison of the results, three aspects are considered, namely the required ΔV, the maximum thrust acceleration and the computational effort. The results regarding required ΔV and maximum thrust acceleration are discussed for each test case and for the time-driven and polar-angle-driven method separately. The computational effort, on the other hand, is discussed in a separate section, since it is not purely test-case dependent.

All higher-order results presented in the chapter have been obtained using an NM optimizer. In addition, for all optimizations, except for the polar-angle-driven method for missions to Tempel 1 and Mercury, initial guesses have been used, see Section 9.4. Furthermore, the maximum number of allowed function evaluations was set to 5000.

Not all results generated in the research are discussed in this report. For example Figure 11.1 shows the Pareto fronts for five different velocity-function combinations. In addition to these fives combinations, many more combinations are possible and many of them have been tested. However, in this chapter only the velocity-function combinations which give good results are discussed. So, only the results of the best lowest-order solutions and the best higher-order solutions are shown in this chapter.

Now the discussion of the results is started by considering the results for a mission to Mars.

11.2. Mars

As an ultimate benchmark for the mission to Mars the ΔV required for a 2D Hohmann transfer is considered. For such a Hohmann transfer from a circular Earth orbit to the apocenter of an elliptical Mars orbit a ΔV of 5.50 km/s is needed and the corresponding time of flight is 281 days. This ΔV is a lowest possible velocity change required to rendezvous with Mars[2]. The transfer which is considered next is in three dimensions, so requiring an inclination change of 1.85 deg, therefore the true minimum required ΔV will be a bit larger. DITAN found a low-thrust trajectory to Mars requiring a ΔV of only 5.66 km/s [Vasile et al., 2007].

11.2.1. Shaping as function of time

In the following the results of hodographic shaping as a function of time are discussed. First the solutions for the lowest-order case are considered and subsequently the higher-order solutions.

Lowest-order solutions The velocity functions of lowest-order solutions do not contain free coefficients. So, for a specific departure date and TOF, the resulting trajectory only depends on the applied velocity base functions. To find out which velocity functions provide the best trajectories, several combinations of different base functions have been investigated. Since the objective is to minimize the ΔV and, in addition, a low required thrust acceleration is desired, Pareto fronts are used to compare the results of different velocity functions. These Pareto fronts show the best results with respect to both ΔV and maximum required thrust acceleration. By plotting multiple Pareto fronts for different velocity functions in one graph, one can quickly see which velocity functions perform best regarding minimum ΔV, thrust acceleration or both. In this way it is also possible to observe which velocity base functions result in good and which in bad solutions such that velocity functions can be made as good as possible.

For the lowest-order solution three different aspects have been investigated: the effect of different radial- and normal-velocity-function combinations, the effect of different power orders for the axial velocity function, and the effect of the applied number of revolutions. These three aspects are discussed separately in the following.

[2]Starting at Earth with Earth's orbital velocity and ending at Mars with Mars' orbital velocity in a heliocentric reference frame.

Radial- and normal-velocity functions Figure 11.1 shows the Pareto fronts for different radial- and normal-velocity functions, whereas the axial-velocity function is fixed. From Figure 11.1 it is clear that the combination CPowPow2-CPowPow2 results in the best trajectories regarding both ΔV and thrust acceleration.

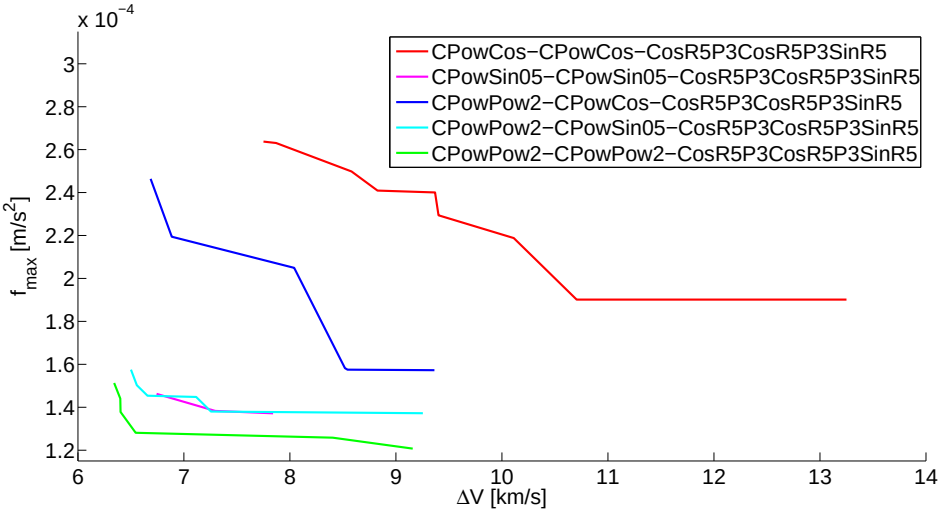


Figure 11.1: Pareto fronts of lowest-order time-driven solutions with different radial- and normal-velocity functions for a mission to Mars.

Table 11.1: Departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories found by lowest-order time-driven solutions with different radial- and normal-velocity functions for a mission to Mars.

V_r	V_θ	V_z	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPowCos	CPowCos	CosR5P3CosR5P3SinR5	9265	1070	7.751	2.64
CPowSin05	CPowSin05	CosR5P3CosR5P3SinR5	10035	1070	6.742	1.46
CPowPow2	CPowCos	CosR5P3CosR5P3SinR5	9245	1090	6.686	2.46
CPowPow2	CPowSin05	CosR5P3CosR5P3SinR5	10025	1050	6.500	1.58
CPowPow2	CPowPow2	CosR5P3CosR5P3SinR5	10025	1050	6.342	1.51

The characteristics of minimum-ΔV trajectories obtained with different velocity-function combinations are shown in Table 11.1. The best obtained ΔV is 6.34 km/s which is quite a bit higher than the 5.66 km/s found by DITAN. So the obtained trajectories are definitely not optimal. On the other hand, the maximum required thrust accelerations are low, which is very nice. The time of flight for all trajectories is approximately the same. For the departure date, however, two different ranges were found where the required ΔV is low, namely around 9255 and 10030 MJD2000. The difference between these range is approximately 780 days, which is equal to the synodic period of Earth-Mars, see Table 10.4. This will be further discussed later in this section.

Furthermore, Table 11.1 shows that the application of power terms in the radial velocity results in lower ΔV's than using trigonometric functions. Furthermore, considering trigonometric terms, the application of Sin05 terms in both the radial and normal velocity functions give better results than applying Cos terms.

Axial velocity function In Section 5.4.2 it was discussed that the order of the axial function affects the change of the inclination over time. In order to see which order provides the best results, Pareto fronts have been generated for axial-velocity functions with different power orders. These Pareto fronts are shown in Figure 11.2 (for $N = 0 - 3$).

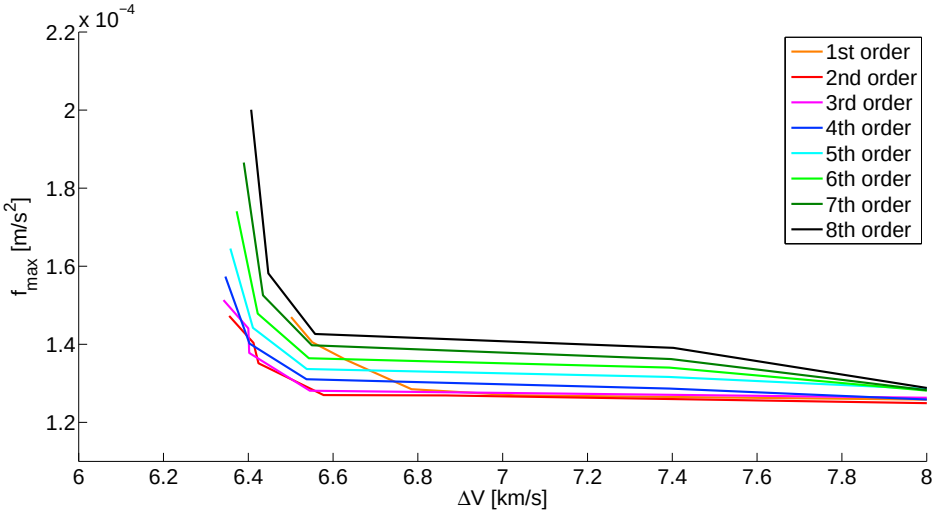


Figure 11.2: Pareto fronts of the lowest-order time-driven solution CPowPow2-CPowPow2-CosR5PXCosR5PXSinR5 for different axial power orders X for a mission to Mars.

It is clear that the application of 3rd-order power terms results in the lowest ΔV, i.e. 3rd order powers are most efficient in changing the inclination regarding ΔV (the Pareto front corresponding 3rd-order powers in Figure 11.2 is equal to the Pareto front of CPowPow2-CPowPow2-CosR5P3CosR5P3SinR5 in Figure 11.1). It should, however, be noted that for different power orders the difference in required ΔV is rather small. Considering the required thrust acceleration, on the other hand, a power order of 2 can best be taken.

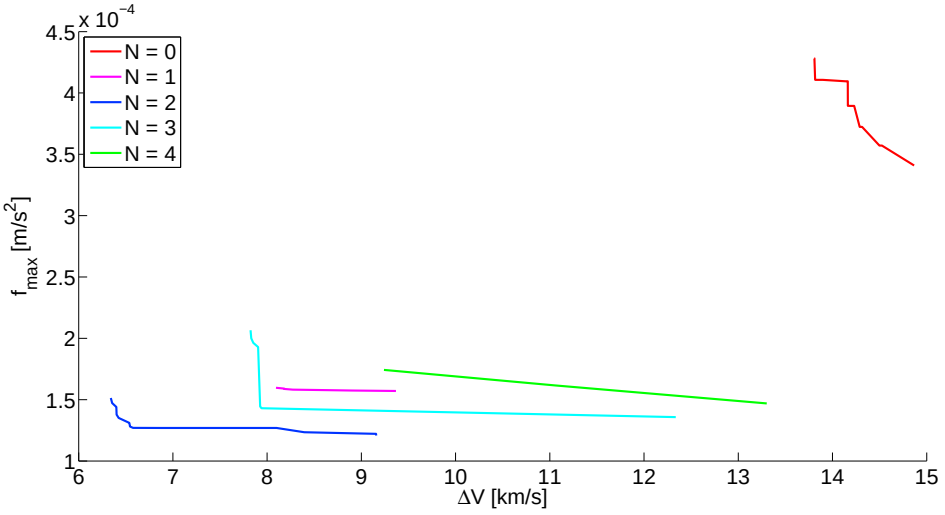


Figure 11.3: Pareto fronts of the lowest-order time-driven solution CPowPow2-CPowPow2-CosR5PCosR5PSinR5 for different number of revolutions N for a mission to Mars.

Number of revolutions The Pareto fronts for different axial power orders were created with $N = 0-3$. However, if one would only plot the results for $N = 2$ then the same figure would be obtained. A number of two revolutions is namely optimal regarding both ΔV and thrust acceleration, as can clearly be seen in Figure 11.3. This figure shows Pareto fronts for different number of revolutions. First of all, this figure shows clearly that for $N = 0$ very bad results are obtained. Furthermore, one would expect that the maximum required thrust acceleration decreases with increasing flight time, since then more time is available to generate the required velocity change. Similarly, when the time of flight increases usually also the number of revolutions increases, and therefore one would expect that the maximum required thrust acceleration reduces with increasing number of revolutions. Surprisingly, this is not the case since two revolutions is best with respect to the maximum required thrust acceleration. Therefore one can conclude that for the lowest-order case the method is not well able to produce optimal many-revolution trajectories.

It is clear that obtained results strongly depend on the chosen number of revolutions. The Pareto fronts for different power orders in the axial velocity function in Figure 11.2 were, however, created using the results for all possible number of revolutions. It appeared that the most efficient power order may differ for different number of revolutions. The power order of three, which was found to be most efficient overall, was found to correspond to two revolutions, the most efficient number of revolutions. In addition, for zero and one revolutions 2^{nd}-order powers are best, and for three revolutions 5^{th}-order powers. Nevertheless, the decision was made to fix the order of the axial powers for all number of revolutions for simplicity, since the differences in obtained ΔV and thrust acceleration for different orders are rather small.

In the Pareto fronts for different number of revolutions (Figure 11.3) some bumps can be spotted, especially in Pareto fronts of zero and two revolutions. These bumps are caused by two different things. Firstly, some bumps are caused by the fact that the results for different axial power orders were used to create the Pareto front. So, actually one looks at the combined Pareto front for different axial power orders. The bumps in the fronts are therefore jumps from one section to another for different axial power orders. Secondly, the search in the flight window over departure date and TOF is a discrete one. The change of required ΔV and thrust acceleration over departure date and TOF may be continuous. However, since the departure dates and TOF are discretized the change in ΔV and thrust acceleration can be discontinuous. Especially, since there are no DoF the resulting trajectories are very dependent on the departure date and time of flight and therefore bumps and steps may be seen in the Pareto fronts.

Figure 11.4 shows the required ΔV for all departure date and TOF combinations for the lowest-order solution: CPowPow2-CPowPow2-CosR5P3CosR5P3SinR5. This ΔV plot clearly shows several regions where the required ΔV is low. Furthermore, a repetitive pattern can be observed which is periodic with the synodic period of

Earth and Mars of 780 days.

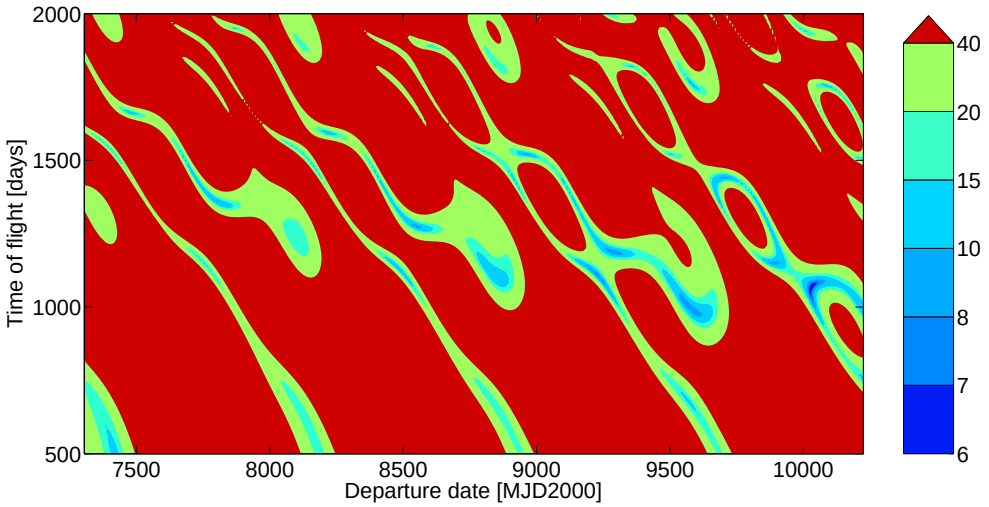


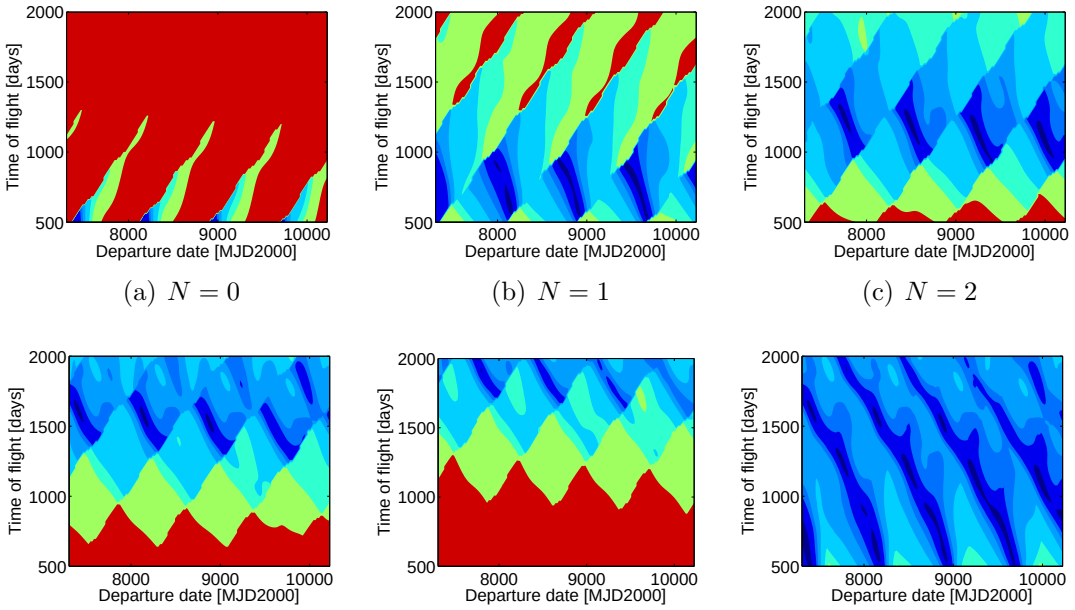
Figure 11.4: ΔV plot for the lowest-order time-driven solution CPowPow2-CPowPow2-CosR5P3CosR5P3SinR5 for a mission to Mars with $N = 0 - 5$.

ΔV plots, like the ones shown in Figures 11.4, 11.7 and 11.8, are created as follows. First, for each number of revolutions N the minimum required ΔV is computed for each departure date and TOF combination. One then obtains ΔV plots for each number of revolutions seperately, see Figures 11.5a to 11.5e, which show the ΔV plots corresponding to Figure 11.8. Subsequently, for each departure date and TOF combination the overall minimum required ΔV is taken, and a ΔV plot is obtained which shows the minimum required ΔV for all number of revolutions, see Figure 11.5f.

From Figure 11.5 it is clear that the optimal TOF depends on the chosen number of revolutions and that the optimal TOF increases with increasing number of revolutions.

Higher-order solutions In this paragraph the results of several higher-order solutions are discussed. For these results it is interesting to find out 1) what the characteristics of the best found trajectories are, 2) whether the optimal departure date and time of flight are the same as the ones found using the lowest-order solutions and 3) whether the higher-order results are (near-)optimal. The latter can be investigated by comparing the higher-order results with results found in literature. This is done for both the time-driven and polar-angle-driven methods at the end of this section.

In Figure 11.6 the Pareto fronts of different higher-order solutions and the best lowest-order solution are shown together with the best results obtained using spherical and pseudo-equinoctial shaping [Novak and Vasile, 2011]. This figure shows two Pareto fronts of the best lowest-order solution, obtained with different grid sizes to sample the flight window. Clearly, the results obtained with the dense grid are best regarding both ΔV and thrust acceleration, so dense grids can be



(a) $N = 0$ (b) $N = 1$ (c) $N = 2$

(d) $N = 3$ (e) $N = 4$ (f) $N = 0 - 4$

Figure 11.5: ΔV plots for the higher-order time-driven solution CPowPow2PSin05PCos05-CPowPow2PSin05PCos05-Cos25P3SinR5P3CosR5P4CosR5P4SinR5 for a mission to Mars for different number of revolutions.

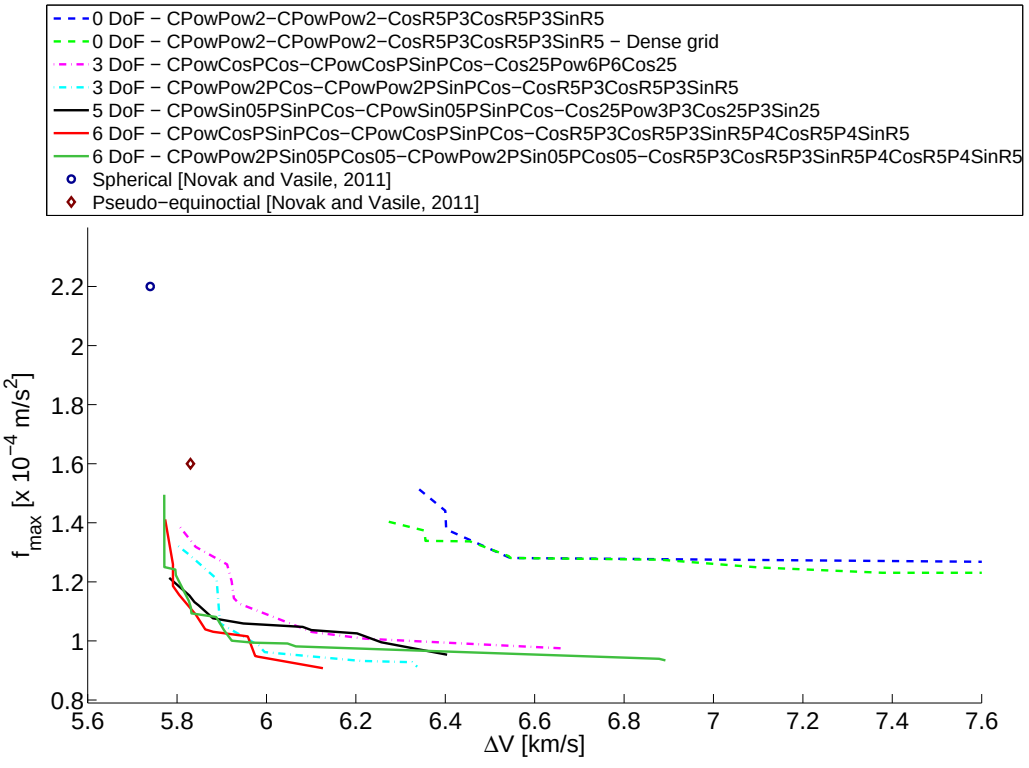


Figure 11.6: Pareto fronts of higher-order time-driven solutions for a mission to Mars.

used for fine-tuning. In addition to reducing the grid size, it is clear that adding degrees of freedom results in (much) better trajectories regarding both ΔV and f_{max}. So, the lowest-order solutions are clearly not optimal. The addition of 3 DoF improves the results already significantly with about 0.5 km/s. Applying more free parameters improves the required ΔV as well as the maximum thrust acceleration, as indicated by the Pareto fronts for 5 and 6 DoF. Even more DoF can be applied to further improve the trajectories. However, adding more DoF not only improves the results, but also complicates the optimization. Therefore the benefit of better trajectories needs to be weighed against the significant increase in computational effort.

A comparison of the higher-order results with the best pseudo-equinoctial and spherical-shaping results is done in the next section.

Table 11.2: Departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories found by the best lowest-order and different higher-order time-driven solutions for a mission to Mars.

V_r	V_θ	V_z	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPowPow2	CPowPow2	CosR5P3CosR5P3SinR5	10025	1050	6.342	1.51
CPowCos PCos	CPowCos PSinPCos	Cos25Pow6P6Cos25	9205	1200	5.807	1.39
CPowPow2 PCos	CPowPow2 PSinPCos	CosR5P3CosR5P3SinR5	9225	1180	5.803	1.32
CPowSin05 PSinPCos	CPowSin05 PSinPCos	CosR5Pow3P3CosR5 P3SinR5	9215	1200	5.783	1.21
CPowCos PSinPCos	CPowCos PSinPCos	CosR5P3CosR5P3SinR5 P4CosR5P4SinR5	9945	1160	5.773	1.41
CPowPow2 PSin05PCos05	CPowPow2 PSin05PCos05	CosR5P3CosR5P3SinR5 P4CosR5P4SinR5	9985	1100	5.771	1.50

Comparison lowest- and higher-order solutions The characteristics of the best trajectories (in ΔV) corresponding to the Pareto fronts in Figure 11.6 are shown in Table 11.2. From this table, it can be noted that the maximum thrust accelerations required for the minimum-ΔV trajectories obtained by the lowest-order solutions are about the same as the ones obtained by higher-order solutions. So the lowest-order solutions give a good estimate of the required thrust. In addition, the departure dates and flight times of the best trajectories for different velocity functions and different number of DoF are close to each other. For all results the departure date is either around 9215 or 9985 MJD2000, and the time of flight around 1150 days. These regions of low ΔV can also be seen in Figures 11.4, 11.7 and 11.8, which show the ΔV plots for one lowest-order and two higher-order solutions. Furthermore, these figures show that the regions of low ΔV indicated by lowest-order solutions are approximately the same as the regions found by higher-order solutions. In addition, it can be noticed that as the number of DoF's is increased the regions of low ΔV become larger and the ΔV values lower. So, applying (more) DoF does not only improve the results, but makes them also less dependent on the departure date and TOF.

From Table 11.2 it was found that the optimal departure date and TOF are approximately the same for different DoF, but depend on the chosen velocity functions. When Figures 11.4, 11.7 and 11.8 are compared closely, one can see that the regions of low ΔV shift a little bit when DoF's are added. So the lowest-order solutions are able to indicate the correct regions of low ΔV, however, not to indicate exactly the optimal departure date and TOF.

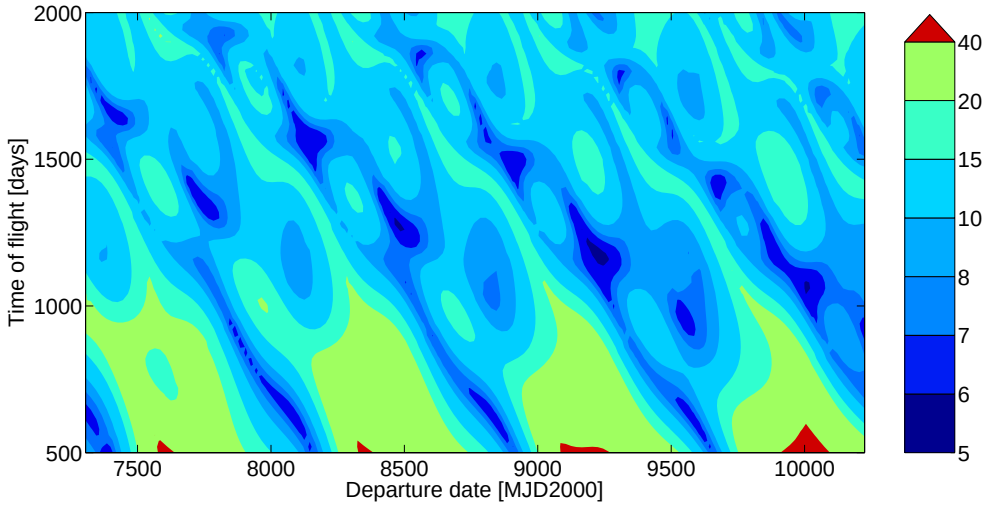


Figure 11.7: ΔV plot for the higher-order time-driven solution CPowPow2PCos-CPowPow2PSinPCos-CosR5P3CosR5P3SinR5 for a mission to Mars with $N = 0 - 5$.

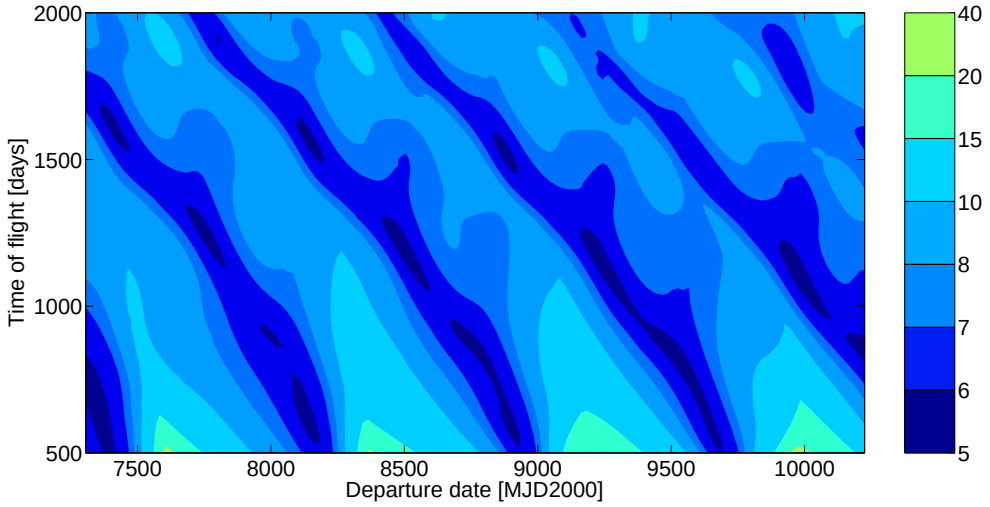
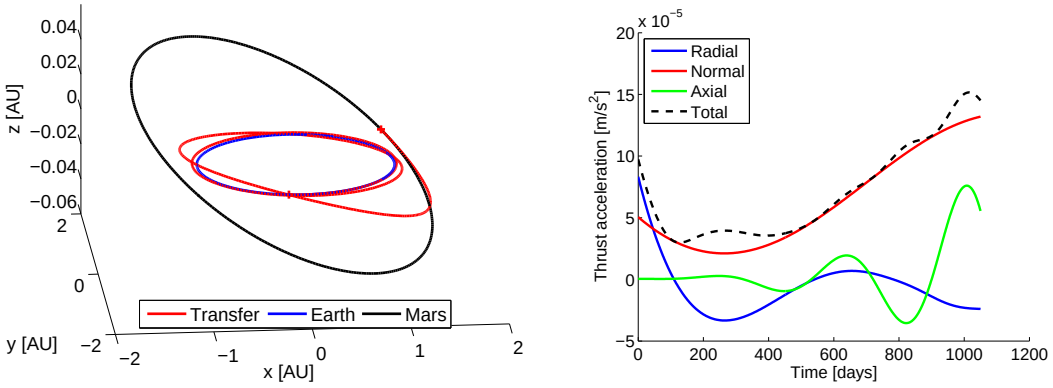


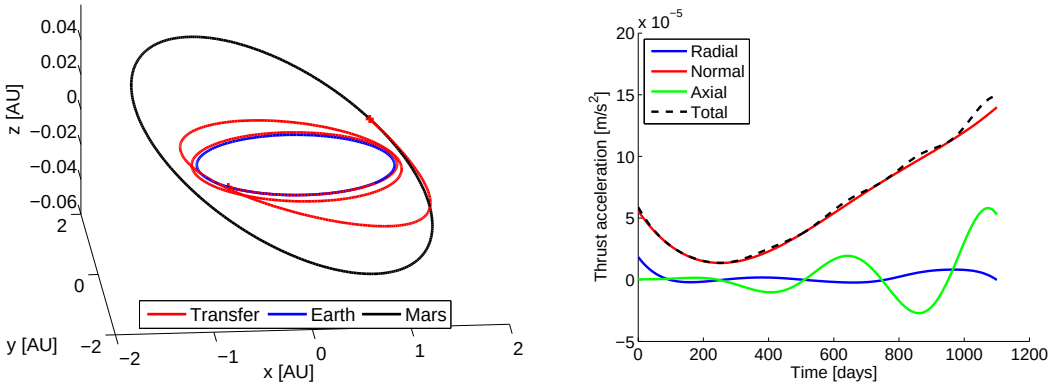
Figure 11.8: ΔV plot for the higher-order time-driven solution CPowPow2PSin05PCos05-CPowPow2PSin05PCos05-Cos25P3SinR5P3CosR5P4CosR5P4SinR5 for a mission to Mars with $N = 0 - 5$.

The trajectories and thrust profiles corresponding to the minimum-ΔV results for the lowest-order and higher-order cases are shown in Figures 11.9 and 11.10, respectively. The trajectories and thrust profiles of the lowest- and higher-order solution look quite the same. This is not very surprising since the departure dates, TOF and velocity functions of both solutions are alike.



(a) Trajectory (b) Thrust profile

Figure 11.9: Trajectory and thrust profile of best lowest-order time-driven solution for Earth-Mars: V_r = CPowPow2, V_θ = CPowPow2 and V_z = CosR5P3CosR5P3SinR5, see Table 11.1.



(a) Trajectory (b) Thrust profile

Figure 11.10: Trajectory and thrust profile of best higher-order time-driven solution for Earth-Mars: V_r = CPowPow2PSin05PCos05, V_θ = CPowPow2PSin05PCos05 and V_z = CosR5P3CosR5P3SinR5P4CosR5P4SinR5, see Table 11.2.

The main difference between both thrust profiles is the radial thrust. The radial thrust of the higher-order solution is, in contrast to the lowest-order solution, near-zero during the transfer. Since radial thrusting results in gravity losses, the near-zero radial thrust indicates that the trajectory is near optimal. Furthermore, both trajectories show an oscillating motion in the axial direction. This oscillating character also appears in the axial thrust profiles. In addition, it can be noted that the change in semi-major axis and inclination mainly take place during the second half of the transfer. For the change in inclination this is logical, since the velocity of the spacecraft decreases along the trajectory and plane changes are performed most efficiently at low velocity. However, the semi-major axis (and orbital energy) are most efficiently changed at high velocity and therefore more normal thrust would be expected during the first part of the transfer.

Finally, since the thrust profiles of the best lowest- and higher-order trajectories are very similar, the best trajectory of a lowest-order solution may already be a

sufficiently good initial guess for a subsequent optimization.

11.2.2. Shaping as function of polar angle

In addition to hodographic shaping as a function of time, also hodographic shaping as function of polar angle, see Section 5.3, has been investigated. However, for the polar-angle-driven method less velocity function combinations have been tested than for the time-driven method. As a consequence, the obtained results may not be the best results which can be obtained by the polar-angle-driven method, however, they give a good insight in the potential of the method.

Lowest-order solutions First, in this paragraph the lowest-order solutions of the polar-angle-driven method are discussed and after that the higher-order results are considered in the next paragraph.

Radial velocity and time evolution functions Figure 11.11 shows the Pareto front for some lowest-order solutions. The radial-velocity and time-evolution functions of the shown solutions consist purely of power terms. It was found that especially the application of power base functions in the radial-velocity and time-evolution functions give good results.

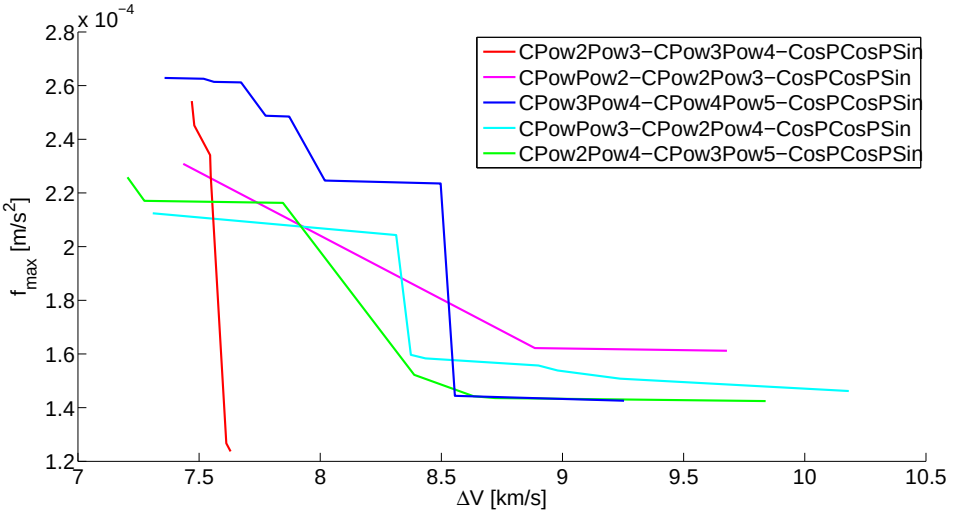


Figure 11.11: Pareto fronts of lowest-order polar-angle-driven solutions for different radial-velocity and time-evolution functions for a mission to Mars.

From Figure 11.11 and Table 11.3 it is clear that the combination CPow2Pow4-CPow3Pow5 provides the best trajectory regarding ΔV, requiring a ΔV of 7.21 km/s. This ΔV is much higher than the best ΔV obtained by the lowest-orders solutions of time-driven method (see Figure 11.1 and Table 11.1). Moreover, the required thrust accelerations are also higher than the best time-driven results, but of comparable magnitude. The thrust acceleration corresponding to the best-ΔV result of the polar-angle-driven method is 2.3×10^{-4} m/s^2, which is 1.5 times the

Table 11.3: Departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories found by lowest-order polar-angle-driven solutions with different radial-velocity and time-evolution functions for a mission to Mars.

R	T	Z	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPow2Pow3	CPow3Pow4	CosPCosPSin	7385.0	600	7.470	2.54
CPowPow2	CPow2Pow3	CosPCosPSin	10035.0	990	7.435	2.31
CPow3Pow4	CPow4Pow5	CosPCosPSin	7375.0	500	7.358	2.63
CPowPow3	CPow2Pow4	CosPCosPSin	10035.0	1000	7.309	2.12
CPow2Pow4	CPow3Pow5	CosPCosPSin	7395.0	530	7.205	2.26

thrust acceleration required for the best-ΔV trajectory found by the time-driven method.

Furthermore, it is noticeable that for the best radial-velocity and time-evolution function combinations the power terms in the time-evolution function have an order which is one higher than the order of the power terms in the radial-velocity function.

Finally, the found optimal departure dates and TOF differ significantly for different velocity functions and differ from the results of the time-driven method.

Axial velocity function Like in the time-driven method, the rate at which the amplitude of the axial motion is changed needs to be tuned such that efficient trajectories can be obtained. In Figure 11.12 the Pareto front for different axial power orders, i.e. different rates of change, are shown. From this figure it is clear that with respect to ΔV an order of 2 is best. It should, however, be noted that the differences in ΔV and thrust acceleration for the lower power orders are rather small.

Number of revolutions The Pareto fronts for different number of revolutions, see Figure 11.13, indicate that $N = 0$ is optimal regarding ΔV, whereas $N = 2$ is best with respect to the required thrust acceleration. This differs from the time-driven method where both regarding ΔV and thrust acceleration $N = 2$ is optimal. Furthermore, compared to the time-driven method the Pareto fronts for $N = 0$ and $N = 1$ are better, however, for $N \geq 2$ the fronts are worse. So, this means that for multiple revolutions the polar-angle-driven method performs worse than the time-driven method. This is opposite to what was expected, since shaping as function of the polar angle was expected to be more correlated to periodic phenomena occurring when revolving.

Higher-order solutions The Pareto fronts of the best lowest-order solution and one higher-order solution are shown in Figure 11.14. Clearly, the higher-order solution performs much better regarding both required ΔV and thrust acceleration.

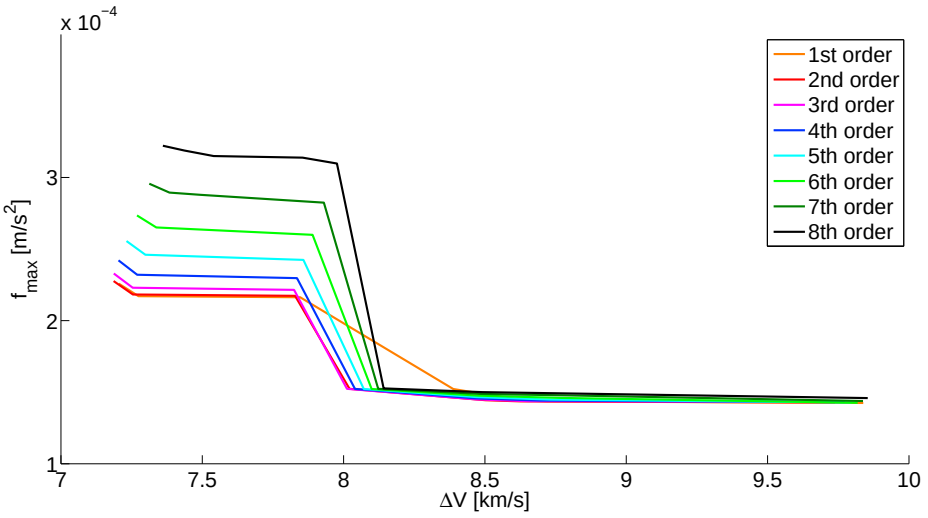


Figure 11.12: Pareto fronts of the lowest-order polar-angle-driven solution CPow2Pow4-CPow3Pow5-CosPXCosPXSin for different axial power orders X for a mission to Mars.

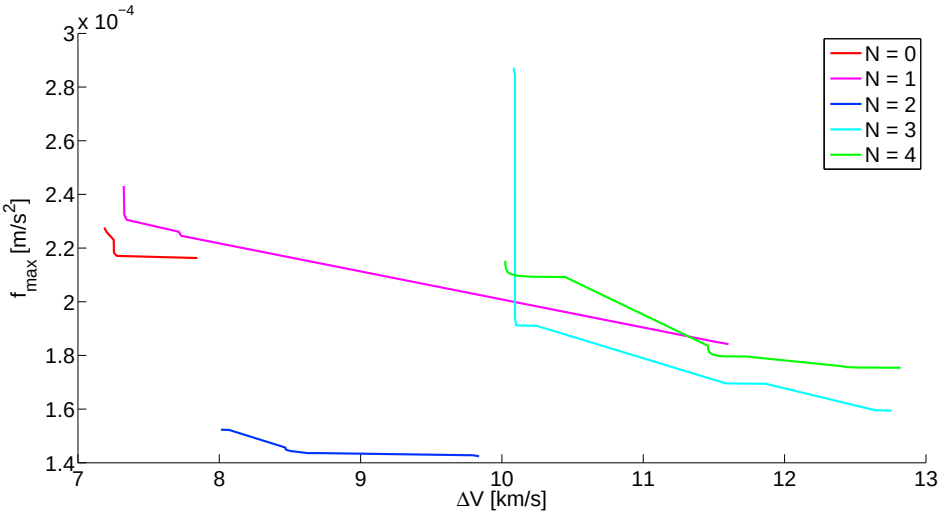


Figure 11.13: Pareto fronts of the lowest-order polar-angle-driven solution CPow2Pow4-CPow3Pow5-CosPCosPSin for different number of revolutions N for a mission to Mars.

Table 11.4: Departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories found by the best lowest-order and a higher-order polar-angle-driven solution for a mission to Mars.

R	T	Z	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPow2Pow4	CPow3Pow5	CosPCosPSin	7395	530	7.205	2.26
CPowPow2 Pow3Pow4	CPowPow2 Pow3Pow4	CosPCosPSin P2CosP2Sin	10225	840	5.814	1.64

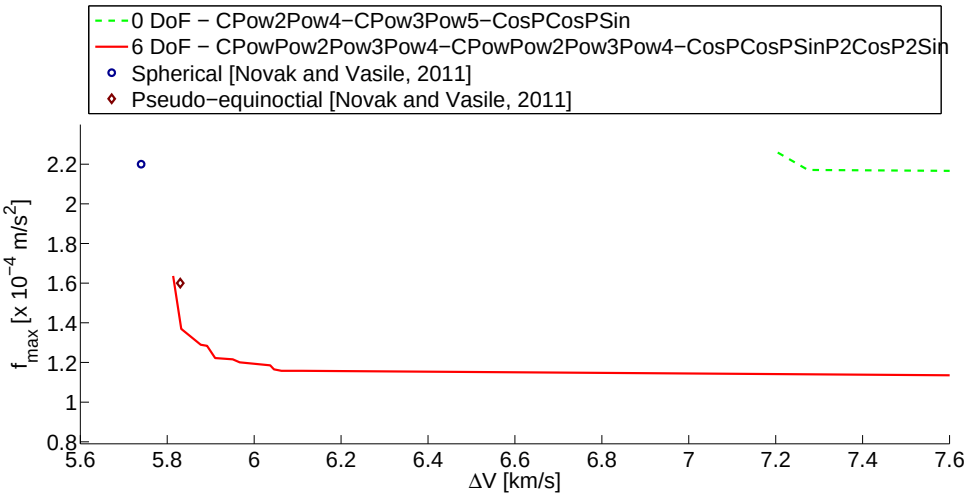


Figure 11.14: Pareto fronts of higher-order polar-angle-driven solutions for a mission to Mars.

The characteristics of the best found trajectories for the lowest- and higher-order case are shown in Table 11.4. The optimal departure date and TOF found by the lowest- and higher-order solutions differ considerably. Moreover, the obtained optimal departure date and TOF also differ from the ones found by the time-driven solutions. Especially, the optimal TOFs found by the polar-angle-driven method are shorter than those obtained by the time-driven method. This corresponds to the lower value of the optimal number of revolutions.

On the other hand, the ΔV plots shown in Figures 11.15 and 11.16 show the same pattern as the ΔV plots for the time-driven method (e.g. compare Figure 11.16 with Figure 11.8). So the same areas of low ΔV are found by both methods, which suggests that these regions truly are the optimal flight windows.

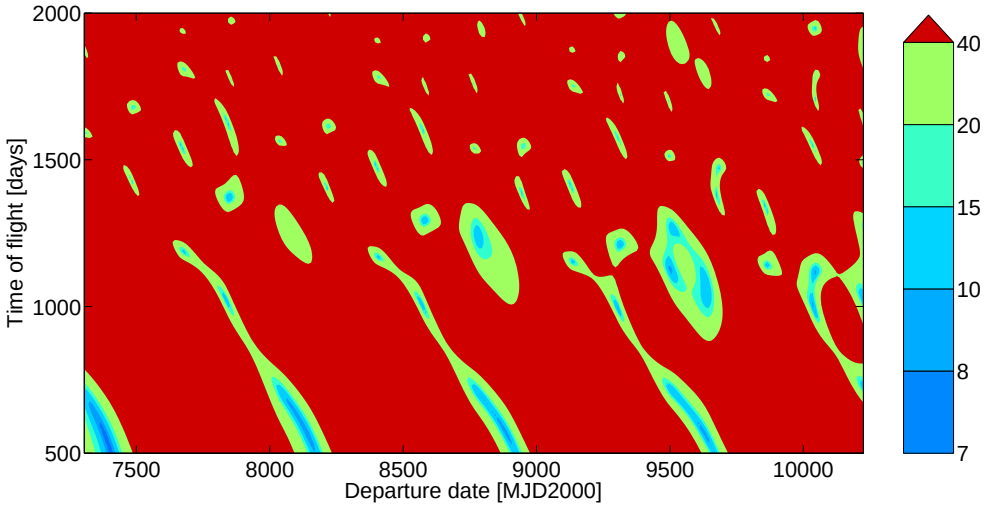


Figure 11.15: ΔV plot for the lowest-order polar-angle-driven solution CPow2Pow4-CPow3Pow5-CosPCosPSin for a mission to Mars with $N = 0 - 4$.

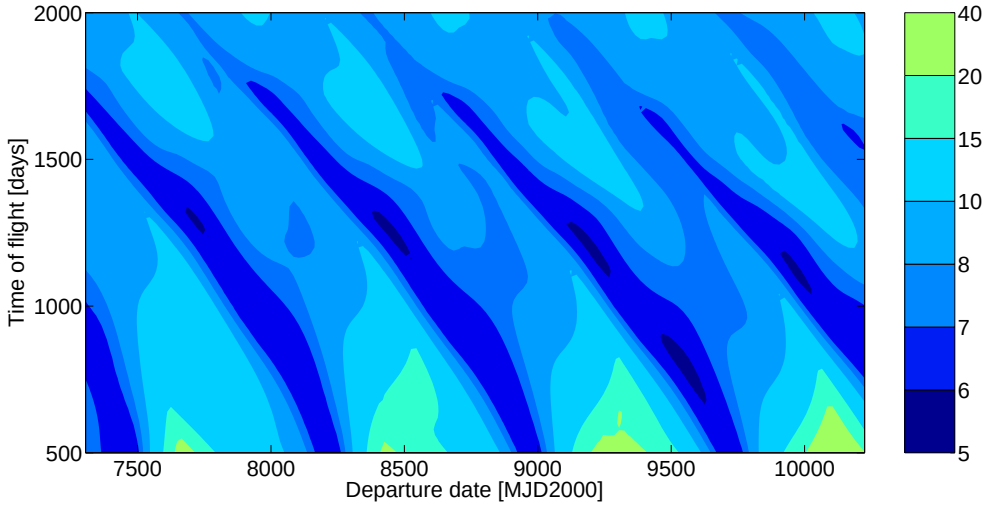


Figure 11.16: ΔV plot for the higher-order polar-angle-driven solution CPowPow2Pow3Pow4-CPowPow2Pow3Pow4-CosPCosPSinP2CosP2Sin for a mission to Mars with $N = 0 - 4$.

Comparison with other methods The best trajectories found by hodographic shaping as a function of time using 3 DoF are better than the best pseudo-equinoctial shaping solution regarding both required ΔV and thrust acceleration, see Figure 11.6. The spherical shaping method, on the other hand, is better with respect to ΔV, however, worse regarding the maximum required thrust compared with 6-DoF results. The best hodographic shaping solution requires an f_{max} of $1.5{\times}10^{-4}$ m/s^2 whereas the best spherical shaping solution requires a peak thrust acceleration of $2.2{\times}10^{-4}$ m/s^2 [Novak and Vasile, 2011]. The polar-angle-driven method, on the other hand, performs only better than pseudo-equinoctial shaping when 6 DoF are applied, see Figure 11.14.

An overview of the best results obtained by different methods is shown in Table 11.5. As stated before, a 2D Hohmann transfer to Mars requires a ΔV of 5.50 km/s and the best low-thrust trajectory to Mars found by DITAN [Vasile et al., 2007] requires a ΔV of 5.66 km/s. These ΔV values are lower than 5.77 km/s found by the hodographic shaping method. So, the hodographic-shaping method did not obtain the truly optimal trajectory.

On the other hand, the thrust acceleration of $1.5{\times}10^{-4}$ m/s^2 required by the best hodographic-shaping result is lower the ones obtained by other methods. Moreover, this thrust acceleration is equivalent to a thrust of 150 mN in case of a spacecraft of 1000 kg. This amount of thrust can be achieved using current technology and therefore the obtained solution is not only near-optimal, but also feasible.

11.3. 1989ML

The near-Earth asteroid 1989ML has a semi-major axis which is smaller than Mars', however its eccentricity and inclination are a bit higher than those of Mars. Since 1989ML is closer to Earth than Mars, the ΔV required for a transfer is lower.

Table 11.5: Required ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectory found by different methods.

Method	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
Hohmann	5.50	-
Hodographic - time	5.77	1.5
Hodographic - polar angle	5.81	1.6
Spherical [Novak and Vasile, 2011]	5.74	2.2
Pseudo-equinoctial [Novak and Vasile, 2011]	5.83	1.6
DITAN [Vasile et al., 2007]	5.66	1.5

For the 2D case, a Hohmann transfer from Earth to 1989ML's apocenter requires a ΔV of 3.22 km/s and a TOF of 247 days. The best 3D low-thrust trajectory to 1989ML found by DITAN requires a ΔV of 4.21 km/s [Novak and Vasile, 2011].

In the following the hodographic shaping results for a mission to asteroid 1989ML are discussed. These results are discussed in the same manner as for the Mars test case.

11.3.1. Shaping as function of time

Lowest-order solutions First the lowest-order solutions of the time-driven method are considered.

Radial and normal velocity function The same radial and normal velocity function combinations as for the Mars case have been tested. The Pareto fronts of these five lowest-order solutions are shown in Figure 11.17. The lowest-ΔV trajectory is obtained using the radial-normal combination CPowPow2-CPowPow2, and requires a ΔV of 5.0 km/s and a thrust acceleration of 3.1×10^{-4} m/s^2. The combination CPowPow2-CPowPow2 was also found to be the best for a mission to Mars. Furthermore, as in the Mars case, the velocity-function combinations which include cosine functions perform worst.

Table 11.6: Departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories found by lowest-order time-driven solutions with different radial- and normal velocity functions for a mission to 1989ML.

V_r	V_θ	V_z	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPowCos	CPowCos	CosR5PCosR5PSinR5	8105	230	5.429	3.86
CPowPow2	CPowCos	CosR5PCosR5PSinR5	8105	230	5.362	4.09
CPowSin05	CPowSin05	CosR5PCosR5PSinR5	8115	240	5.037	3.06
CPowPow2	CPowSin05	CosR5PCosR5PSinR5	8115	240	5.037	3.06
CPowPow2	CPowPow2	CosR5PCosR5PSinR5	8115	240	4.995	3.07
CPowPow2	CPowPow2	CosR5P6CosR5P6SinR5	7875	410	4.842	4.84

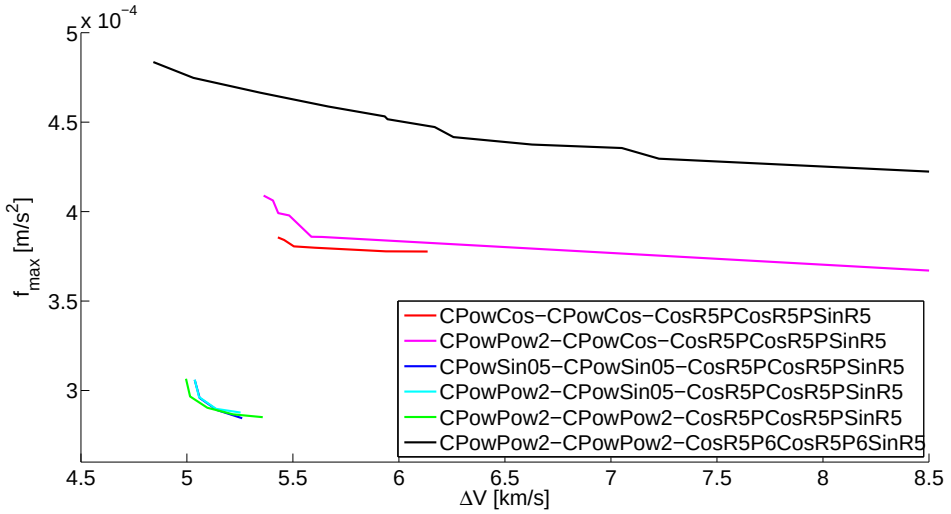


Figure 11.17: Pareto fronts of lowest-order time-driven solutions for different radial- and normal-velocity functions for a mission to 1989ML.

In Table 11.6 the characteristics of the best trajectories found by the different velocity-function combinations are shown. It is clear that the different velocity-function combinations find approximately the same optimal departure date and time of flight. This indicates that the results are very dependent on the departure and arrival state.

In Table 11.6 also the best result obtained with a higher axial power order is shown. Clearly, this result is better with respect to ΔV. The required thrust acceleration, however, is much higher than required when using an axial power order of one. In addition, this result shows that the order of the axial power not only has an effect on the obtained ΔV and thrust acceleration, but also on the found optimal departure date and TOF. The obtained optimal departure date and TOF clearly differ significantly from other obtained optimal departure dates and TOFs. Nevertheless, Figure 11.21 indicates that the different optimal departure dates and flight times belong to the same larger region of low ΔV.

Axial velocity function The Pareto fronts for different axial orders are shown in Figure 11.18. This figure confirms that the application of 6^{th}-order axial powers results in the best trajectory regarding ΔV. However, the figure also shows that the application of 1^{st}-order powers results in much smaller thrust accelerations. So, as in the Mars case, it is clear that the required ΔV and thrust acceleration depend on the chosen axial power order.

Number of revolutions In Figure 11.19 the Pareto fronts for different number of revolutions are shown. This figure indicates that the good results for 6^{th}-order axial powers correspond to $N = 1$ and the good results for 1^{st}-order axial powers to $N = 0$. So, the order of the axial powers which is best regarding ΔV depends on the number of revolutions. In this case, axial power orders of 1, 6 and 7 are

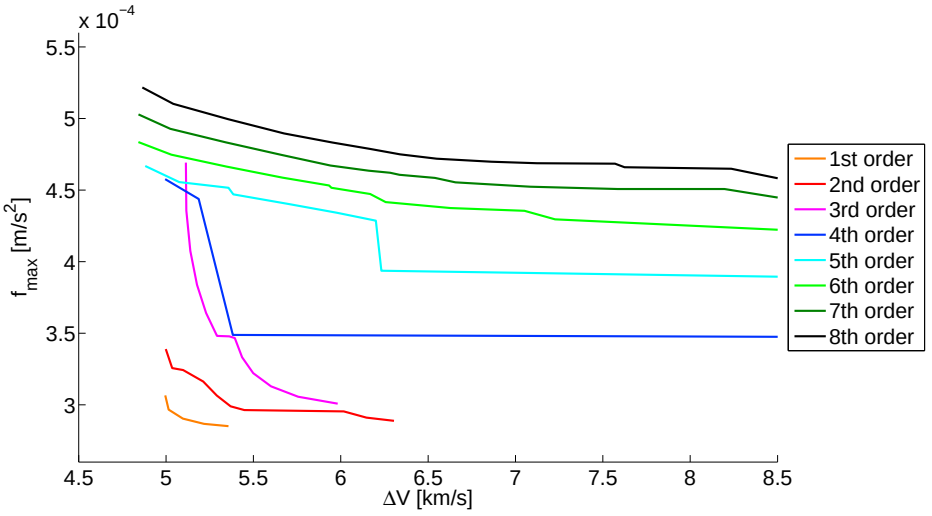


Figure 11.18: Pareto fronts of the lowest-order time-driven solution CPowPow2-CPowPow2-CosR5PXCosR5PXSinR5 for different axial power orders X for a mission to 1989ML.

best regarding ΔV for $N = 0$, $N = 1$ and $N = 2$, respectively.

The fact that (again) the best axial power order depends on the number of revolutions explains why the optimal departure date and TOF for 6^{th}-order axial powers differ from the ones for 1^{st}-order axial powers. The 6^{th}-order axial powers are namely best when $N = 1$ instead of when $N = 0$ and therefore different departure dates and TOFs are optimal, as was discussed in Section 11.2.

Furthermore, it can be noticed from Figure 11.19 that $N = 0$ is best regarding required thrust acceleration. This is surprising since more revolutions correspond to longer flight times and therefore presumably lower thrust accelerations.

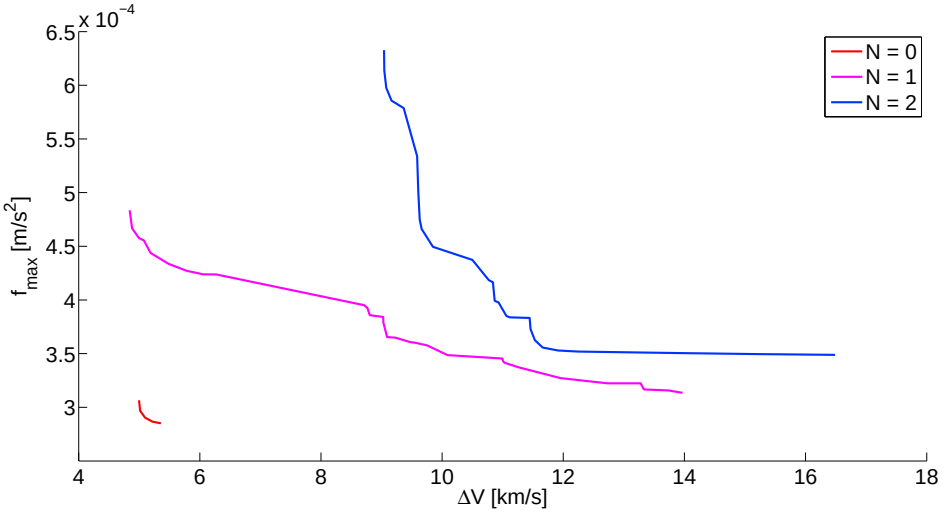


Figure 11.19: Pareto fronts of the lowest-order time-driven solution CPowPow2-CPowPow2-CosR5PCosR5PSinR5 for different number of revolutions N for a mission to 1989ML.

Higher-order solutions In Figure 11.20 the Pareto fronts for higher-order solutions are shown. As expected, the results improve when the number of DoF is increased. In addition, Table 11.7 shows the characteristics of the best trajectories corresponding to the higher-order results. It can be noticed that contrary to the lowest-order results the optimal departure date and TOF of the higher-order results are not dependent on the order of the axial powers. So, apparently the addition of DoF reduces the effect of the order of axial powers on the departure date and time of flight.

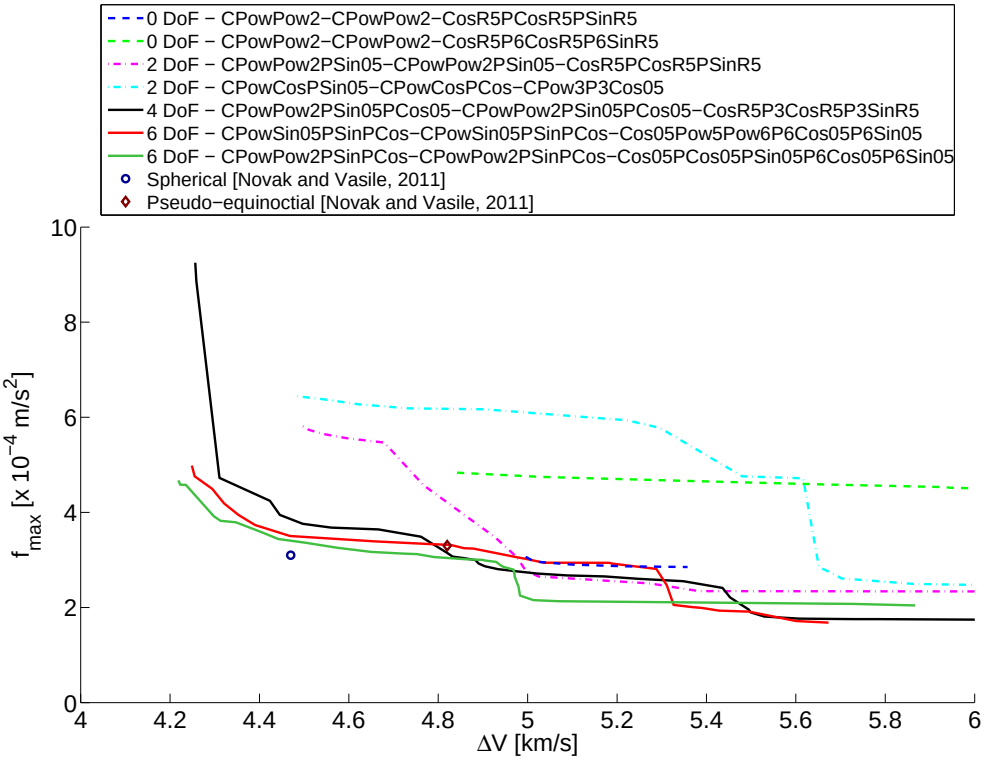


Figure 11.20: Pareto fronts of higher-order time-driven solutions for a mission to 1989ML.

In the Mars case, it was found that the required thrust accelerations for the best lowest- and higher-order solutions are approximately the same. For a mission to asteroid 1989ML this is also the case, see Table 11.7.

Furthermore, the optimal departure date and TOF corresponding to the best higher-order results are approximately the same as the ones obtained by the lowest-order soluions with an axial power order of one. So, as in the Mars case, the lowest-order solutions are able to locate the correct region of low ΔV. Figures 11.21 to 11.23 confirm this and show that areas of low ΔV increase and shift a little bit when the number of DoF is increased. Furthermore, in these ΔV plots a pattern can be observed which is approximately 1200-days periodic. This corresponds to the synodic period of Earth and 1989ML with respect to the Sun, which is 1204 days.

Table 11.7: Departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories found by different higher-order time-driven solutions for a mission to 1989ML.

V_r	V_θ	V_z	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPowPow2	CPowPow2	CosR5PCosR5PSinR5	8115	240	4.995	3.07
CPowPow2	CPowPow2	CosR5P6CosR5P6SinR5	7875	410	4.842	4.84
CPowPow2 PSin05	CPowPow2 PSin05	CosR5PCosR5PSinR5	8025	260	4.498	5.81
CPowCos PSin05	CPowCos PCos	CPow3P3Cos05	8045	220	4.485	6.44
CPowPow2 PSin05PCos05	CPowPow2 PSin05PCos05	CosR5P3CosR5P3SinR5	8045	220	4.257	9.25
CPowSin05 PSinPCos	CPowSin05 PSinPCos	Cos05Pow5Pow6 P6Cos05P6Sin05	8045	240	4.249	4.98
CPowPow2 PSinPCos	CPowPow2 PSinPCos	Cos05PCos05PSin05 P6Cos05P6Sin05	8025	260	4.219	4.67

It can be noticed in Figures 11.21 to 11.23 that some contour lines seem to be discontinuous, e.g. at a date of 8000 MJD2000 and TOF of 400 days. These irregularities are caused by the fact that for the trigonometric terms in the axial-velocity function a frequency is applied which depends on the number of revolutions. The number of revolutions which is optimal regarding ΔV depends, however, strongly on the departure date and TOF (see Figure 11.5). So, when looping over the departure date and TOF, the optimal number of revolutions can suddenly change. Consequently, the frequency of the trigonometric terms changes which causes a discontinuity in the ΔV results. In case the applied frequency would be the same, independent of the number of revolutions, the irregularities would not be present in the ΔV plot. In Figures 11.7 and 11.8 these irregularities can also be seen, but are less clear.

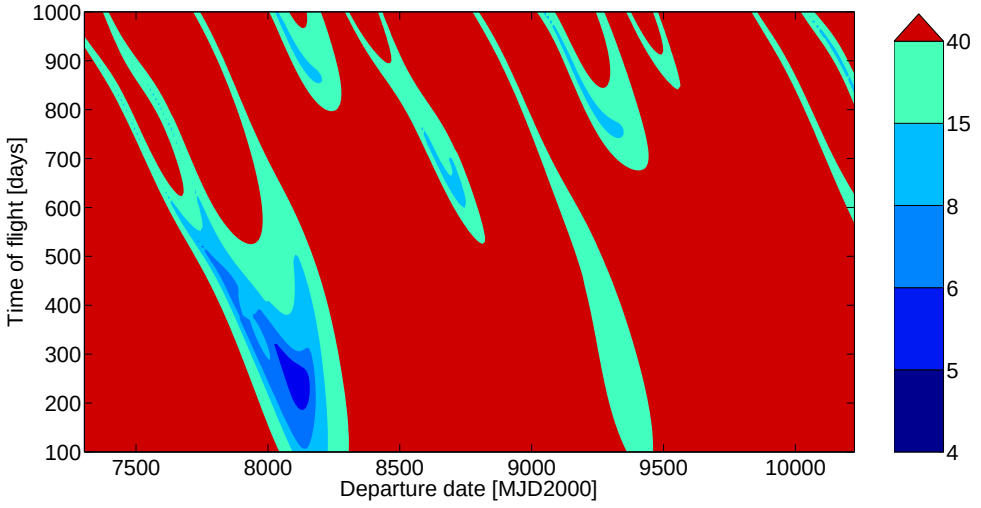


Figure 11.21: ΔV plot for the lowest-order time-driven solution CPowPow2-CPowPow2-CosR5PCosR5PSinR5 for a mission to asteroid 1989ML with $N = 0 - 2$.

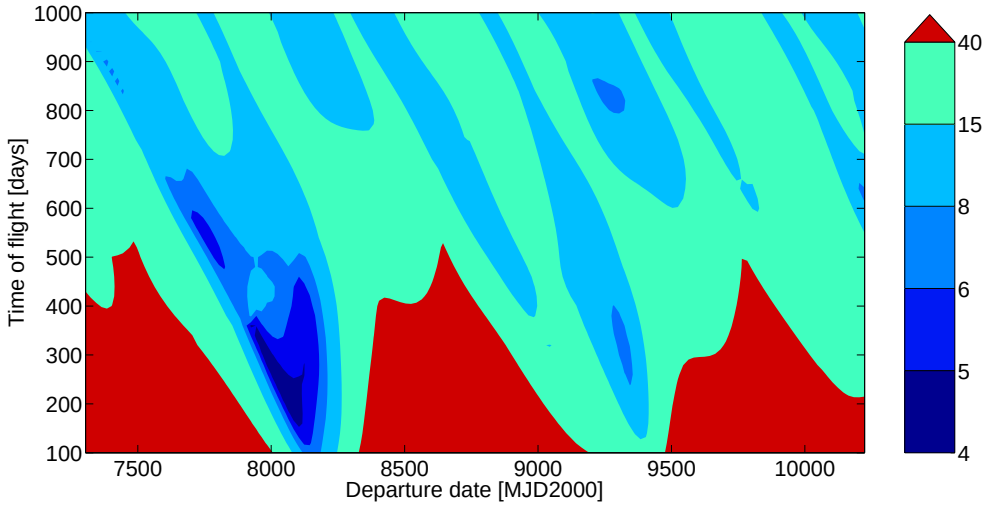


Figure 11.22: ΔV plot for the higher-order time-driven solution CPowPow2PSin05-CPowPow2PSin05-CosR5PCosR5PSinR5 for a mission to asteroid 1989ML with $N = 0 - 2$.

Figures 11.24 and 11.25 show the trajectories and thrust profiles corresponding to the best solutions found for the lowest-order and higher-order case. Remarkably, for the best found trajectories the rendezvous with asteroid 1989ML takes place near its pericenter. From the theory of non-perturbed transfers, see Section 2.6, one would expect that the optimal rendezvous location is around apocenter. However, the ΔV-plots for the spherical and the pseudo-equinoctial shaping method (see e.g. Figure 8.1) [Novak and Vasile, 2011] indicate the same regions of low ΔV. So the optimal trajectories found by the spherical and the pseudo-equinoctial shaping method (most probably) also rendezvous with 1989ML at its pericenter. In addition, the ΔV obtained by the time-driven method are about equal to the ΔV obtained by DITAN, see Table 11.10. So, either the low-thrust (shaping) methods are not able to find the optimal departure date and TOF, or a rendezvous with 1989ML near its pericenter is actually optimal. This should be investigated in the future.

11.3.2. Shaping as function of polar angle

The results of hodographic shaping as a function of the polar angle are discussed here.

Lowest-order solutions The Pareto fronts and an overview of the lowest-order results are shown in Figure 11.26 and Table 11.8, respectively. The best trajectories are obtained with CPow2Pow4-CPow3Pow5 and CPow2Pow3-CPow3Pow4 which both require a ΔV of 4.95 km/s. The combination CPow2Pow4-CPow3Pow5 requires the lowest thrust acceleration of the two, namely 3.2×10^{-4} m/s^2.

These results are as good as the time-driven method's lowest-order results. So for the lowest-order case the polar-angle-driven method performs well for the mission

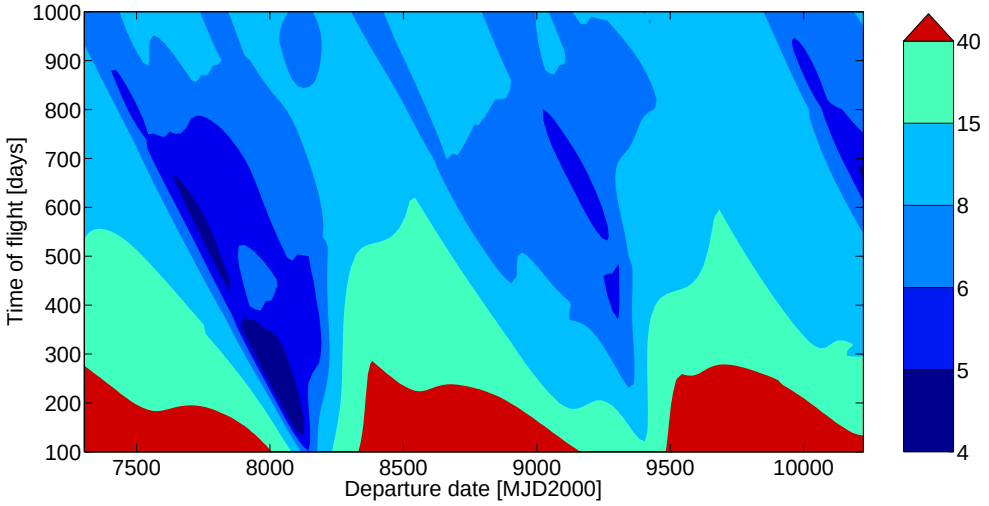
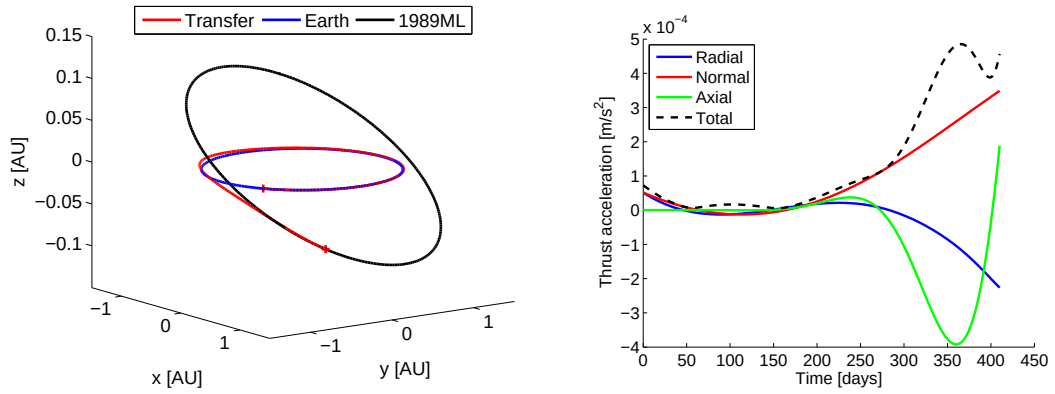
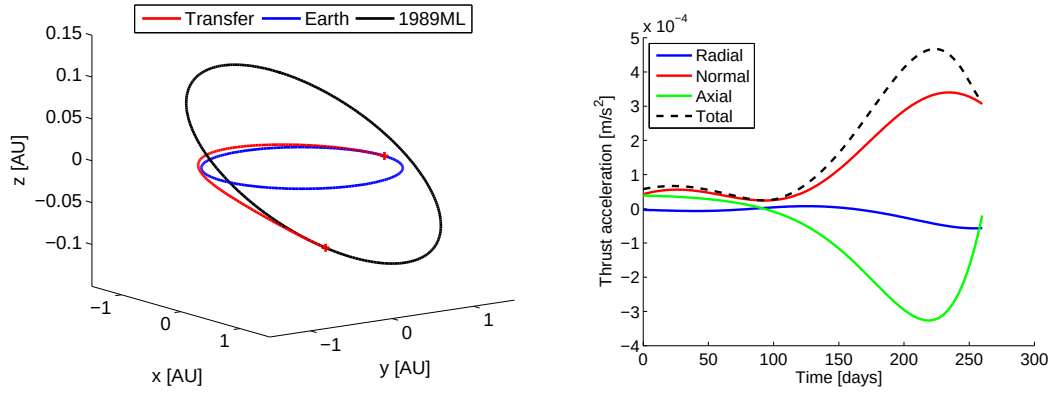


Figure 11.23: ΔV plot for the higher-order time-driven solution CPowPow2PSin05PCos05-CPowPow2PSin05PCos05-CosR5P3CosR5P3SinR5 for a mission to asteroid 1989ML with $N = 0-2$.



(a) Trajectory (b) Thrust profile

Figure 11.24: Trajectory and thrust profile of best lowest-order time-driven solution for Earth-1989ML: V_r = CPowPow2, V_θ = CPowPow2 and V_z = CosR5P6CosR5P6SinR5, see Table 11.6.



(a) Trajectory (b) Thrust profile

Figure 11.25: Trajectory and thrust profile of best higher-order time-driven solution for Earth-1989ML: V_r = CPowPow2PSin05PCos05, V_θ = CPowPow2PSin05PCos05 and V_z = CosR5P3CosR5P3SinR5P4CosR5P4SinR5, see Table 11.7.

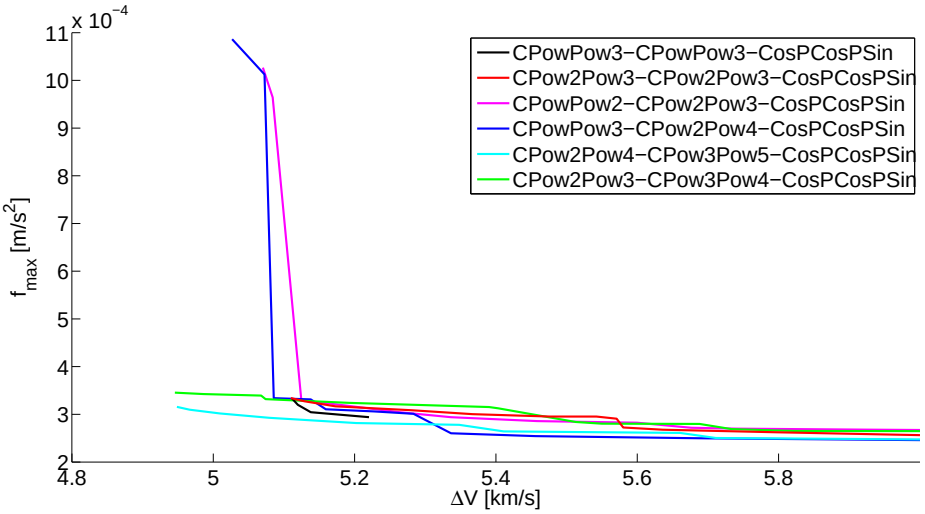


Figure 11.26: Pareto fronts of lowest-order polar-angle-driven solutions for different radial-velocity and time-evolution functions for a mission to 1989ML.

to asteroid 1989ML. Furthermore, similar to the Mars case the best trajectories are obtained with time-evolution functions which contain powers which are one order higher than the powers in the radial-velocity function.

Table 11.8: Departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories found by lowest-order polar-angle-driven solutions with different radial-velocity and time-evolution functions for a mission to 1989ML.

R	T	Z	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPowPow3	CPowPow3	CosPCosPSin	8115	250	5.111	3.33
CPow2Pow3	CPow2Pow3	CosPCosPSin	8115	290	5.110	3.34
CPowPow2	CPow2Pow3	CosPCosPSin	8075	180	5.070	10.27
CPowPow3	CPow2Pow4	CosPCosPSin	8065	190	5.027	10.86
CPow2Pow4	CPow3Pow5	CosPCosPSin	8125	340	4.949	3.16
CPow2Pow3	CPow3Pow4	CosPCosPSin	8135	300	4.946	3.45

In addition to different radial-velocity and time-evolution functions also the influence of different axial power orders and the number of revolutions has been investigated. The corresponding Pareto front plots can be found in Appendix C. It was found that with 2$^{\text{nd}}$-order axial powers and $N = 0$ the best results are obtained regarding both ΔV and thrust acceleration.

Furthermore, it was found that in comparison with the time-driven method the polar-angle-driven method performs better for $N = 0$ and $N = 2$ where as for $N = 1$ it performs worse.

Higher-order solutions One combination of higher-order velocity functions has been tested for a mission to asteroid 1989ML. The corresponding Pareto front and

characteristics are shown in Figure 11.27 and Table 11.9, respectively.

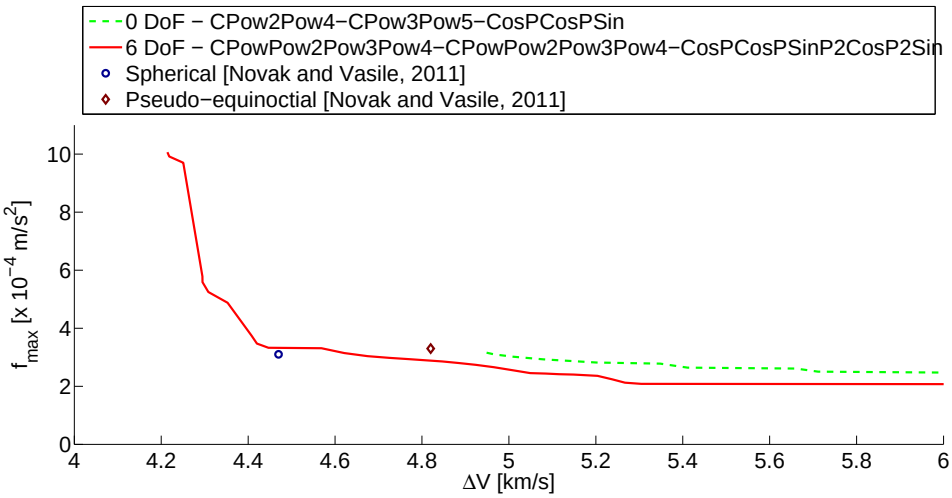


Figure 11.27: Pareto fronts of higher-order polar-angle-driven solutions for a mission to 1989ML.

Table 11.9: Departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories (and one lower-f_{max} trajectory) found by the best lowest-order and a higher-order polar-angle-driven solution for a mission to 1989ML.

R	T	Z	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPow2Pow3	CPow3Pow4	CosPCosPSin	8135	300	4.946	3.45
CPowPow2 Pow3Pow4	CPowPow2 Pow3Pow4	CosPCosPSin P2CosP2Sin	8025	260	4.295	5.79
			8045	220	4.214	10.07

The Pareto front of the higher-order solution of the polar-angle-driven method is almost the same as the best one of the time-driven method. On the one hand, the trajectories found by the polar-angle-driven method require a bit higher thrust acceleration. On the other hand, the best polar-angle-driven trajectory requires a ΔV of 4.21 km/s which is less than the 4.22 km/s required for the best time-driven trajectory. So, in conclusion, both the lowest-order and higher-order solutions of the polar-angle-driven method perform well for a mission to asteroid 1989ML.

Comparison with other shaping methods In Table 11.10 the minimum ΔV and corresponding required thrust acceleration found by different methods are shown. From the table it is clear that both hodographic shaping methods perform much better than the spherical and pseudo-equinoctial shaping methods regarding ΔV. In addition, the Pareto fronts in Figures 11.20 and 11.27 show that the hodographic, spherical and pseudo-equinoctial shaping methods perform equally well regarding required thrust acceleration.

From Figure 11.20 it is furthermore clear that using only 2 DoF, the time-driven method can find a trajectory which requires a lower ΔV than the best pseudo-equinoctial shaping one. The best spherical-shaping result is beaten regarding

ΔV using 4 DoF and is equaled with respect to both ΔV and thrust acceleration using 6 DoF. The best ΔV found by the time-driven method equals 4.22 km/s, which is 0.25 km/s lower than the best ΔV obtained by spherical shaping. The corresponding required thrust acceleration is, however, 1.5 times higher for the hodographic-shaping result than for the spherical-shaping result. The best polar-angle-driven trajectory, on the other hand, requires a slightly smaller ΔV, however, a much higher thrust acceleration.

Based on the best spherical-shaping result, the trajectory optimizer DITAN found a trajectory which requires a ΔV of 4.21 km/s and a thrust acceleration of 3.1×10^{-4} m/s^2. The hodographic-shaping methods equal this performance regarding ΔV, however, perform worse regarding the required thrust. So, regarding ΔV the hodographic shaping methods perform very well.

The reason why the ΔV required for a 2D Hohmann transfer to 1989ML is much lower than the ΔV obtained by the hodographic-shaping method and DITAN should be investigated in the future. Most probably this difference is caused by the required orbital-plane change. For an inclination change of 4.38° at 1989ML's apocenter a ΔV of 1.65 km/s is required. Another reason for this difference in ΔV might be that only a limited range of departure dates has been applied and that the truly optimal departure date is not within this range.

Table 11.10: Required ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectory found by different methods.

Method	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
Hohmann	3.22	-
Hodographic - time	4.22	4.7
Hodographic - polar angle	4.21	10.1
Spherical [Novak and Vasile, 2011]	4.47	3.1
Pseudo-equinoctial [Novak and Vasile, 2011]	4.82	3.3
DITAN [Novak and Vasile, 2011]	4.21	3.1

11.4. Tempel 1

A mission to comet Tempel 1 is a challenging one because the orbit of Tempel 1 is highly eccentric and has relatively high inclination. The ΔV required for a 2D Hohmann transfer to Tempel 1 is 9.92 km/s. Furthermore, the best ΔV for a 3D low-thrust trajectory found by DITAN is 10.69 km/s [Novak and Vasile, 2011].

In the following both the results of the time-driven and polar-angle-driven method are discussed.

11.4.1. Shaping as function of time

Lowest-order solutions In Figure 11.28 and Table 11.11 the results for different lowest-order solutions are shown. Since velocity functions including a cosine base function appeared to perform very badly, the results of two other velocity function combinations are shown.

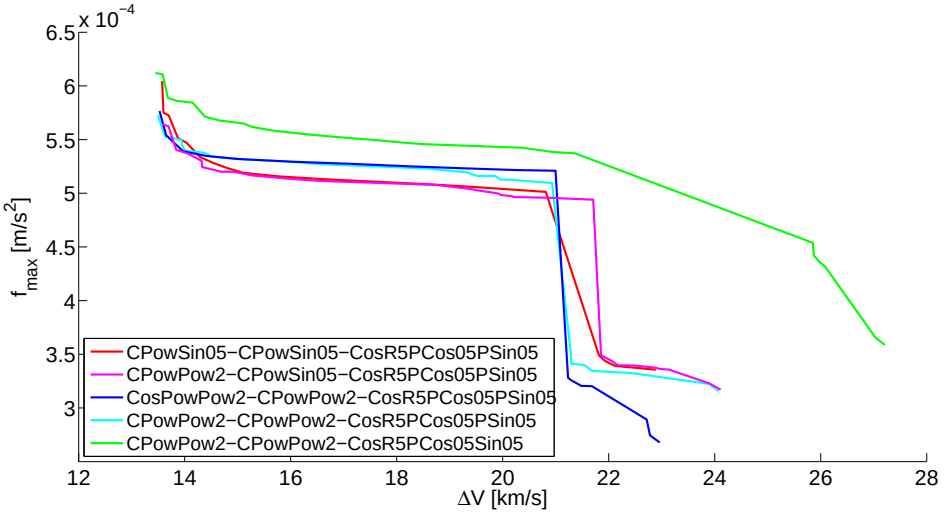


Figure 11.28: Pareto fronts of lowest-order time-driven solutions for different radial- and normal-velocity functions for a mission to Tempel 1.

Table 11.11: Departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories found by lowest-order time-driven solutions with different radial- and normal-velocity functions for a mission to Tempel 1.

V_r	V_θ	V_z	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPowSin05	CPowSin05	CosR5PCos05PSin05	5649	510	13.575	6.05
CPowPow2	CPowSin05	CosR5PCos05PSin05	5649	520	13.571	5.64
CosPowPow2	CPowPow2	CosR5PCos05PSin05	5649	520	13.527	5.77
CPowPow2	CPowPow2	CosR5PCos05PSin05	5649	520	13.494	5.73
CPowPow2	CPowPow2	CosR5PCos05Sin05	5639	520	13.449	6.12

As for missions to Mars and asteroid 1989ML, the radial and normal velocity-function combination CPowPow2-CPowPow2 performs best. In addition, for the Tempel 1 case, the application of a sine function instead of a power times sine function in the axial velocity function appears to provide good results.

From Table 11.11, it can be noticed that the found optimal departure dates and TOFs for different velocity functions are very close to each other. So, apparently the required ΔV is very dependent on the departure date and TOF, i.e. on the geometry of the problem.

The Pareto fronts for different axial power orders and different number of revolutions are shown in Appendix C. It was found that 1$^{\text{st}}$-order axial powers are best for $N = 0$ and 6$^{\text{th}}$-order power for $N = 1$.

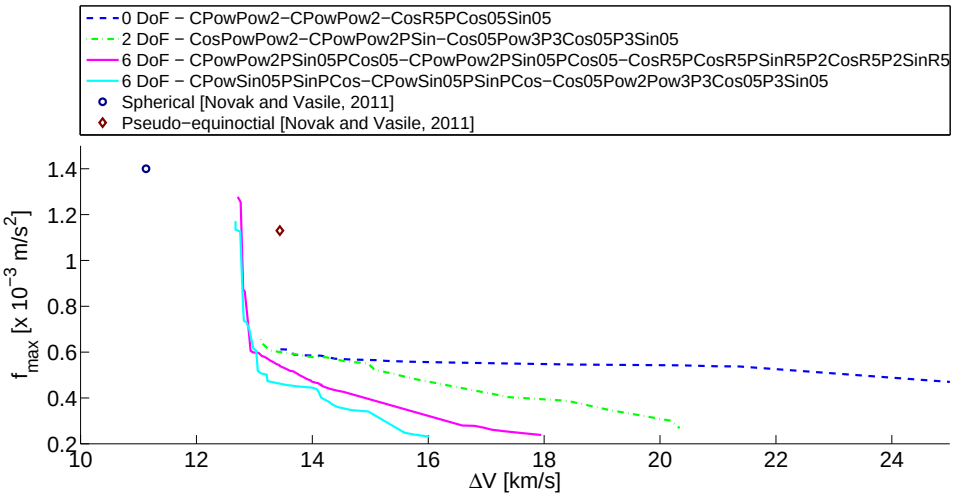


Figure 11.29: Pareto fronts of higher-order time-driven solutions for a mission to Tempel 1.

Table 11.12: Departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories (and one lower-f_{max} trajectory) found by the best lowest-order and different higher-order time-driven solutions for a mission to Tempel 1.

V_r	V_θ	V_z	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPowPow2	CPowPow2	CosR5PCos05Sin05	5639	520	13.449	6.12
CosPowPow2 PSin	CPowPow2	Cos05Pow3P3Cos05 P3Sin05	5659	500	13.107	6.55
CPowPow2 PSin05PCos05	CPowPow2 PSin05PCos05	CosR5PCosR5PSinR5 P2CosR5P2SinR5	5719	400	12.712	12.78
CPowSin05 PSinPCos	CPowSin05 PSinPCos	Cos05Pow2Pow3 P3Cos05P3Sin05	5719	420	12.820	7.36
			5699	420	12.676	11.72

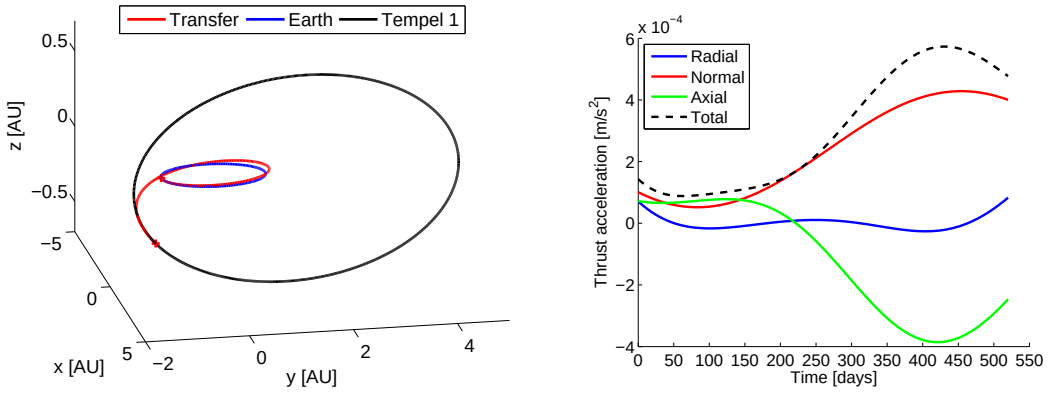
Higher-order solutions The higher-order results are shown in Figure 11.29 and Table 11.12. Similar trends as for a mission to Mars and asteroid 1989ML can be observed; the results improve when the number of DoF is increased and the lowest-ΔV trajectories require increased thrust accelerations. Figure 11.29 clearly shows that for the two 6-DoF solutions the required thrust acceleration almost triples as the ΔV comes below 13 km/s. So on the hand, low-ΔV trajectories are obtained at the cost of a high required thrust acceleration. On the other hand, the required thrust acceleration can easily be reduced by allowing for a higher ΔV. In Table 11.12 two results are shown for the same velocity function combination CPowSin05PSinPCos - CPowSin05PSinPCos - Cos05Pow2Pow3P3Cos05P3Sin05. These two results show that at the cost of only 0.15 km/s the required thrust can be reduced with more than 35%.

With respect to the lowest-order solutions the ΔV obtained by using 6 DoF has improved with 5.7%. For missions to Mars and asteroid 1989ML these improvements were better, namely 9% and 12.9%, respectively. So, either the lowest-order solutions are already quite good or the tested higher-order solutions are not well able to find low-ΔV trajectories.

In Appendix C the ΔV plots of the best lowest- and higher-order solution are shown. These plots show that the regions of low ΔV run diagonal from large TOFs to large departure dates. This indicates that the required ΔV mainly depends on the arrival date. In addition, two pattern can observed in these plots. The largest pattern is periodic with approximately 2000 days which corresponds to the orbital period of Tempel 1 of 2017 days. A smaller pattern can be observed which has a periodicity of about 400 days. This periodicity is related to orbital period of the Earth (365 days) or the synodic period of the Earth-Tempel 1-system (446 days) or both.

Best trajectories Figures 11.30 and 11.31 show the trajectories for the best lowest- and higher-order solutions. From the trajectory plots it can be noticed that the rendezvous take place near the pericenter of Tempel 1. This was also the case for the best trajectories to asteroid 1989ML. So, this suggests that a rendezvous near pericenter is optimal for low-thrust trajectories in contrast to non-perturbed transfers (as discussed in Section 2.6).

Furthermore, the thrust profile corresponding to the best higher-order result shows that the maximum required thrust acceleration of 11.72×10^{-4} m/s^2 occurs as a peak at the end of the trajectory. This peak thrust is the result of high normal and axial required thrust. These thrust components are related to the change in semi-major axis and the inclination, see Eqs. (2.28) and (2.29). Therefore, regarding the required thrust acceleration it seems to be better to change the semi-major axis and the inclination more gradually over an extended period of time instead of quickly of a short period.

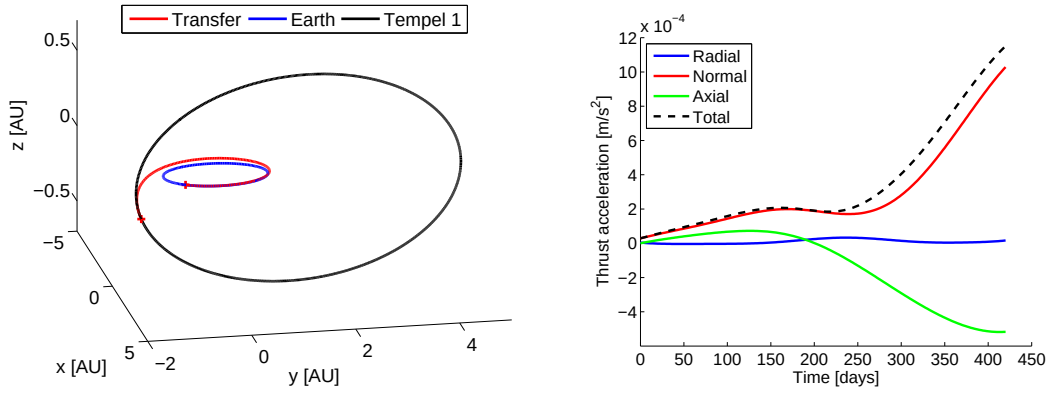


(a) Trajectory (b) Thrust profile

Figure 11.30: Trajectory and thrust profile of best lowest-order time-driven solution for Earth-Tempel 1: V_r = CPowPow2, V_θ = CPowPow2 and V_z = CosR5PCos05PSin05, see Table 11.11.

11.4.2. Shaping as function of polar angle

Lowest-order solutions The results of the lowest-order solutions are shown in Table 11.13. The corresponding Pareto fronts are shown in Figure C.13 in Ap-



(a) Trajectory (b) Thrust profile

Figure 11.31: Trajectory and thrust profile of best higher-order time-driven solution for Earth-Tempel 1: V_r = CPowSin05PSinPCos, V_θ = CPowSin05PSinPCos and V_z = Cos05Pow2Pow3P3Cos05P3Sin05, see Table 11.12.

pendix C. Both regarding ΔV and thrust acceleration, these results are worse than those found by the time-driven method. Similar to the time-driven results, the obtained optimal departure dates and TOFs are close to each other. However, the found optimal departure dates are clearly located in another region of low ΔV than the ones found by the time-driven method. Nevertheless, both regions are indicated by the time-driven method and by the polar-angle-driven method, see Figures C.11 and C.16 in Appendix C. Furthermore, again the best velocity-function combinations have powers in the time-evolutions function which are one order higher than the ones in the radial-velocity function.

Table 11.13: Departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories found by lowest-order polar-angle-driven solutions with different radial-velocity and time-evolution functions for a mission to Tempel 1.

R	T	Z	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPow2Pow3	CPow2Pow3	CosP4CosP4Sin	1659	430	14.296	9.23
CPow3Pow4	CPow3Pow4	CosP4CosP4Sin	1639	470	13.978	7.54
CPow4Pow5	CPow4Pow5	CosP4CosP4Sin	1639	480	13.956	6.77
CPow2Pow4	CPow3Pow5	CosP4CosP4Sin	1629	480	13.940	7.13
CPow3Pow4	CPow4Pow5	CosP4CosP4Sin	1629	480	13.894	7.00

The Pareto front for different axial power orders and different number of revolutions can be found in Appendix C. The usage of 6$^{\text{th}}$-order axial powers in combination with $N = 1$ appeared to be best regarding ΔV. This is different from the time-driven results where $N = 0$ was found to be best.

Higher-order solutions The higher-order results of the polar-angle-driven method are, just like the lower-order results, worse than the time-driven ones. However, the

found optimal departure date and TOF are almost equal to the ones obtained by the time-driven method. The found optimal departure date of 5719 MJD2000 is, however, totally different from the ones found by the lowest-order solutions. Nevertheless, this departure date is located in a region of low ΔV which is indicated by the lowest-order solutions as well, see Figures C.16 and C.17 in Appendix C. The regions of low ΔV found by the polar-angle-driven method are, however, much smaller than the regions of low ΔV found by the time-driven method, see Figures C.12 and C.17.

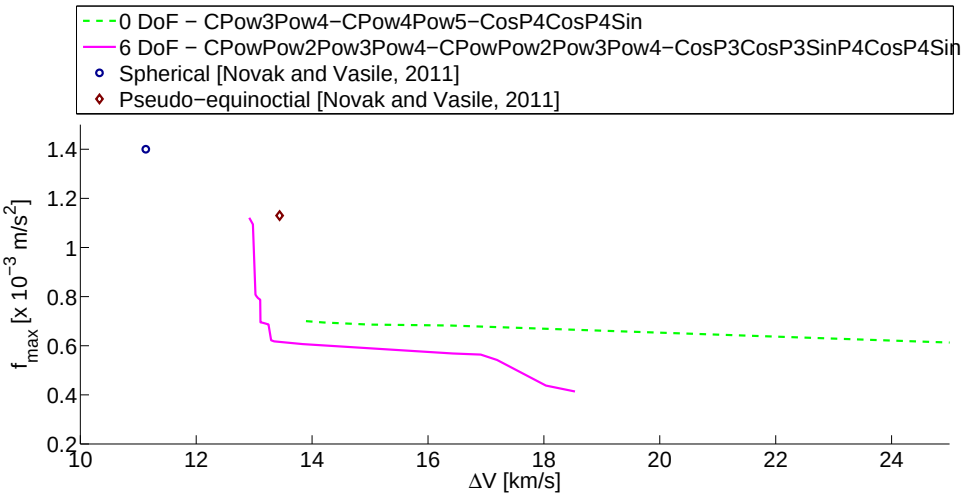


Figure 11.32: Pareto fronts of higher-order polar-angle-driven solutions for a mission to Tempel 1.

Table 11.14: Departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories (and one lower-f_{max} trajectory) found by the best lowest-order and a higher-order polar-angle-driven solution for a mission to Tempel 1.

R	T	Z	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPow3Pow4	CPow4Pow5	CosP4CosP4Sin	1629	480	13.894	7.00
CPowPow2 Pow3Pow4	CPowPow2 Pow3Pow4	CosPCosPSin P2CosP2Sin	1719	400	13.110	6.95
			5719	400	12.916	11.21

Although the required ΔV and thrust acceleration corresponding to the best polar-angle-driven result are worse than those of the best time-driven result, their values are of comparable magnitude. Since only one polar-angle-driven higher-order solution has been tested, it is expected that the polar-angle-driven method is able to perform better in case other velocity functions are used.

Comparison with other methods

Table 11.15 gives an overview of the best results obtained by different low-thrust trajectory design methods. Clearly, all shaping methods obtain trajectories which require more or less the same thrust acceleration. The trajectories found by the

hodographic-shaping methods are clearly not optimal, since the best found ΔV differs 2 km/s from the lowest ΔV obtained by DITAN.

On the other hand, compared to pseudo-equinoctial shaping, hodographic shaping provides much better results. The best lowest-order solution requires the same ΔV as the best pseudo-equinoctial shaping results and a much lower thrust acceleration of 6.1×10^{-4} m/s^2 instead of 11.3×10^{-4} m/s^2. The addition of DoF results in improved values for the required ΔV. The best obtained ΔV is 12.68 km/s, which is lower than the best ΔV of 13.44 km/s for pseudo-equinoctial shaping. This ΔV is, however, much higher than the best ΔV obtained using spherical shaping of 11.13 km/s.

Compared to the spherical and pseudo-equinoctial shaping results, the hodographic-shaping results require low thrust accelerations, see Figures 11.29 and 11.32. So, regarding f_{max} the hodographic-shaping method performs well. This observation is, however, of minor importance with respect to the fact that the hodographic-shaping results are not very good regarding ΔV.

Table 11.15: Required ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectory found by different methods.

Method	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
Hohmann	9.92	-
Hodographic - time	12.68	11.7
Hodographic - polar angle	12.92	11.2
Spherical [Novak and Vasile, 2011]	11.13	14.0
Pseudo-equinoctial [Novak and Vasile, 2011]	13.44	11.3
DITAN [Novak and Vasile, 2011]	10.69	14.0

11.5. Mercury

The Mercury case is a special one since it is the only test case with an inner body as target. For a mission to an outer body the orbital period increases and therefore even for long flight times only a few revolutions are required. However, when flying towards the Sun the orbital period decreases, and therefore multiple revolutions need to be completed when flying for a long period of time. Moreover, in order to rendezvous with Mercury the semi-major axis, eccentricity and inclination all need to be changed largely. So, to change these orbital elements gradually, a large TOF is required which consequently requires many revolutions. Therefore, the Mercury case is suited to test the many-revolution performance of the hodographic-shaping methods.

11.5.1. Shaping as function of time

Lowest-order solutions Again, first the lowest-order time-driven results are discussed.

Radial and normal velocity functions Figure 11.33 and Table 11.16 show the results for five different lowest-order velocity-function combinations. The combination CPowPow2-CPowPow2 again performs best, and the combinations which include cosine terms again worst.

Remarkable is the fact that the required thrust accelerations are very high; they are about an order of magnitude larger than the thrust accelerations found in previous test cases. In addition, the found optimal TOFs are very short; only 160 days for the best trajectory. Considering that the minimum ΔV required for a transfer to Mercury is 16.32 km/s (2D Hohmann), then with a TOF of 160 days the required average thrust acceleration is 11.8×10^{-4} m/s^2 ($= \Delta V/\text{TOF}$). Such a thrust acceleration magnitude is unfeasibly high and is only a required average. So, clearly a TOF of 160 days is too short to enable feasible thrust accelerations.

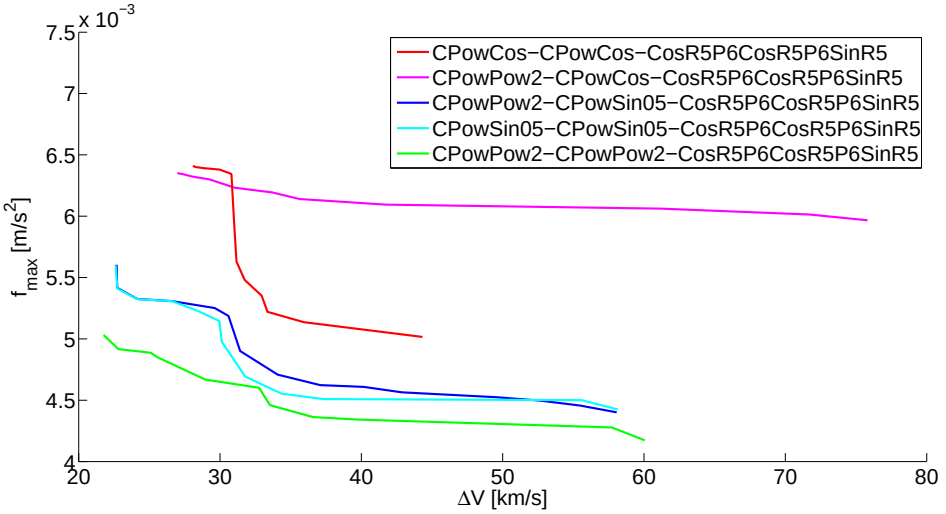


Figure 11.33: Pareto fronts of lowest-order time-driven solutions for radial- and normal-velocity functions for a mission to Mercury.

Table 11.16: Number of revolutions, departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories found by lowest-order time-driven solutions with different radial- and normal-velocity functions for a mission to Mercury.

V_r	V_θ	V_z	N	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPowCos	CPowCos	CosR5P6CosR5P6SinR5	1	5025	440	28.082	64.1
CPowPow2	CPowCos	CosR5P6CosR5P6SinR5	1	5015	450	26.997	63.5
CPowPow2	CPowSin05	CosR5P6CosR5P6SinR5	0	4675	190	22.683	56.0
CPowSin05	CPowSin05	CosR5P6CosR5P6SinR5	0	4675	190	22.613	56.0
CPowPow2	CPowPow2	CosR5P6CosR5P6SinR5	0	4355	160	21.766	50.3

Number of revolutions Figure 11.34 shows the results for different numbers of revolutions. It can be noticed that $N = 0$ is optimal, which corresponds to the short flight times of Table 11.16. Again, based on the values for f_{max}, it can be concluded that the lowest-order solutions do not obtain good (multi-revolution) trajectories.

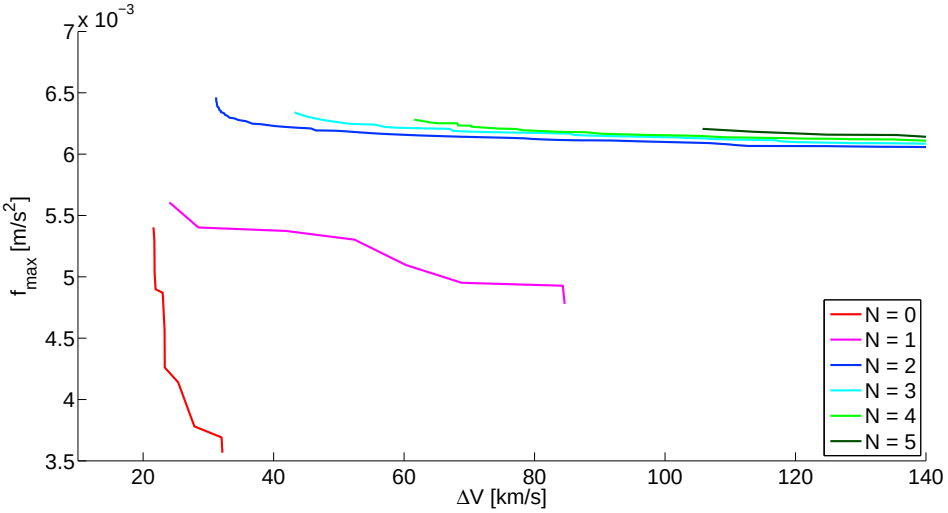


Figure 11.34: Pareto fronts of the lowest-order time-driven solution CPowPow2-CPowPow2-CosR5PCosR5PSinR5 for different number of revolutions N for a mission to Mercury.

Furthermore, it was found that an axial power order of eight is best for all numbers of revolutions. An even higher-order power might, however, be better, since this was the highest order tested. It is surprising that such a high order is best, since that means that the largest part of the inclination change takes place at the end of the trajectory where the spacecraft's velocity is very high.

Higher-order solutions Figure 11.35 and Table 11.17 show that the trajectories can be improved much by adding DoF; the ΔV is reduced by 2 km/s. Unfortunately, the required thrust acceleration is approximately the same. Considering the thrust acceleration required for the best obtained trajectory of 48.6×10^{-4} m/s^2 and assuming a spacecraft mass of 1000 kg the required thrust is 4.86 N. This thrust is not feasible using current electric-propulsion technology, see Chapter 3.

Table 11.17: Departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories found by the best lowest-order and two higher-order time-driven solutions for a mission to Mercury.

V_r	V_θ	V_z	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPowPow2	CPowPow2	CosR5P6CosR5P6SinR5	4355	160	21.766	50.3
CPowPow2 PSin05PCos05	CPowPow2 PSin05PCos05	CosR5P5CosR5P5SinR5 P6CosR5P6SinR5	3945	220	19.781	45.6
CPowPow2 PSinPCos	CPowPow2 PSinPCos	Cos05Pow5Pow6 P6Cos05P6Sin05	5285	280	19.707	48.6

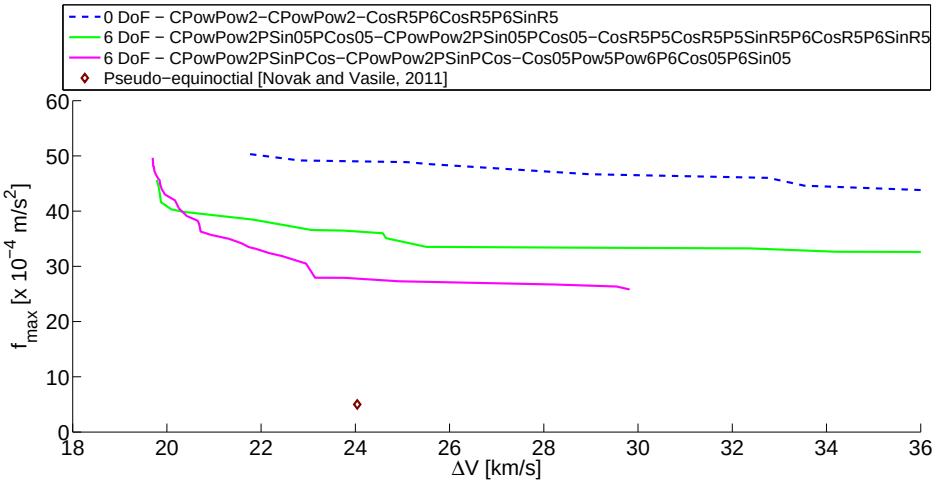


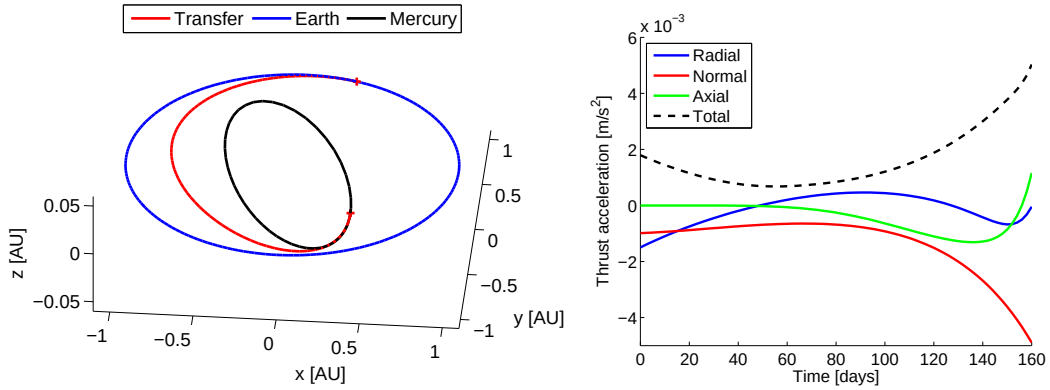
Figure 11.35: Pareto fronts of different higher-order time-driven solutions for a mission to Mercury.

Furthermore, from Tables 11.16 and 11.17 it can be noticed that the found optimal departure dates differ significantly for different velocity-function combinations. The different velocity-function combinations do, however, not find different regions of low ΔV, but they find many different regions of low ΔV, see Figures C.21 and C.22 in Appendix C. As a result, for many different departure dates low-ΔV trajectories can be found. The patterns in Figures C.21 and C.22 are periodic with the synodic period of Earth and Mercury of 116 days and with the orbital period of Earth of 365 days. These short periods explain why many regions of low ΔV are found.

In addition, it can be noticed that in the ΔV plots the regions of low ΔV run purely diagonally from long TOFs to late departure dates. As in the Tempel 1 case, this means that the required ΔV is very dependent on the arrival date and little dependent on the departure date. This seems to be logical since the Earth orbit is near-circular, whereas Mercury's orbit is highly eccentric. Therefore, the moment at which the rendezvous takes place has much more effect on the trajectory than the moment at which one departs from Earth.

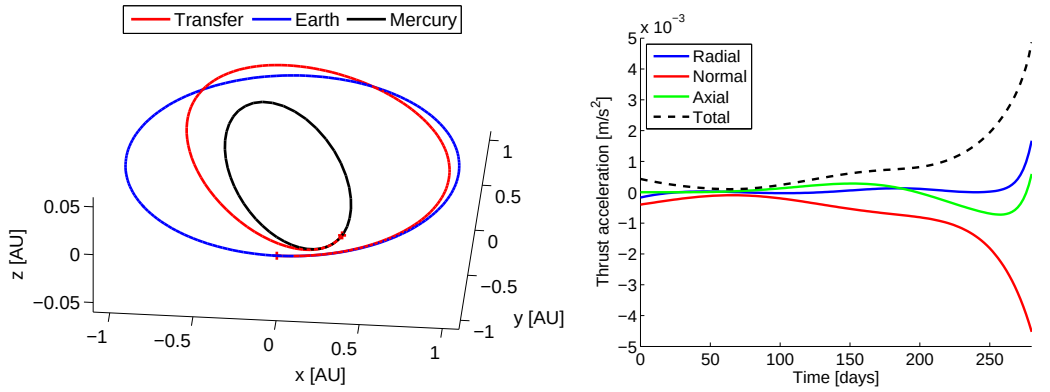
Trajectories Figures 11.36 and 11.37 show the trajectories and thrust profiles of the best lowest-order and higher-order results. The thrust profiles show that the additional DoF mainly reduce the ΔV by decreasing the radial thrust. Furthermore, both thrust profiles include a peak thrust at the end of the trajectory, which is primarily caused by the normal thrust component. This indicates that the spacecraft has to brake a lot just before the rendezvous with Mercury.

Peaks in the normal thrust component were also found for trajectories to Tempel 1. These peaks are related to the change in orbital energy. The change of the orbital energy over time corresponding to the best found trajectories of each test case is shown in Figure 11.38. This figure clearly shows that the orbital energy is changed most gradually during the transfer to Mars and least gradually during the



(a) Trajectory (b) Thrust profile

Figure 11.36: Trajectory and thrust profile of best lowest-order time-driven solution for Earth-Mercury: V_r = CPowPow2, V_θ = CPowPow2 and V_z = CosR5P6CosR5P6SinR5, see Table 11.16.



(a) Trajectory (b) Thrust profile

Figure 11.37: Trajectory and thrust profile of best higher-order time-driven solution for Earth-Mercury: V_r = CPowPow2PSinPCos, V_θ = CPowPow2PSinPCos and V_z = Cos05Pow5Pow6P6Cos05P6Sin05, see Table 11.17.

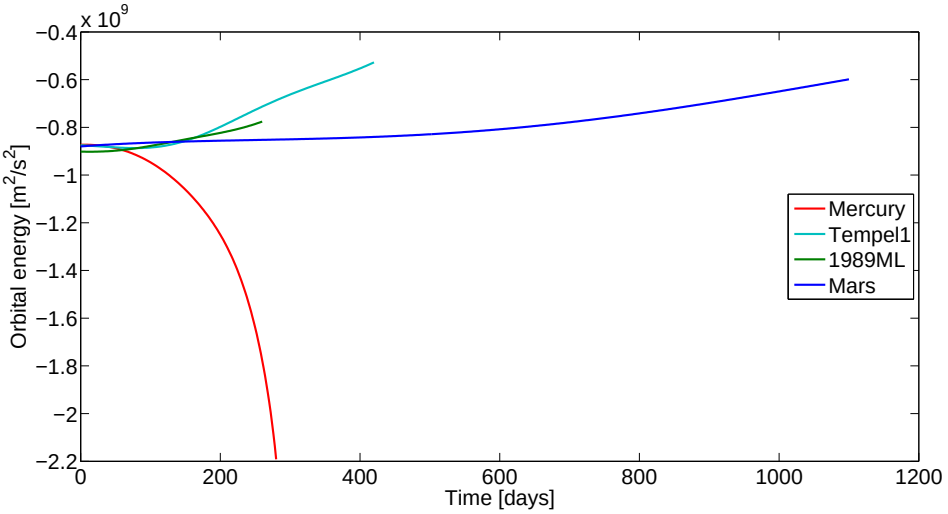


Figure 11.38: Orbital energy over time for best-ΔV trajectories found for missions to Mars, 1989ML, Tempel 1 and Mercury [3].

transfer to Mercury. The steepness of the orbital energy plots are clearly correlated to the required (normal) thrust acceleration. Therefore, if the orbital energy for a transfer to Mercury would be changed more gradually over time, the required thrust acceleration would be much lower. This can be achieved by increasing the TOF such that the steepness of the orbital energy change is reduced.

As discussed, increasing flight times require multi- or even many-revolution trajectories. However, when the spacecraft is flying towards the Sun, the local orbital characteristics, such as the orbital velocity and orbital period, change considerably. Therefore the velocity functions need to be able to cope with these large changes. The periodicity of the velocity functions is, however, fixed and the time-driven velocity functions are therefore not well able to deal with changing orbital periods. The polar-angle-driven velocity functions, on the other hand, should be better able to cope with changing orbital periods, since they depend on the polar angle instead of time. The velocity functions are therefore always one-revolution periodic, independent of the local orbital period.

11.5.2. Shaping as function of polar angle

Lowest-order solutions In Table 11.18 the characteristics of the best lowest-order polar-angle-driven solutions are shown. The results are clearly much better than the ones of the time-driven method, especially regarding ΔV. Furthermore, the optimal flight times are in general much longer than the ones found by the time-driven method and the number of revolutions is on average one higher. So, the polar-angle-driven method is able to obtain good trajectories with relatively long flight times. The required thrust accelerations are, however, still unfeasibly high.

Table 11.18: Number of revolutions, departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories found by lowest-order polar-angle-driven solutions with different radial-velocity and time-evolutions functions for a mission to Mercury.

R	T	Z	N	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPow2Pow3	CPow2Pow3	CosPCosPSin	1	3965	460	21.839	26.7
CPowPow3	CPowPow3	CosPCosPSin	2	3965	720	21.083	33.2
CPow2Pow3	CPow3Pow4	CosPCosPSin	1	3635	430	20.888	72.2
CPow2Pow4	CPow3Pow5	CosPCosPSin	1	3845	320	20.888	127.5
CPowPow2	CPow2Pow3	CosPCosPSin	1	4935	280	20.612	68.3
CPowPow3	CPow2Pow4	CosPCosPSin	2	4195	580	20.180	41.8

Figure 11.39 shows the Pareto fronts for different numbers of revolutions for the velocity-function combination CPowPow3-CPow2Pow4. A number of two revo-

[3]It can be noticed that the orbital-energy plots for the different transfers do not start at the same orbital-energy value. This is the result of the fact that the Earth's orbit used for the computations is non-Keplerian. Therefore the orbital energy of the Earth changes over time.

lutions appears to be optimal regarding ΔV. Furthermore, regarding required thrust acceleration $N = 2$ also performs best for low ΔV and equally to $N = 1$ and $N = 3$ for higher ΔV. So, the polar-angle-driven method performs better for multi-revolution transfers to Mercury than the time-driven method. In addition, Figure C.24 in Appendix C shows that the application of 1st- or 2nd-order axial powers provides the best results.

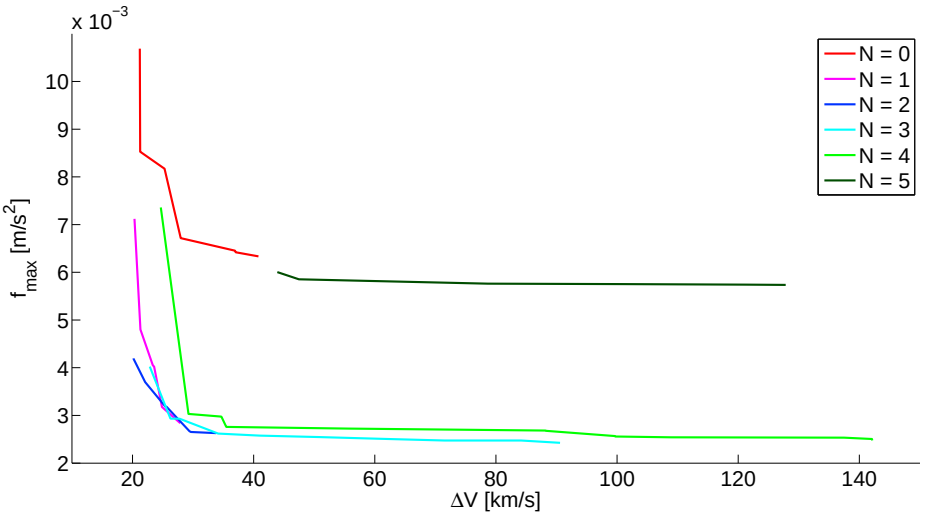


Figure 11.39: Pareto fronts of the lowest-order polar-angle-driven solution CPowPow3-CPow2Pow4-CosPCosPSin for different number of revolutions N for a mission to Mercury.

Higher-order results The best results of one higher-order solution are shown in Figure 11.40 and Table 11.19. The flight times of the obtained low-ΔV trajectories are short in contrast to the flight times corresponding to the lowest-order solutions, but similar to the ones found by the time-driven solutions.

Compared to the time-driven method, the polar-angle-driven method performs better regarding both the required ΔV and thrust acceleration. The best found ΔV is 19.38 km/s, which is 0.32 km/s smaller than the best time-driven one. Furthermore, a trajectory was found which requires a thrust acceleration of only $2.7{\times}10^{-3}$ m/s^2 and a ΔV which is only 0.14 km/s higher than the best ΔV. This thrust acceleration is significantly lower than $4.9{\times}10^{-3}$ m/s^2 and $9.1{\times}10^{-3}$ m/s^2 corresponding to the lowest-ΔV trajectories obtained by the time-driven and polar-angle-driven method, respectively. Furthermore, the Pareto fronts in Figure 11.40 show that the polar-angle-driven trajectories require much less thrust acceleration then the time-driven ones. So, both regarding ΔV and thrust acceleration the polar-angle-driven method finds good trajectories.

Furthermore, if one compares the ΔV plots of the time-driven and polar-angle-driven method, see Figures C.21, C.22, C.25 and C.26 in Appendix C, respectively, it is clear that the polar-angle-driven method finds much better results for larger TOF. So, the polar-angle-driven method is indeed better able to obtain good many-revolution trajectories to inner targets than the time-driven method.

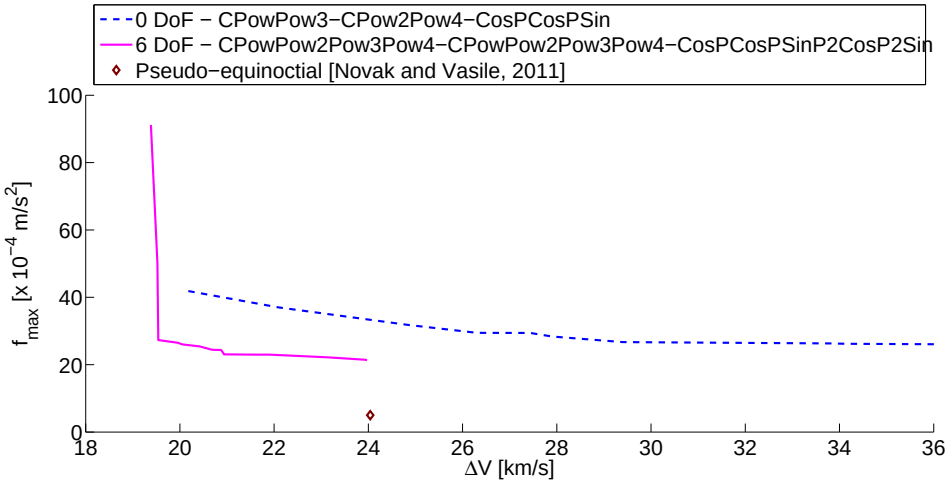


Figure 11.40: Pareto fronts of higher-order polar-angle-driven solutions for a mission to Mercury.

Table 11.19: Departure date, time of flight, ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectories (and one lower-f_{max} trajectory) found by the best lowest-order and a higher-order polar-angle-driven solution for a mission to Mercury.

R	T	Z	Departure [MJD2000]	TOF [days]	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
CPowPow3	CPow2Pow4	CosPCosPSin	4195	580	20.180	41.8
CPowPow2 Pow3Pow4	CPowPow2 Pow3Pow4	CosPCosPSin P2CosP2Sin	4285	240	19.541	27.3
			3965	200	19.384	91.2

Comparison with other methods Table 11.20 shows the best-ΔV results for different low-thrust methods. The ΔVs obtained by the hodographic shaping methods are clearly much lower than the ones obtained by the pseudo-equinoctial shaping method and by DITAN and much closer to the Hohmann-transfer ΔV. So, the hodographic-shaping methods perform well regarding ΔV. It should, however, be noticed that the pseudo-equinoctial and DITAN results have been obtained by optimizating including a constraint on the maximum thrust acceleration. So, the pseudo-equinoctial shaping method and DITAN may have found trajectories which require less ΔV if the thrust accelerations were not constrained.

The overall lowest required thrust acceleration found by the hodographic-shaping methods is about 20×10^{-4} m/s^2, which is four times higher than the 5×10^{-4} m/s^2 required for the best pseudo-equinoctial-shaped trajectory. So, one can say that the hodographic-shaping method is able to find low-ΔV trajectories to Mercury at the cost of relatively high required thrust accelerations.

11.6. Computational Effort

In the previous sections the results of the hodographic-shaping methods for all test cases have been discussed regarding ΔV and thrust acceleration. In addition

Table 11.20: Required ΔV and maximum thrust acceleration corresponding to minimum-ΔV trajectory found by different methods for a mission to Mercury.

Method	ΔV [km/s]	f_{max} [10^{-4} m/s^2]
Hohmann	16.32	-
Hodographic - time	19.71	48.6
Hodographic - polar angle	19.38	91.2
Pseudo-equinoctial [De Pascale and Vasile, 2006]	24.04	5.0
DITAN [De Pascale and Vasile, 2006]	22.72	5.0

to providing good results, it is, however, also important for a low-thrust trajectory design method to obtain the results very fast. Therefore, in this section the hodographic-shaping method's performance regarding computational effort is discussed.

The computational effort has been measured in time. For this the computations have been done with an Intel Core i5-430M processor (2.27 GHz) on a 64-bits Windows platform. Furthermore, the hodographic-shaping method has been programmed in C++.

It was found that the required computation time depends mainly on the number of DoF. The application of more DoF complicates the optimization, requiring more function evaluations per trajectory optimization. To a lesser extent the computational effort depends on the applied velocity functions and number of revolutions. For example, (higher-order) power-times-trigonometric base functions require (much) more time for the evaluation of their integral than other base functions, since their integral is a series, see Section 5.4.1. In addition, the complexity of optimizing free coefficients can differ due to the different behavior of the corresponding base functions. Furthermore, for some combinations of TOF and number of revolutions it is very hard to find proper trajectories. This is especially the case for short flight times in combination with many revolutions. In such cases the optimal values of the free coefficients are often very large and therefore the optimizer may need much time to find these optimal values. Finally, the computational effort depends on the applied optimization method and corresponding settings, which were discussed in Chapter 9.

So, in short, the computations take least time when the number of DoF is minimum, when no power-times-trigonometric base functions are used and when the optimal number of revolutions is applied together with a fast optimization method.

In the following first the computation time required for the lowest-order solutions and after that the time required to obtain optimal higher-order solutions is discussed. Finally, the computation times of the hodographic-shaping method and other shaping methods are compared.

11.6.1. Lowest-order solutions

For the lowest-order case the computation time appeared to depend on the choice of the axial power order only. The required computation times for different axial power orders and missions are shown in Table 11.21. These times are identical for the time-driven and polar-angle-driven method. Clearly, the computations are done very fast and one trajectory computation may require less than one millisecond. In addition, the computation time increases with approximately 0.15 ms when the order of the axial powers is increased by one. Furthermore, the trajectory computations for missions to Mars and Mercury requires slightly more computation time.

A very short computation time of only 1.5 ms means, in practice, that the entire flight window of the Tempel 1 test case can be searched with a grid size of 10 days (65,000 grid points) within 100 s. So, using lowest-order solutions one can very quickly obtain an indication of the required ΔV and find regions in the flight windows where the ΔV is low.

Table 11.21: Computation times required to compute one trajectory in milliseconds using lowest-order solutions for different missions and axial power orders. These computation times apply to the time-driven lowest-order solution: V_r = CPowPow2, V_θ = CPowPow2, V_z = CosR5PCosR5PSinR5 and to the polar-angle-driven lowest-order solution: R = CPow2Pow4, T = CPow3Pow5, Z = CosPCosPSin.

Axial power	Mars	1989ML	Tempel 1	Mercury
1^{st} order	1.0	0.9	0.9	1.0
3^{rd} order	1.3	1.2	1.2	1.3
5^{th} order	1.6	1.5	1.5	1.7
7^{th} order	2.0	1.9	1.9	2.0

11.6.2. Higher-order solutions

For the higher-order solutions the computation time also depends on the number of DoF and the applied number of revolutions. It was found that for a non-optimal number of revolutions, the computation time can be up to five times longer than for the optimal number of revolutions. However, usually the difference in computation times for different number of revolutions is less than a factor two.

In Table 11.22 the average computation times for different DoF for the four test cases are shown. These results are averages for different numbers of revolutions and different velocity functions. The applied axial power orders are the optimal orders regarding ΔV.

The computation times for the time-driven method for 3 DoF are about one hundred times longer than for 0 DoF, and for 6 DoF a thousand times longer. On average applying an extra DoF increases the computational effort by a factor three. The trajectories in the polar-angle-driven method are a bit faster optimized than

those in the time-driven method, especially for missions to asteroid 1989ML and Mercury.

Table 11.22 also shows the times required to compute one trajectory with the spherical and pseudo-equinoctial shaping methods as presented in [Novak and Vasile, 2011]. In comparison with the spherical and pseudo-equinoctial shaping methods, the hodographic-shaping method performs very well. The lowest-order solutions (0 DoF) are computed up to 200 times faster. Furthermore, up to 3 DoF the hodographic-shaping computations are done much faster than the spherical and pseudo-equinoctial shaping calculations. For 6 DoF the hodographic-shaping method, however, requires, on average, 4 times (polar-angle-driven) to 7 times (time-driven) the computation time of the spherical and pseudo-equinoctial shaping methods.

It should be noted that an one-on-one comparison of the computation times of the different shaping methods may not be valid. First of all, the computations have been done on different machines with probably different clock speeds. In addition, the hodographic-shaping method has been coded in C++, whereas the spherical and pseudo-equinoctial shaping methods were coded in MATLAB. C++ programs in general run much faster than MATLAB programs, so this may explain partly why the hodographic shaping method is faster for low numbers of DoF. However, it is known that spherical and pseudo-equinoctial shaping require several iterations to solve all boundary conditions whereas hodographic shaping does not. Furthermore, spherical and pseudo-equinoctial shaping require the solving of 12 boundary conditions and hodographic shaping only needs to solve 9 boundary conditions, see Section 5.2.1. Therefore, hodographic shaping with 0 DoF is most likely the fastest method regardless of the machine speed and coding language.

Table 11.22: Average computation times required to compute one trajectory in milliseconds for different methods and different missions; † without initial guesses.

	Mars	1989ML	Tempel 1	Mercury	Average
Hodographic - t - 0 DoF	1.3	1.8	0.9	2.2	1.6
Hodographic - t - 2 DoF	-	65	38	-	52
Hodographic - t - 3 DoF	151	-	-	-	151
Hodographic - t - 4 DoF	-	679	-	-	679
Hodographic - t - 6 DoF	1191	2056	1036	3597	1970
Hodographic - θ - 0 DoF	1.3	1.8	0.9	2.2	1.6
Hodographic - θ - 6 DoF	938	670	2017†	973	1150
Spherical	316	316	318	-	317
Pseudo-equinoctial	238	264	286	-	263

Now all results have been discussed, conclusions about these results are drawn in the following section. In the next chapter conclusions regarding the overall research are given.

11.7. Conclusions

The results in the previous sections show that the hodographic-shaping methods are able to quickly obtain good trajectories. In addition, the hodographic-shaping methods perform best for trajectories with low numbers of revolutions. For many-revolutions trajectories the methods have problems to obtain good trajectories.

Time-driven versus polar-angle-driven Regarding ΔV the time-driven method and polar-angle-driven method give comparable results for missions to Mars and asteroid 1989ML. For a mission to comet Tempel 1 the time-driven method, however, performs better and for a mission to Mercury the polar-angle-driven method gives better results (especially for multi-revolution trajectories). On the other hand, regarding thrust acceleration the time-driven method performs better for all missions except for a mission to Mercury.

Lowest-order solutions First of all, the lowest-order solutions can be computed very fast, within 1 or 2 milliseconds. Furthermore, the ΔV and thrust accelerations obtained by the lowest-order solutions give a good indication of optimal ΔV and thrust acceleration values; the ΔV values are at maximum 15% off. Therefore, the lowest-order solutions can be used to search for e.g. feasible/good target bodies. In addition, for all test cases it was found that the lowest-order solutions are capable of indicating the correct regions of low ΔV. Therefore, the lowest-order solutions may be used to search the flight window for good departure dates and TOF. Subsequently the flight window can be reduced and searched using higher-order velocity models to obtain near-optimal trajectories. Furthermore, since the thrust profiles of the best lowest-order result were found to be already close to the optimal thrust profiles, the lowest-order results may also be used directly as initial guess for a subsequently trajectory optimization. Finally, since lowest-order solutions can be obtained very fast, they are suited to search for feasible target bodies out of many possible targets, where nowadays this is often done using Lambert targeting which is not related to low-thrust.

Higher-order solutions The higher-order solutions require much more computational effort than the lowest-order solutions. The obtained trajectories are, however, much better regarding ΔV and f_{max} and the higher-order solutions depend less on the departure date and TOF. For missions to Mars and 1989ML very good trajectories have been found. Higher-order solutions can therefore very well be used to find near-optimal trajectories.

Hodographic shaping versus other shaping methods Using only 3 DoF the hodographic-shaping method provides better results regarding ΔV than the pseudo-equinoctial shaping method for all test cases. Compared to the spherical-shaping

method, the hodographic-shaping method is only better regarding ΔV for a mission to asteroid 1989ML. Furthermore, when comparing the results with results of DITAN, the obtained trajectories to Mars and 1989ML can be said to be near-optimal regarding both ΔV and thrust acceleration. So, the results provided by the hodographic-shaping methods range from quite good to very good compared to other shaping methods and optimal trajectories found by DITAN.

Best lowest-order solutions For the time-driven method the radial- and normal-velocity-function combination CPowPow2-CPowPow2 was found to perform best for all test cases. For the axial velocity the function CosR5PCosR5PSinR5 performs very well. In this axial-velocity function the order of the powers needs to be tuned to obtain better trajectories. Furthermore, a frequency of 0.5 instead of $N + 0.5$ may for some missions provide better results. For the polar-angle-driven method it was found that the application of purely powers in the radial-velocity and time-evolution function gives good results. Especially, when the powers in the time-evolution functions are one order higher than those in radial velocity function. In all test cases, except Mercury, the combination CPow2Pow4-CPow3Pow5 was found to perform best. For the axial velocity the function CosPCosPSin performs very well, where again the order of powers may need fine tuning.

Best higher-order solutions For the time-driven method the best higher-order solutions were found by simply expanding the best lowest-order solution. The radial- and normal-velocity functions can best be expanded with PSinPCos and the axial-velocity function with PCosR5PSinR5 with a proper power order. Furthermore, instead of Pow2 as third base function one can use Sin05 and instead of PSinPCos, PSin05PCos05 may provide better results. For the polar-angle-driven method only one higher-order solution has been tested. This solution provides rather good results, but better results may be found using other velocity functions.

12. Conclusions

The goal of this research is to develop an analytic method for preliminary design of interplanetary low-thrust trajectories. Inspired by the use velocity hodographs for the computation of non-perturbed transfer orbits, a novel low-thrust trajectory design method is developed based on shaping the velocity of spacecraft during the transfer.

Orbital mechanics In order to reduce the complexity of the trajectory design problem, all forces except the gravitational attraction from the Sun and the thrust force are neglected. Since the preliminary design of low-thrust trajectories is considered, neglecting of perturbing forces is acceptable for first-order results.

Hodographic-shaping method For the shaping of the velocity components, velocity functions are used which consist of a sum of simple base functions. These base functions can be integrated analytically, such that the change in position can be obtained analytically. Since the velocity is shaped, the conditions on initial and final velocity can be solved very easily and exactly. In addition, the boundary conditions on position can be solved exactly without the need of iterative computations. Furthermore, since velocities are shaped, only nine boundary conditions need to be satisfied for a three-dimensional rendezvous trajectory, namely two on velocity and one position for each direction. Next to nine parameters required to satisfy the boundary conditions, extra parameters can be added to make the design and optimization of trajectories more flexible. Moreover, no constraint on the thrust direction is required and therefore the thrust vector is allowed to point in any direction. Finally, two different methods have been developed; one which shapes the velocity as a function of time and another one which shapes as a function of the polar angle.

Numerical integration For the time-driven method the computation of the polar angle requires numerical integration. In addition, both methods require numerical integration for the computation of the required ΔV. These integrations are done using an RK4 integrator. Since both the derivatives of ΔV and polar angle (the thrust acceleration and angular velocity, respectively) can be obtained as function of the independent variable, the derivative values can be computed prior to the integration. As a result the RK4 integration scheme can be expressed as Simpson's rule and only two derivative values need to be computed per integration step. The accuracy of the computed ΔV and polar angle was found to be sufficient using only 25 integration steps. However, the accuracy can simply be improved by increasing the number of steps.

Verification The hodographic-shaping methods have been verified by numerically propagating the initial spacecraft state vector using thrust profiles found by

the shaping methods. Both the time-driven and polar-angle-driven method were found to function well, and the obtained trajectories and thrust profiles agree almost exactly.

Optimization For the optimization of the free coefficients of the velocity function a Nelder-Mead technique was applied. This optimizer was found to be much faster than a Differential Evolution optimizer and also more robust for trajectories which require little ΔV. In addition, the initial guesses for the free coefficients have been used to speed up the optimization by a factor 2 to 3. These initial guesses work very well, except for complex mission geometries for the polar-angle-driven method. Finally, the speed of the optimization depends on the number of optimization parameters (DoF), the optimizer settings and the mission geometry, such as the number of revolutions.

Results Both hodographic-shaping methods have been tested for missions to Mars, the near-Earth asteroid 1989ML, comet Tempel 1 and Mercury. For missions to Mars and 1989ML the two methods give comparable results. The time-driven method was, however, better able to find good trajectories to Tempel 1 (a far outer target), whereas the polar-angle-driven method was found to obtain better trajectories to Mercury (an inner target). Regarding maximum required thrust acceleration, the time-driven method gives better results, except for a mission to Mercury. Furthermore, the hodographic-shaping methods were found to be unable to obtain good many-revolution trajectories. Only up to two full revolutions the obtained trajectories are good.

In comparison with the spherical and pseudo-equinoctial shaping methods, the hodographic-shaping methods perform well for a mission to Mars and very well for a mission to asteroid 1989ML. For a mission to Tempel 1, trajectories were found which require low thrust accelerations, but, in addition, not very low ΔV. On the other hand, for a mission to Mercury low ΔV were obtained, but high thrust acceleration. Furthermore, compared with the results of DITAN (the benchmark), the obtained trajectories to Mars and 1989ML can be said to be near-optimal regarding both ΔV and thrust acceleration. So, overall the hodographic-shaping methods perform very well.

The near-optimal results come at the cost of computational speed. The best results were found using 6 DoF which require a computation time of 2 s per trajectory on average. The lowest-order solutions (0 DoF), on the other hand, require only 1.6 ms per trajectory on average. So, the hodographic-shaping methods are very fast in case of no DoF and moderately fast with 6 DoF. As a trade-off, for example 3 DoF can be used such that the results as well as the computation speed are rather good.

Finally, the lowest-order solutions were found to be unable to obtain near-optimal trajectories. However, they are able to indicate the correct regions of low ΔV in the flight windows and to give a good indication of the required ΔV and thrust

acceleration. Therefore, lowest-order results are very suitable as initial guess for further optimizations, and lowest-order solutions can be used to very quickly search for feasible target bodies out of many possible targets.

13. Recommendations

In this chapter some recommendations are given for further research on and development of the hodographic-shaping methods.

- Spherical instead of cylindrical coordinates can be used to shape velocity hodographs. This is beneficial since the spherical radial coordinate coincides with the distance from the Sun and therefore shaping of the spherical radial velocity is more related to two-body dynamics than shaping of the cylindrical radial velocity. Since a solution was found to solve the boundary conditions on angular coordinates exactly, spherical coordinates can be used without loss of accuracy or computational speed.

- Novak, as well as Petropoulos and Wall, constrained the in-plane thrust component to be tangential to the velocity vector [Novak and Vasile, 2011] [Petropoulos and Longuski, 2004] [Wall and Conway, 2009]. The condition for in-plane tangential thrust for spherical coordinates is given by Eqs. (4.14) and (4.15). A similar condition can be found for cylindrical coordinates and can be implemented in the polar-angle-driven method. It should be investigated whether such a condition results in better trajectories.

- The objective of the optimization of the velocity functions was to minimize the required ΔV. In addition to this objective, a constraint on the thrust acceleration can be implemented using e.g. penalty functions. It should be investigated whether such a constrained optimization results in better trajectories regarding thrust acceleration and ΔV. In addition to that, the objective of minimizing the required ΔV can be changed to minimizing the propellant mass ratio or maximizing the final mass, since that may give better results [De Vogeleer, 2008].

- For the polar-angle-driven method only one higher-order solution has been tested. The results of other higher-order solutions need to be investigated to be able to assess the performance of the polar-angle-driven method well.

- The local optimizer Nelder-Mead was found to perform very well for optimizing higher-order solutions up to 6 DoF. It should be investigated whether other local optimizers perform even better on this optimization problem, especially regarding computation time.

- For missions to comet Tempel 1 and Mercury minimum-ΔV trajectories were found which require high thrust accelerations. These trajectories were obtained by simulating a flight from Earth to Tempel 1 or Mercury. However, instead one could simulate the flight by flying from Tempel 1 or Mercury backwards to the Earth. In that case one starts off with the orbital characteristics of the target body instead of those of the Earth. It should be investigated whether such an approach provides better results.

- The shaping of the velocity hodographs has been done by shaping the velocity as function of time or polar angle. A velocity hodograph can in the

planar case also be parameterized by the two orbital-hodograph parameters; namely the radius of hodograph R and the distance of the center from the origin C, see Figure 2.3. So, instead of shaping the velocity components the orbital-hodograph parameters R and C may be shaped to obtain a transfer trajectory. It should be investigated whether this could result in good transfer trajectories.

- One might argue that shaping of velocity hodographs in velocity space is equivalent to shaping trajectories in real space, since velocity is only a derivative of traveled distance[4]. Therefore, it might be interesting to find out whether the approach of the hodographic-shaping methods also works well in real space, i.e. use analytically integrable functions to shape the trajectory. For the time-driven method one would then shape r, θ and z as a function of time t and for the polar-angle-driven method one would shape r, t and z as a function of the polar angle θ. In case trajectories are shaped instead of velocity profiles, more boundary conditions need to be solved. As a result more-complex velocity functions can be used as lowest-order solutions, which may be advantageous.
- To assess the complexity of the optimization problem, the solution space has been analyzed. This has been done by performing Monte-Carlo searches in the search space. However, it was found that using grid searches instead of Monte Carlo searches the solution space can be investigated much easier and better. The reason for this is that one can analyze cross-sections of the solution space using grid-search results, since all sample points are aligned. This is not possible using Monte-Carlo-search results, since sample points are located randomly.
- Finally, various velocity-function combinations have been tested. However, not all possible base functions have been considered in this research. e.g. the application of exponential base functions has not been discussed. Therefore, the use of base functions other than those applied in the test cases should be investigated.

[4]It is however problematic to obtain a function which derivative is the normal velocity V_θ.

References

[Ali and Törn, 2004] Ali, M. and Törn, A. (2004). Population set-based global optimization algorithms: some modifications and numerical studies. *Computers & Operations Research*, 31(10):1703 – 1725.

[Altman, 1965] Altman, S. P. (1965). *Orbital hodograph analysis*. AAS Science and Technology Series. Printed and distributed by Western Periodicals Co.

[Benkhoff et al., 2010] Benkhoff, J., van Casteren, J., Hayakawa, H., Fujimoto, M., Laakso, H., Novara, M., Ferri, P., Middleton, H., and Ziethe, R. (2010). BepiColombo–comprehensive exploration of Mercury: Mission overview and science goals. *Planetary and Space Science*, 58(1-2):2 – 20. Comprehensive Science Investigations of Mercury: The scientific goals of the joint ESA/JAXA mission BepiColombo.

[Benyon, 1976] Benyon, P. R. (1976). Statistical algorithms: Remark AS R15: Function minimization using a simplex procedure. 25(1):97–97. See [O'Neill, 1971, Chambers and Ertel, 1974].

[Burkardt, 2008] Burkardt, J. (2008). ASA047 - Nelder-Mead Minimization Algorithm. `http://people.sc.fsu.edu/~jburkardt/cpp_src/asa047/asa047.html`. [Online; accessed 24-May-2012].

[Chambers and Ertel, 1974] Chambers, J. M. and Ertel, J. E. (1974). Statistical algorithms: Remark AS R11: A remark on Algorithm AS 47 "Function Minimization using a Simplex Procedure". 23(2):250–251. See [O'Neill, 1971,Benyon, 1976, Hill, 1978].

[Cornelisse et al., 1979] Cornelisse, J., Schöyer, H., and Wakker, K. (1979). *Rocket propulsion and spaceflight dynamics*. Aerospace Engineering Series. Pitman.

[De Pascale and Vasile, 2006] De Pascale, P. and Vasile, M. (2006). Preliminary design of low-thrust multiple gravity-assist trajectories. *Journal of Spacecraft and Rockets*, 43(5):1065–1076.

[De Vogeleer, 2008] De Vogeleer, B. (2008). Automatic and fast generation of sub-optimal and feasible low-thrust trajectories using a boundary-value pseudo-spectral method. Master's thesis, Delft University of Technology, Delft, Netherlands.

[ESA, 2004] ESA (2004). Science and Technology: SMART-1. `http://smart.esa.int`. [Online; accessed 25-July-2011].

[Feoktistov, 2006] Feoktistov, V. (2006). *Differential Evolution: In Search of Solutions (Springer Optimization and Its Applications)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

[Gao, 2007] Gao, Y. (2007). Near-optimal very low-thrust Earth-orbit transfers and guidance schemes. *Journal of Guidance, Control, and Dynamics*, 30(2):529–539.

[Hill, 1978] Hill, I. D. (1978). Statistical algorithms: Remark AS R28: A remark on Algorithm AS 47: Function minimization using a simplex procedure. 27(3):380–382. See [O'Neill, 1971, Chambers and Ertel, 1974].

[Izzo, 2006] Izzo, D. (2006). Lambert's problem for exponential sinusoids. *Journal of Guidance, Control, and Dynamics*, 29(5):1242–1245.

[Kawaguchi et al., 2006] Kawaguchi, J., Kuninaka, H., Fujiwara, A., and Uesugi, T. (2006). MUSES-C, its launch and early orbit operations. *Acta Astronautica*, 59(8-11):669 – 678. Selected Proceedings of the Fifth IAA International Conference on Low Cost Planetary Missions.

[Kuninaka, 2008] Kuninaka, H. (2008). Microwave discharge ion engines onboard Hayabusa asteroid explorer. *AIP Conference Proceedings*, 997(1):572–581.

[Lantoine and Russell, 2009] Lantoine, G. and Russell, R. (2009). The Stark model: An exact, closed-form approach to low-thrust trajectory optimization. In *International Symposium on Space Flight Dynamics, Toulouse, France*.

[McConaghy et al., 2002] McConaghy, T., Debban, T., Petropoulos, A., and Longuski, J. (2002). An approach to design and optimization of low-thrust trajectories with gravity assists. In *Advances in the Astronautical Sciences*, volume 109 III.

[McConaghy et al., 2003] McConaghy, T., Debban, T., Petropoulos, A., and Longuski, J. (2003). Design and optimization of low-thrust trajectories with gravity assists. *Journal of Spacecraft and Rockets*, 40(3):380–387.

[Montenbruck and Gill, 2000] Montenbruck, O. and Gill, E. (2000). *Satellite orbits: models, methods, and applications*. Physics and Astronomy Online Library. Springer.

[NASA - JPL, 2009] NASA - JPL (2009). Dawn mission. `http://dawn.jpl.nasa.gov/mission/`. [Online; accessed 25-July-2011].

[NASA - JPL, 2012a] NASA - JPL (2012a). Solar System Dynamics - Keplerian Elements for Approximate Positions of the Major Planets. `http://ssd.jpl.nasa.gov/?planet_pos`. [Online; accessed 8-June-2012].

[NASA - JPL, 2012b] NASA - JPL (2012b). Solar System Dynamics - Small-Body Orbital Elements. `http://ssd.jpl.nasa.gov/?sb_elem`. [Online; accessed 8-June-2012].

[Noomen, 2010a] Noomen, R. (2010a). *Lecture Notes AE4878 - Mission Geometry & Orbit Design: Integrators*. Delft University of Technology, Delft, Netherlands.

[Noomen, 2010b] Noomen, R. (2010b). *Lecture Notes AE4878 - Mission Geometry & Orbit Design: Optimization*. Delft University of Technology, Delft, Netherlands.

[Novak and Vasile, 2011] Novak, D. and Vasile, M. (2011). Improved shaping ap-

proach to the preliminary design of low-thrust trajectories. *Journal of Guidance, Control, and Dynamics*, 34(1):128–147.

[O'Neill, 1971] O'Neill, R. (1971). Statistical algorithms: Algorithm AS 47: Function minimization using a simplex procedure. 20(3):338–345. See remarks [Chambers and Ertel, 1974, Benyon, 1976, Hill, 1978].

[Petropoulos and Longuski, 2002] Petropoulos, A. and Longuski, J. (2002). A shape-based algorithm for the automated design of low-thrust, gravity-assist trajectories. In *Advances in the Astronautical Sciences*, volume 109 III.

[Petropoulos and Longuski, 2004] Petropoulos, A. and Longuski, J. (2004). Shape-based algorithm for automated design of low-thrust, gravity-assist trajectories. *Journal of Spacecraft and Rockets*, 41(5):787–796.

[Petropoulos and Sims, 2002] Petropoulos, A. and Sims, J. (2002). A review of some exact solutions to the planar equations of motion of a thrusting spacecraft. In *2nd International Symposium Low Thrust Trajectories, Toulouse, France*. NASA.

[Press, 2007] Press, W. (2007). *Numerical recipes: the art of scientific computing*. Cambridge University Press.

[Racca et al., 2002] Racca, G. D., Marini, A., Stagnaro, L., van Dooren, J., di Napoli, L., Foing, B. H., Lumb, R., Volp, J., Brinkmann, J., Grünagel, R., Estublier, D., Tremolizzo, E., McKay, M., Camino, O., Schoemaekers, J., Hechler, M., Khan, M., Rathsman, P., Andersson, G., Anflo, K., Berge, S., Bodin, P., Edfors, A., Hussain, A., Kugelberg, J., Larsson, N., Ljung, B., Meijer, L., Mörtsell, A., Nordebäck, T., Persson, S., and Sjöberg, F. (2002). SMART-1 mission description and development status. *Planetary and Space Science*, 50(14-15):1323 – 1337.

[Rayman et al., 2006] Rayman, M., Fraschetti, T., Raymond, C., and Russell, C. (2006). Dawn: A mission in development for exploration of main belt asteroids Vesta and Ceres. *Acta Astronautica*, 58(11):605 – 616.

[Rayman and Williams, 2002] Rayman, M. and Williams, S. (2002). Design of the first interplanetary solar electric propulsion mission. *Journal of Spacecraft and Rockets*, 39(4):589 – 595.

[Sauer Jr., 1997] Sauer Jr., C. (1997). Solar electric propulsion performance for Medlite and Delta class planetary missions. *Advances in the Astronautical Sciences*, 97, part 2:1951–1968.

[Storn and Price, 1997] Storn, R. and Price, K. (1997). Differential evolution a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11:341–359. 10.1023/A:1008202821328.

[Stuhlinger, 1964] Stuhlinger, E. (1964). *Ion propulsion for space flight*. McGraw-Hill Series in Missile and Space Technology. McGraw-Hill.

[Süli and Mayers, 2003] Süli, E. and Mayers, D. (2003). *An Introduction to Numerical Analysis*. Cambridge University Press.

[Vasile, 2006] Vasile, M. (2006). DITAN: A Tool for Optimal Space Trajectory Design. `http://trajectory.estec.esa.int/Astro/3rd-astro-workshop-presentations/DITAN-%20A%20Tool%20for%20Optimal%20Space%20Trajectory%20Design.pdf`. [Online; accessed 25-July-2011].

[Vasile et al., 2007] Vasile, M., De Pascale, P., and Casotto, S. (2007). On the optimality of a shape-based approach based on pseudo-equinoctial elements. *Acta Astronautica*, 61(1-6):286–297.

[Vinkó et al., 2007] Vinkó, T., Izzo, D., and Bombardelli, C. (2007). Benchmarking different global optimisation techniques for preliminary space trajectory design. In *International Astronautical Federation - 58th International Astronautical Congress 2007*, volume 6, pages 4181–4190.

[Wakker, 2007] Wakker, K. (2007). *Lecture Notes AE4874 - Astrodynamics*. Delft University of Technology, Delft, Netherlands.

[Walker et al., 1985] Walker, M. J. H., Ireland, B., and Owens, J. (1985). A set modified equinoctial orbit elements. *Celestial Mechanics and Dynamical Astronomy*, 36:409–419. 10.1007/BF01227493.

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

[Wall and Conway, 2009] Wall, B. and Conway, B. (2009). Shape-based approach to low-thrust rendezvous trajectory design. *Journal of Guidance, Control, and Dynamics*, 32(1):95–102.

[Wall and Novan, 2012] Wall, B. and Novan, D. (2012). A 3d shape-based approximation method for low-thrust trajectory design. In *Advances in the Astronautical Sciences Volume 142, AAS Paper 11-479*.

[Weise, 2009] Weise, T. (2009). *Global Optimization Algorithms - Theory and Application*. it-weise.de (self-published).

[Wertz and Larson, 1999] Wertz, J. and Larson, W. (1999). *Space mission analysis and design*. Space Technology Library. Microcosm.

[Wikipedia, 2004] Wikipedia (2004). Numerical ordinary differential equations — Wikipedia, The Free Encyclopedia. `http://en.wikipedia.org/wiki/Numerical_ordinary_differential_equations`. [Online; accessed 1-July-2011].

A. Orbital Mechanics

In this appendix velocity transformations and Gauss' planetary equations are given.

A.1. Velocity Transformations

In the following the transformations of velocities in the Cartesian, polar, cylindrical and spherical coordinate systems are given.

A.1.1. Cylindrical coordinates

The velocity components in cylindrical coordinates are V_r, V_θ and V_z, which are directed in the radial, normal and axial direction, respectively, see Figure 1(b). These velocities can be calculated using Cartesian coordinates and velocities as follows:

$$V_r = \dot{r} = \frac{x\dot{x} + y\dot{y}}{\sqrt{x^2 + y^2}} \tag{A.1}$$

$$V_\theta = r\dot{\theta} = \frac{x\dot{y} - y\dot{x}}{\sqrt{x^2 + y^2}} \tag{A.2}$$

$$V_z = \dot{z} \tag{A.3}$$

Vice versa, the Cartesian velocities can be expressed in cylindrical coordinates and velocities as follows:

$$\dot{x} = V_r \cos\theta - V_\theta \sin\theta \tag{A.4}$$

$$\dot{y} = V_r \sin\theta + V_\theta \cos\theta \tag{A.5}$$

$$\dot{z} = V_z \tag{A.6}$$

$$\tag{A.7}$$

A.1.2. Spherical coordinates

The velocity components in spherical coordinates are V_r and V_θ and V_φ, which are directed along the radial, azimuthal and latitudinal base vectors, respectively, see Figure 1(c). These velocities can be calculated using Cartesian coordinates and velocities as follows:

$$V_r = \dot{r} = \frac{x\dot{x} + y\dot{y} + z\dot{z}}{\sqrt{x^2 + y^2 + z^2}} \tag{A.8}$$

$$V_\theta = r\cos\varphi\dot{\theta} = \frac{x\dot{y} - y\dot{x}}{\sqrt{x^2 + y^2}} \tag{A.9}$$

$$V_\varphi = r\dot{\varphi} = \frac{r\dot{z} - z\dot{r}}{\sqrt{r^2 - z^2}} = \frac{(x^2 + y^2 + z^2)\dot{z} - z(x\dot{x} + y\dot{y} + z\dot{z})}{\sqrt{x^2 + y^2}\sqrt{x^2 + y^2 + z^2}} \tag{A.10}$$

Vice versa, the Cartesian velocities can be expressed in spherical coordinates and velocities as follows:

$$\dot{x} = V_r\cos\varphi\cos\theta - V_\theta\sin\theta - V_\varphi\sin\varphi\cos\theta \tag{A.11}$$

$$\dot{y} = V_r\cos\varphi\sin\theta + V_\theta\cos\theta - V_\varphi\sin\varphi\sin\theta \tag{A.12}$$

$$\dot{z} = V_r\sin\varphi + V_\varphi\cos\varphi \tag{A.13}$$

A.2. Gauss' Planetary Equations

Gauss' form of Lagrange's Planetary equations expresses the effect of a perturbing acceleration on the Keplerian osculating orbital elements as follows [Wakker, 2007]:

$$\begin{aligned}
\frac{da}{dt} &= 2\frac{a^2}{\sqrt{\mu p}}\left[f_r e\sin\theta + f_\theta\frac{p}{r}\right] \\
\frac{de}{dt} &= \sqrt{\frac{p}{\mu}}\left[f_r\sin\theta + f_\theta(\cos E + \cos\theta)\right] \\
\frac{di}{dt} &= f_h\frac{r}{\sqrt{\mu p}}\cos(\omega + \theta) \\
\frac{d\omega}{dt} &= -\sqrt{\frac{p}{\mu}}\left[f_h\frac{r}{p}\cot i\sin(\omega + \theta) + \frac{1}{e}\left\{f_r\cos\theta - f_\theta\left(1 + \frac{r}{p}\right)\sin\theta\right\}\right] \\
\frac{d\Omega}{dt} &= f_h\frac{r}{\sqrt{\mu p}\sin i}\sin(\omega + \theta) \\
\frac{dM}{dt} &= n - f_r\left[\frac{2r}{\sqrt{\mu a}} - \frac{1 - e^2}{e}\sqrt{\frac{a}{\mu}}\cos\theta\right] - f_\theta\frac{1 - e^2}{e}\sqrt{\frac{a}{\mu}}\left(1 + \frac{r}{p}\right)\sin\theta
\end{aligned} \tag{A.14}$$

where f_r, f_θ and f_h are the components of the perturbation acceleration in the radial, normal and out-of-plane (angular momentum) direction, respectively.

Gauss' planetary equations can also be written for the modified equinoctial elements as follows [Walker et al., 1985]:

$$
\begin{aligned}
\frac{dp}{dt} &= \frac{2p}{w}\sqrt{\frac{p}{\mu}}f_\theta \\
\frac{df}{dt} &= \sqrt{\frac{p}{\mu}}\left[f_r \sin L + [(w+1)\cos L + f]\,\frac{f_\theta}{w} - (h\sin L - k\cos L)\frac{g f_h}{w}\right] \\
\frac{dg}{dt} &= \sqrt{\frac{p}{\mu}}\left[-f_r \cos L + [(w+1)\sin L + g]\,\frac{f_\theta}{w} + (h\sin L - k\cos L)\frac{f f_h}{w}\right] \\
\frac{dh}{dt} &= \sqrt{\frac{p}{\mu}}\frac{s^2 f_h}{2w}\cos L \\
\frac{dk}{dt} &= \sqrt{\frac{p}{\mu}}\frac{s^2 f_h}{2w}\sin L \\
\frac{dL}{dt} &= \sqrt{\mu p}\left(\frac{w}{p}\right)^2 + \frac{1}{w}\sqrt{\frac{p}{\mu}}(h\sin L - k\cos L) f_h
\end{aligned}
\tag{A.15}
$$

where $w = 1 + f\cos L + g\sin L$ and $s^2 = 1 + h^2 + k^2$.

B. Optimization Search Space Assessment

In this appendix the range of the coefficient values of the optimal solutions are shown. These ranges are shown for the best 1% and best 10% results and for the overall results for different numbers of revolutions and different missions, for both the time-driven and polar-angle-driven method.

Table B.1: Minimum and maximum values of optimal free coefficients for a mission to Mars for the time-driven method with $V_r = C_1 + C_2 t + C_3 t^2 + C_4 t \sin(\pi t) + C_5 t \cos(\pi t)$, $V_\theta = C_6 + C_7 t + C_8 t^2 + C_9 t \sin(\pi t) + C_{10} t \cos(\pi t)$ and $V_z = C_{11} \cos((N+0.5)\pi t) + C_{12} t^3 \cos((N+0.5)\pi t) + C_{13} t^3 \sin((N+0.5)\pi t) + C_{14} t^4 \cos((N+0.5)\pi t) + C_{15} t^4 \sin((N+0.5)\pi t)$.

N	Results	Values	C_4	C_5	C_9	C_{10}	C_{14}	C_{15}
0	Best 1%	min	-1.9×10^4	5.5×10^3	-2.5×10^4	-3.5×10^4	-1.7×10^4	-1.0×10^4
		max	1.2×10^4	1.7×10^4	-1.3×10^3	1.5×10^4	9.6×10^3	9.3×10^3
	Best 10%	min	-5.8×10^4	-2.5×10^3	-4.3×10^4	-8.6×10^4	-1.7×10^4	-1.0×10^4
		max	1.2×10^4	1.7×10^4	-1.3×10^3	1.5×10^4	1.1×10^4	9.3×10^3
	Overall	min	-1.3×10^5	-1.8×10^5	-7.7×10^4	-9.3×10^4	-1.0×10^6	-1.0×10^6
		max	1.2×10^4	2.9×10^4	8.5×10^4	1.5×10^5	1.0×10^6	9.6×10^5
2	Best 1%	min	-3.9×10^3	-3.8×10^3	-6.3×10^3	-7.7×10^3	-1.2×10^3	-4.8×10^3
		max	2.3×10^3	8.6×10^3	4.5×10^3	4.2×10^3	2.1×10^3	-5.0×10^1
	Best 10%	min	-5.8×10^3	-8.2×10^3	-1.2×10^4	-1.7×10^4	-6.2×10^3	-5.6×10^3
		max	8.5×10^3	1.4×10^4	6.0×10^3	1.6×10^4	4.5×10^3	5.9×10^3
	Overall	min	-3.5×10^4	-4.7×10^4	-2.8×10^4	-5.4×10^4	-7.9×10^3	-1.0×10^4
		max	3.6×10^4	3.0×10^4	8.9×10^4	1.8×10^5	6.1×10^3	1.3×10^4
4	Best 1%	min	-2.9×10^3	-9.1×10^2	-5.9×10^3	-9.5×10^3	-4.3×10^3	-4.7×10^3
		max	6.0×10^2	4.1×10^3	6.1×10^3	2.3×10^3	-1.7×10^1	1.3×10^3
	Best 10%	min	-4.6×10^3	-9.3×10^3	-1.3×10^4	-1.9×10^4	-6.2×10^3	-5.9×10^3
		max	5.4×10^3	9.7×10^3	9.6×10^3	2.4×10^4	4.3×10^3	7.7×10^3
	Overall	min	-4.6×10^3	-5.2×10^4	-1.3×10^4	-1.9×10^4	-7.3×10^3	-1.4×10^4
		max	4.2×10^4	9.7×10^3	1.7×10^5	3.3×10^5	5.4×10^3	1.7×10^4

Table B.2: Minimum and maximum values of optimal free coefficients for a mission to 1989ML for the time-driven method with $V_r = C_1 + C_2 t + C_3 t^2 + C_4 t \sin(\pi t) + C_5 t \cos(\pi t)$, $V_\theta = C_6 + C_7 t + C_8 t^2 + C_9 t \sin(\pi t) + C_{10} t \cos(\pi t)$ and $V_z = C_{11} \cos(0.5\pi t) + C_{12} t \cos(0.5\pi t) + C_{13} t \sin(\pi t) + C_{14} t^6 \cos(0.5\pi t) + C_{15} t^6 \sin(0.5\pi t)$.

N	Results	Values	C_4	C_5	C_9	C_{10}	C_{14}	C_{15}
0	Best 1%	min	-8.6×10^2	-1.8×10^3	-1.0×10^3	-5.1×10^2	-3.7×10^3	-5.5×10^3
		max	1.0×10^3	9.6×10^2	2.3×10^3	1.2×10^3	2.7×10^3	5.7×10^3
	Best 10%	min	-1.3×10^4	-6.3×10^3	-6.6×10^3	-1.5×10^4	-5.5×10^3	-1.1×10^4
		max	1.1×10^4	6.8×10^3	7.6×10^3	1.3×10^4	5.8×10^3	1.2×10^4
	Overall	min	-9.5×10^4	-3.6×10^4	-3.2×10^4	-7.4×10^4	-6.2×10^4	-2.9×10^5
		max	2.8×10^4	4.1×10^4	3.8×10^4	2.6×10^4	7.6×10^4	2.3×10^5
2	Best 1%	min	-3.6×10^3	-3.1×10^3	-2.7×10^3	-3.2×10^3	-1.7×10^3	-1.1×10^4
		max	4.0×10^3	1.9×10^3	2.9×10^3	4.4×10^3	2.1×10^3	1.0×10^4
	Best 10%	min	-4.9×10^3	-4.9×10^3	-4.0×10^3	-4.9×10^3	-1.9×10^3	-1.4×10^4
		max	5.3×10^3	3.2×10^3	5.4×10^3	6.1×10^3	2.4×10^3	1.3×10^4
	Overall	min	-1.6×10^5	-3.2×10^5	-3.4×10^4	-1.5×10^5	-4.0×10^5	-8.0×10^5
		max	1.4×10^5	1.2×10^5	5.7×10^4	5.6×10^4	3.2×10^5	1.4×10^6

Table B.3: Minimum and maximum values of optimal free coefficients for a mission to Tempel 1 for the time-driven method with $V_r = C_1 + C_2 t + C_3 \sin(0.5\pi t) + C_4 t \sin(\pi t) + C_5 t \cos(\pi t)$, $V_\theta = C_6 + C_7 t + C_8 \sin(0.5\pi t) + C_9 t \sin(\pi t) + C_{10} t \cos(\pi t)$ and $V_z = C_{11} \cos(0.5\pi t) + C_{12} t^2 + C_{13} t^3 + C_{14} t^3 \cos(0.5\pi t) + C_{15} t^3 \sin(0.5\pi t)$.

N	Results	Values	C_4	C_5	C_9	C_{10}	C_{14}	C_{15}
0	Best 1%	min	-1.7×10^3	-6.9×10^3	-1.1×10^3	-3.7×10^3	-2.9×10^4	-1.3×10^4
		max	2.8×10^3	9.9×10^2	8.0×10^3	5.3×10^3	1.7×10^4	1.3×10^4
	Best 10%	min	-9.6×10^3	-1.2×10^4	-8.1×10^3	-9.4×10^3	-2.9×10^4	-1.3×10^4
		max	1.6×10^4	9.9×10^3	1.2×10^4	1.6×10^4	3.2×10^4	2.0×10^4
	Overall	min	-6.9×10^4	-5.1×10^4	-2.2×10^4	-3.7×10^4	-4.9×10^5	-1.3×10^5
		max	3.2×10^4	5.1×10^4	4.4×10^4	3.0×10^4	5.5×10^5	1.8×10^5
2	Best 1%	min	-1.4×10^3	-7.0×10^3	-6.2×10^3	-3.1×10^3	-1.6×10^4	-1.1×10^4
		max	1.0×10^4	3.1×10^3	5.6×10^3	1.0×10^4	3.0×10^4	1.6×10^4
	Best 10%	min	-7.1×10^3	-7.5×10^3	-7.1×10^3	-6.9×10^3	-4.7×10^4	-1.3×10^4
		max	1.0×10^4	5.6×10^3	7.8×10^3	1.0×10^4	4.2×10^4	2.0×10^4
	Overall	min	-8.9×10^4	-1.1×10^5	-5.0×10^4	-6.7×10^4	-1.7×10^5	-3.1×10^4
		max	2.1×10^4	4.1×10^4	5.1×10^4	4.2×10^4	1.0×10^6	4.1×10^5

Table B.4: Minimum and maximum values of optimal free coefficients for a mission to Mercury for the time-driven method with $V_r = C_1 + C_2 t + C_3 t^2 + C_4 t \sin(\pi t) + C_5 t \cos(\pi t)$, $V_\theta = C_6 + C_7 t + C_8 t^2 + C_9 t \sin(\pi t) + C_{10} t \cos(\pi t)$ and $V_z = C_{11} \cos(0.5\pi t) + C_{12} t^5 + C_{13} t^6 + C_{14} t^6 \cos(0.5\pi t) + C_{15} t^6 \sin(0.5\pi t)$.

N	Results	Values	C_4	C_5	C_9	C_{10}	C_{14}	C_{15}
0	Best 1%	min	-2.0×10^4	-5.5×10^4	-3.3×10^4	-4.7×10^4	-2.4×10^5	-1.3×10^5
		max	6.4×10^3	5.0×10^4	3.8×10^4	3.2×10^4	2.0×10^5	1.5×10^5
	Best 10%	min	-4.3×10^4	-6.6×10^4	-8.5×10^4	-1.4×10^5	-4.5×10^5	-3.2×10^5
		max	1.2×10^5	1.4×10^5	1.1×10^5	1.4×10^5	7.4×10^5	3.3×10^5
	Overall	min	-3.5×10^5	-5.3×10^5	-1.0×10^5	-1.7×10^5	-5.6×10^6	-6.3×10^6
		max	1.8×10^5	5.0×10^5	3.5×10^5	5.7×10^5	1.0×10^7	3.7×10^6
4	Best 1%	min	5.0×10^3	-7.1×10^3	-1.7×10^3	1.8×10^4	-1.1×10^2	-1.3×10^4
		max	1.2×10^4	1.2×10^3	1.6×10^4	4.8×10^4	6.7×10^3	-4.4×10^3
	Best 10%	min	-1.7×10^4	-2.6×10^4	-7.2×10^3	-5.8×10^4	-9.2×10^3	-2.2×10^4
		max	2.4×10^4	5.3×10^3	4.2×10^4	9.8×10^4	2.2×10^4	-1.5×10^2
	Overall	min	-4.2×10^5	-6.2×10^5	-2.7×10^5	-4.0×10^5	-3.3×10^6	-4.0×10^6
		max	2.2×10^5	1.6×10^5	5.3×10^5	9.4×10^5	2.9×10^6	4.5×10^6

Table B.5: Minimum and maximum values of optimal free coefficients for a mission to Mars for the polar-angle-driven method with $R = C_1 + C_2\theta + C_3\theta^2 + C_4\theta^3 + C_5\theta^4$, $T = C_6 + C_7\theta + C_8\theta^2 + C_9\theta^3 + C_{10}\theta^4$ and $Z = C_{11}\cos\theta + C_{12}\theta\cos\theta + C_{13}\theta\sin\theta + C_{14}\theta^2\cos\theta + C_{15}\theta^2\sin\theta$.

N	Results	Values	C_4	C_5	C_9	C_{10}	C_{14}	C_{15}
0	Best 1%	min	-9.2×10^{11}	-2.2×10^9	-1.5×10^8	-6.0×10^7	-1.9×10^{10}	-1.3×10^{10}
		max	5.3×10^{10}	4.5×10^{11}	1.0×10^8	6.3×10^7	5.9×10^9	7.4×10^9
	Best 10%	min	-9.2×10^{11}	-3.3×10^{11}	-6.5×10^8	-6.0×10^7	-6.9×10^{10}	-3.0×10^{10}
		max	2.7×10^{12}	7.2×10^{11}	1.0×10^8	3.2×10^8	7.6×10^{10}	3.1×10^{10}
	Overall	min	-5.6×10^{16}	-4.8×10^{16}	-5.7×10^{12}	-4.7×10^{11}	-3.1×10^{15}	-2.9×10^{15}
		max	9.1×10^{16}	3.5×10^{16}	4.3×10^{11}	2.8×10^{12}	3.4×10^{15}	3.3×10^{15}
2	Best 1%	min	-5.4×10^{10}	-4.8×10^{10}	-6.9×10^7	6.3×10^6	-7.9×10^9	-8.2×10^9
		max	1.9×10^{11}	6.0×10^{10}	-2.4×10^6	4.1×10^7	7.4×10^9	7.5×10^9
	Best 10%	min	-4.2×10^{11}	-9.0×10^{10}	-1.4×10^8	-4.7×10^7	-9.8×10^9	-1.1×10^{10}
		max	3.7×10^{11}	1.5×10^{11}	6.8×10^7	7.5×10^7	9.0×10^9	1.1×10^{10}
	Overall	min	-3.5×10^{12}	-1.1×10^{12}	-5.9×10^8	-1.2×10^8	-1.4×10^{11}	-1.1×10^{11}
		max	1.5×10^{12}	1.5×10^{12}	2.1×10^8	3.0×10^8	1.2×10^{11}	3.8×10^{11}
4	Best 1%	min	-2.0×10^{10}	-4.6×10^{10}	-8.4×10^7	6.8×10^6	-7.4×10^9	-3.7×10^9
		max	2.1×10^{11}	3.1×10^{10}	-8.0×10^5	5.2×10^7	8.7×10^9	1.0×10^{10}
	Best 10%	min	-4.6×10^{11}	-9.0×10^{10}	-1.4×10^8	-8.1×10^7	-9.5×10^9	-1.1×10^{10}
		max	2.9×10^{11}	1.3×10^{11}	1.3×10^8	7.9×10^7	9.9×10^9	1.1×10^{10}
	Overall	min	-1.8×10^{13}	-3.1×10^{12}	-1.4×10^8	-1.2×10^{11}	-5.9×10^{13}	-2.8×10^{13}
		max	5.9×10^{12}	9.9×10^{12}	2.1×10^{11}	7.9×10^7	2.0×10^{13}	8.8×10^{12}

Table B.6: Minimum and maximum values of optimal free coefficients for a mission to 1989ML for the polar-angle-driven method with $R = C_1 + C_2\theta + C_3\theta^2 + C_4\theta^3 + C_5\theta^4$, $T = C_6 + C_7\theta + C_8\theta^2 + C_9\theta^3 + C_{10}\theta^4$ and $Z = C_{11}\cos\theta + C_{12}\theta\cos\theta + C_{13}\theta\sin\theta + C_{14}\theta^2\cos\theta + C_{15}\theta^2\sin\theta$.

N	Results	Values	C_4	C_5	C_9	C_{10}	C_{14}	C_{15}
0	Best 1%	min	-5.4×10^11	-9.8×10^10	-1.4×10^7	-2.2×10^7	-2.0×10^10	-4.6×10^10
		max	1.5×10^11	2.4×10^11	3.1×10^7	8.7×10^6	3.0×10^10	4.5×10^10
	Best 10%	min	-7.9×10^11	-1.0×10^12	-3.7×10^8	-9.0×10^7	-3.0×10^11	-2.9×10^11
		max	1.8×10^12	5.0×10^11	1.7×10^8	1.7×10^8	1.3×10^11	2.0×10^11
	Overall	min	-3.0×10^15	-1.8×10^13	-2.3×10^12	-3.3×10^8	-1.3×10^18	-4.8×10^15
		max	9.4×10^13	1.5×10^15	4.5×10^8	1.1×10^12	2.9×10^18	4.9×10^16
2	Best 1%	min	-2.4×10^11	-3.8×10^10	-9.0×10^7	-1.3×10^7	-1.7×10^10	-1.5×10^10
		max	2.3×10^11	1.2×10^11	2.1×10^7	4.9×10^7	9.0×10^9	1.6×10^10
	Best 10%	min	-5.7×10^11	-1.9×10^11	-1.2×10^8	-8.2×10^7	-1.9×10^10	-1.9×10^10
		max	4.8×10^11	1.2×10^11	1.4×10^8	6.7×10^7	2.0×10^10	2.1×10^10
	Overall	min	-4.2×10^13	-1.9×10^13	-3.5×10^8	-1.4×10^10	-4.5×10^12	-7.7×10^12
		max	3.6×10^13	2.1×10^13	2.5×10^10	3.6×10^8	2.6×10^13	2.9×10^13

Table B.7: Minimum and maximum values of optimal free coefficients for a mission to Tempel 1 for the polar-angle-driven method with $R = C_1 + C_2\theta + C_3\theta^2 + C_4\theta^3 + C_5\theta^4$, $T = C_6 + C_7\theta + C_8\theta^2 + C_9\theta^3 + C_{10}\theta^4$ and $Z = C_{11}\cos\theta + C_{12}\theta\cos\theta + C_{13}\theta\sin\theta + C_{14}\theta^2\cos\theta + C_{15}\theta^2\sin\theta$.

N	Results	Values	C_4	C_5	C_9	C_{10}	C_{14}	C_{15}
0	Best 1%	min	-1.3×10^12	-1.8×10^12	-1.4×10^8	-1.7×10^7	-8.9×10^10	-7.6×10^11
		max	2.3×10^12	8.3×10^11	6.1×10^7	1.2×10^8	9.8×10^11	5.3×10^10
	Best 10%	min	-4.0×10^12	-3.1×10^12	-4.0×10^8	-3.3×10^8	-6.4×10^11	-2.0×10^12
		max	4.0×10^12	1.9×10^12	4.3×10^8	2.4×10^8	1.2×10^12	3.5×10^11
	Overall	min	-4.1×10^15	-1.6×10^15	-2.6×10^12	-4.8×10^12	-4.5×10^15	-3.7×10^15
		max	1.4×10^15	2.4×10^15	1.1×10^12	1.3×10^12	7.6×10^16	1.6×10^16
2	Best 1%	min	-8.6×10^11	1.4×10^11	-2.6×10^8	-6.2×10^7	-1.0×10^11	-1.1×10^11
		max	1.6×10^11	3.9×10^11	7.8×10^7	1.4×10^8	5.6×10^10	1.2×10^11
	Best 10%	min	-2.0×10^12	-2.4×10^11	-6.7×10^8	-2.7×10^8	-2.8×10^11	-2.2×10^11
		max	2.2×10^12	5.3×10^11	4.2×10^8	4.2×10^8	1.4×10^11	1.8×10^11
	Overall	min	-2.9×10^13	-4.5×10^14	-1.7×10^9	-4.3×10^11	-8.3×10^14	-4.8×10^14
		max	8.0×10^14	1.4×10^13	7.2×10^11	9.7×10^8	6.6×10^14	6.8×10^14

Table B.8: Minimum and maximum values of optimal free coefficients for a mission to Mercury for the polar-angle-driven method with $R = C_1 + C_2\theta + C_3\theta^2 + C_4\theta^3 + C_5\theta^4$, $T = C_6 + C_7\theta + C_8\theta^2 + C_9\theta^3 + C_{10}\theta^4$ and $Z = C_{11}\cos\theta + C_{12}\theta\cos\theta + C_{13}\theta\sin\theta + C_{14}\theta^2\cos\theta + C_{15}\theta^2\sin\theta$.

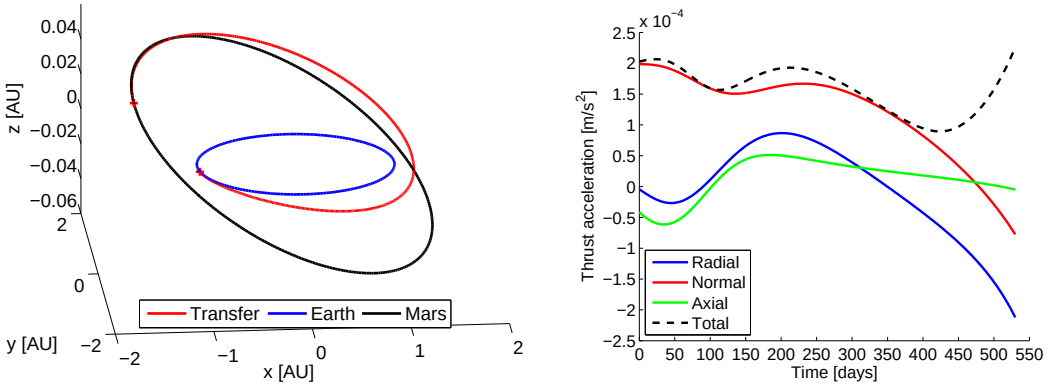
N	Results	Values	C_4	C_5	C_9	C_{10}	C_{14}	C_{15}
0	Best 1%	min	-7.1×10^11	-5.8×10^11	-4.9×10^7	-3.2×10^7	-7.7×10^10	-6.4×10^10
		max	1.4×10^12	1.8×10^11	7.4×10^7	2.9×10^7	6.2×10^10	6.0×10^10
	Best 10%	min	-2.4×10^12	-2.9×10^12	-6.5×10^8	-3.8×10^7	-3.9×10^11	-3.8×10^11
		max	7.6×10^12	8.7×10^11	9.1×10^7	3.2×10^8	2.4×10^11	1.9×10^11
	Overall	min	-3.2×10^13	-8.4×10^15	-8.0×10^12	-2.9×10^8	-1.1×10^15	-5.7×10^15
		max	6.1×10^15	2.4×10^12	1.6×10^9	1.0×10^13	1.9×10^15	1.7×10^15
4	Best 1%	min	-1.8×10^10	-7.2×10^10	-1.8×10^7	-3.8×10^7	-6.8×10^9	-8.2×10^9
		max	5.5×10^10	-2.0×10^9	7.6×10^7	6.0×10^6	7.8×10^9	5.3×10^9
	Best 10%	min	-7.1×10^10	-1.3×10^11	-8.2×10^7	-5.1×10^7	-7.2×10^9	-8.3×10^9
		max	1.3×10^11	7.2×10^10	1.0×10^8	3.9×10^7	8.7×10^9	7.9×10^9
	Overall	min	-6.0×10^13	-4.2×10^13	-4.8×10^8	-1.7×10^10	-2.6×10^13	-3.5×10^13
		max	8.4×10^13	3.0×10^13	4.0×10^10	2.1×10^9	3.2×10^13	2.1×10^13

C. Results

In this appendix results of the hodographic-shaping methods are shown. These results include Pareto front plots, ΔV plots, and trajectory and thrust profile plots corresponding to the best lowest- and higher-order solutions.

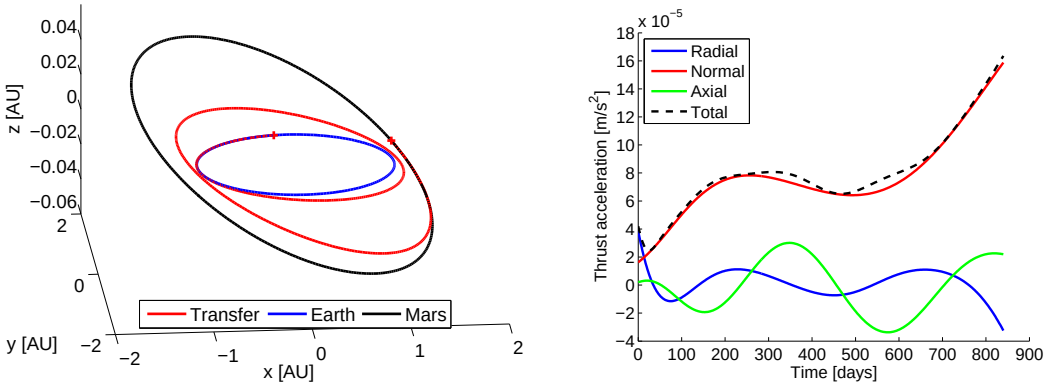
- **Mars**
  - Shaping as function of polar angle
    - Trajectories and thrust profiles: Figures C.1 and C.2
- **1989ML**
  - Shaping as function of polar angle
    - Pareto fronts different axial power orders: Figure C.3
    - Pareto fronts different number of revolutions: Figure C.4
    - ΔV plots: Figures C.5 and C.6
    - Trajectories and thrust profiles: Figures C.7 and C.8
- **Tempel 1**
  - Shaping as function of time
    - Pareto fronts different axial power orders: Figure C.9
    - Pareto fronts different number of revolutions: Figure C.10
    - ΔV plots: Figures C.11 and C.12
  - Shaping as function of polar angle
    - Pareto fronts lowest-order solutions: Figure C.13
    - Pareto fronts different axial power orders: Figure C.14
    - Pareto fronts different number of revolutions: Figure C.15
    - ΔV plots: Figures C.16 and C.17
    - Trajectories and thrust profiles: Figures C.18 and C.19
- **Mercury**
  - Shaping as function of time
    - Pareto fronts different axial power orders: Figure C.20
    - ΔV plots: Figures C.21 and C.22
  - Shaping as function of polar angle
    - Pareto fronts lowest-order solutions: Figure C.23
    - Pareto fronts different axial power orders: Figure C.24
    - ΔV plots: Figures C.25 and C.26
    - Trajectories and thrust profiles: Figures C.27 and C.28

Mars



(a) Trajectory (b) Thrust profile

Figure C.1: Trajectory and thrust profile of best lowest-order polar-angle-driven solution for Earth-Mars: R = CPow2Pow4, T = CPow3Pow5 and Z = CosPCosPSin, see Table 11.3.



(a) Trajectory (b) Thrust profile

Figure C.2: Trajectory and thrust profile of best higher-order polar-angle-driven solution for Earth-Mars: R = CPowPow2Pow3Pow4, T = CPowPow2Pow3Pow4 and Z = CosPCosPSinP2CosP2Sin, see Table 11.4.

1989ML

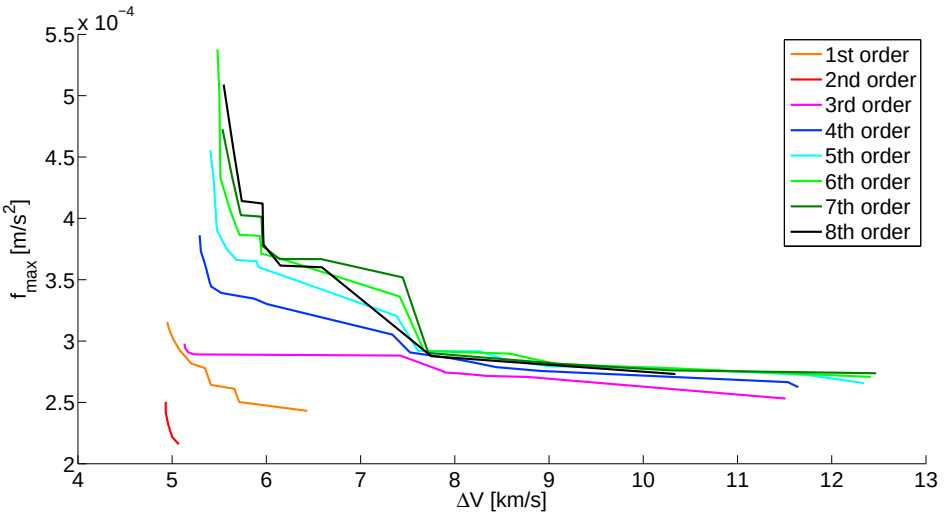


Figure C.3: Pareto fronts of the lowest-order polar-angle-driven solution CPow2Pow4-CPow3Pow5-CosPXCosPXSin for different axial power orders X for a mission to 1989ML.

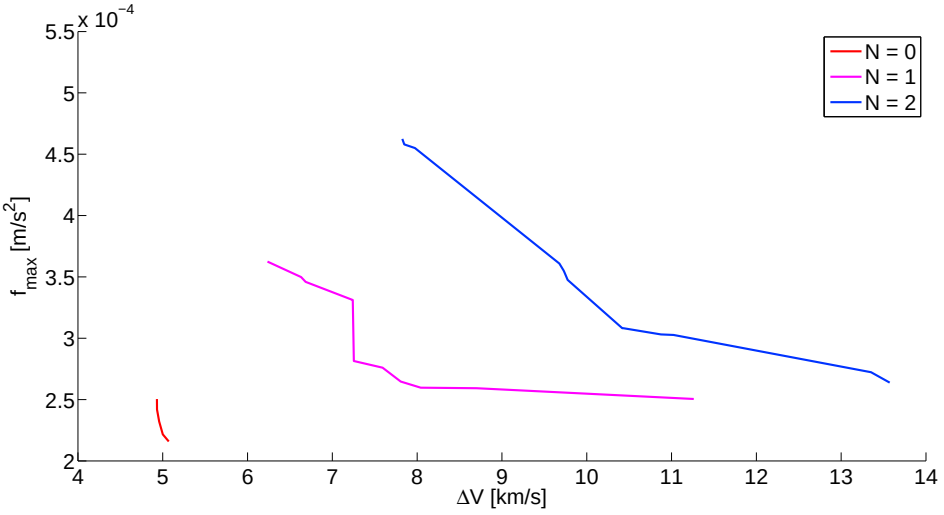


Figure C.4: Pareto fronts of the lowest-order polar-angle-driven solution CPow2Pow4-CPow3Pow5-CosPCosPSin for different number of revolutions N for a mission to 1989ML.

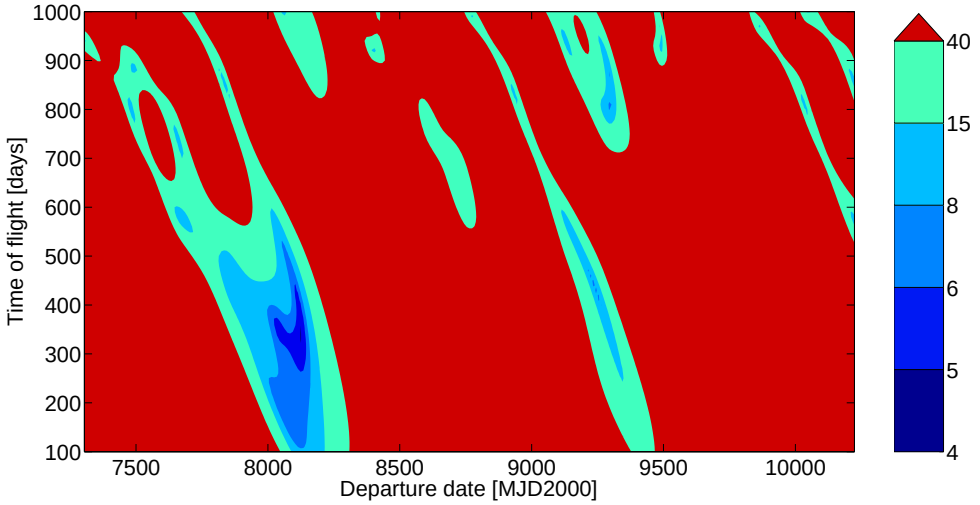


Figure C.5: ΔV plot for the lowest-order polar-angle-driven solution CPow2Pow4-CPow3Pow5-CosPCosPSin for a mission to asteroid 1989ML with $N = 0 - 3$.

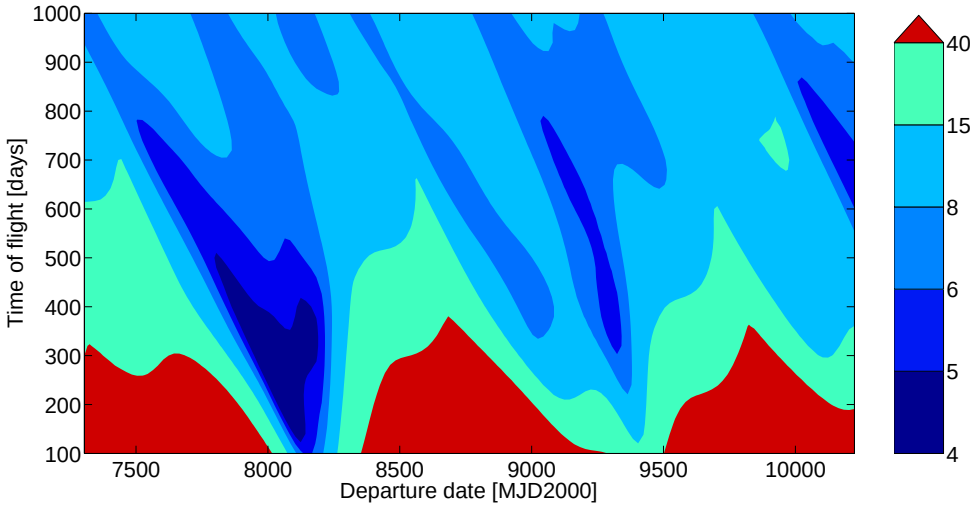
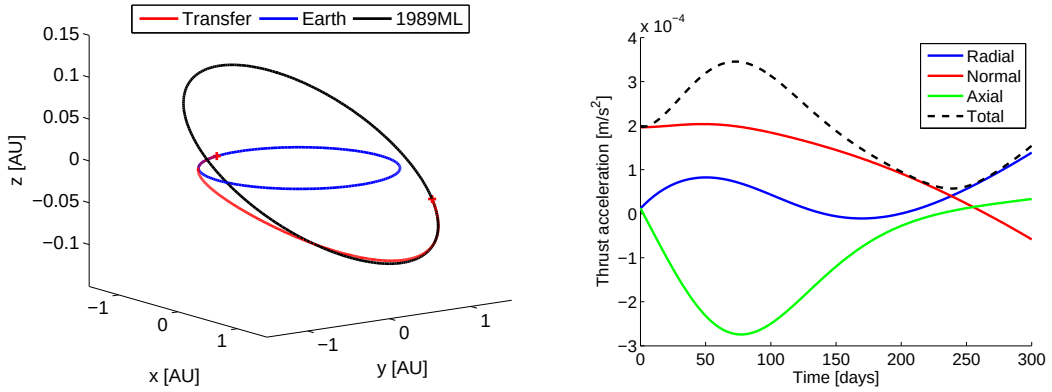
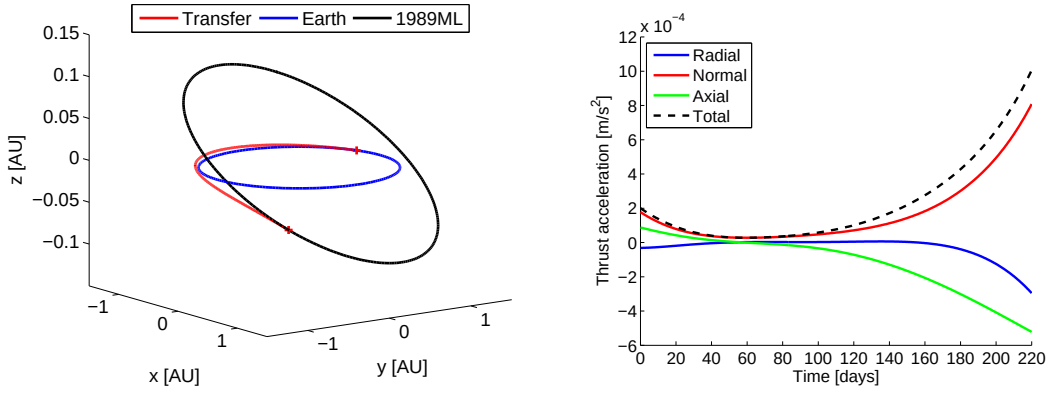


Figure C.6: ΔV plot for the higher-order polar-angle-driven solution CPowPow2Pow3Pow4-CPowPow2Pow3Pow4-CosPCosPSinP2CosP2Sin for a mission to asteroid 1989ML with $N = 0 - 2$.



(a) Trajectory (b) Thrust profile

Figure C.7: Trajectory and thrust profile of best lowest-order polar-angle-driven solution for Earth-1989ML: $R =$ CPow2Pow3, $T =$ CPow3Pow4 and $Z =$ CosPCosPSin, see Table 11.8.



(a) Trajectory (b) Thrust profile

Figure C.8: Trajectory and thrust profile of best higher-order polar-angle-driven solution for Earth-1989ML: $R =$ CPowPow2Pow3Pow4, $T =$ CPowPow2Pow3Pow4 and $Z =$ CosPCosPSinP2CosP2Sin, see Table 11.9.

Tempel 1

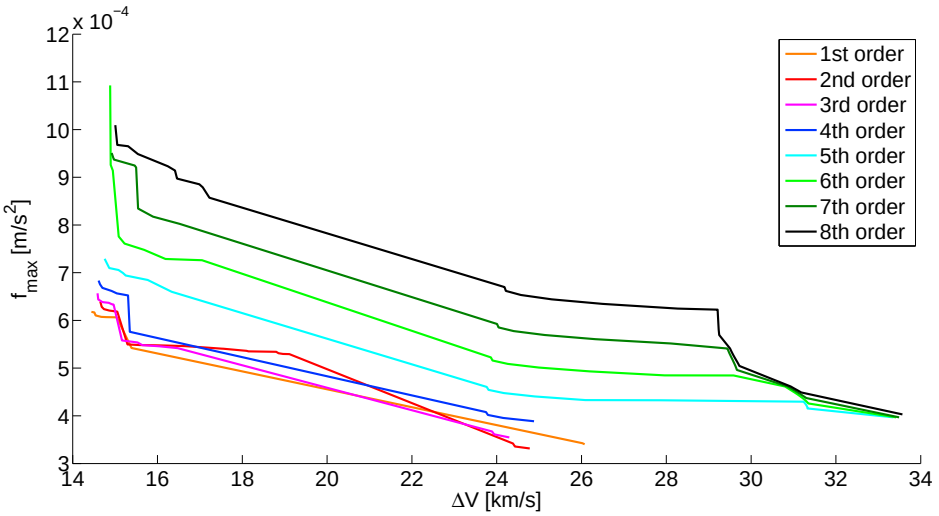


Figure C.9: Pareto fronts of the lowest-order time-driven solution CPowPow2-CPowPow2-CosR5PXCosR5PXSinR5 for different axial power orders X for a mission to Tempel1.

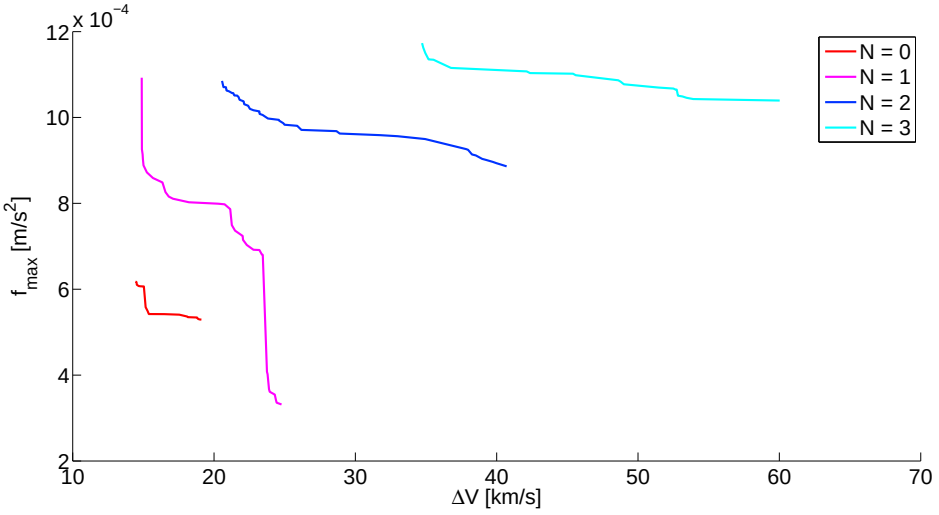


Figure C.10: Pareto fronts of the lowest-order time-driven solution CPowPow2-CPowPow2-CosR5PCosR5PSinR5 for different number of revolutions N for a mission to Tempel1.

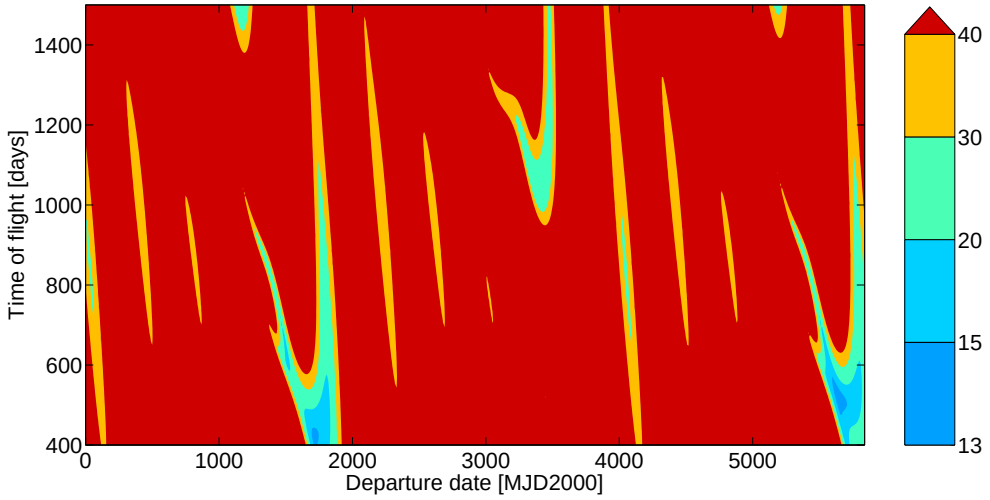


Figure C.11: ΔV plot for the lowest-order time-driven solution CPowPow2-CPowPow2-CosR5PCos05PSin05 for a mission to comet Tempel 1 with $N = 0 - 3$.

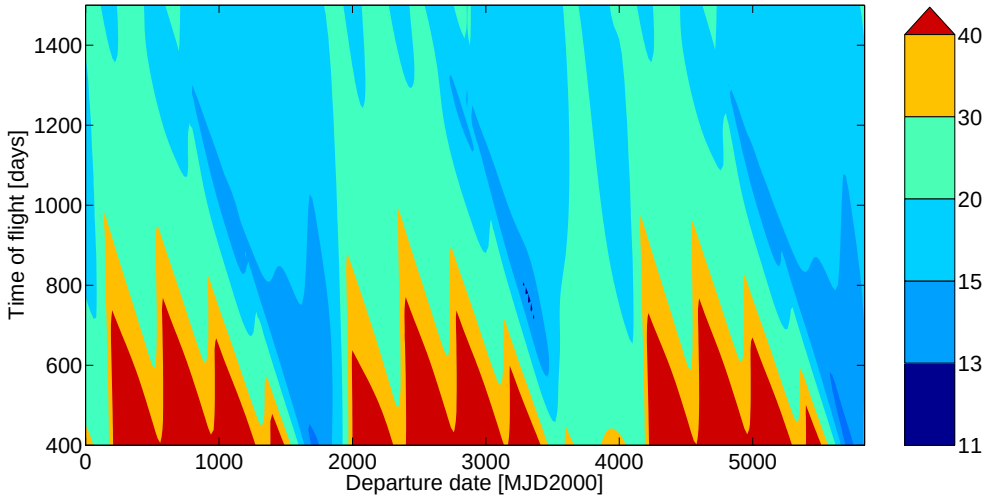


Figure C.12: ΔV plot for the higher-order time-driven solution CPowPow2PSin05PCos05-CPowPow2PSin05PCos05-Cos05PCos05PSin05P2Cos05P2Sin05 for a mission to comet Tempel 1 with $N = 0 - 3$.

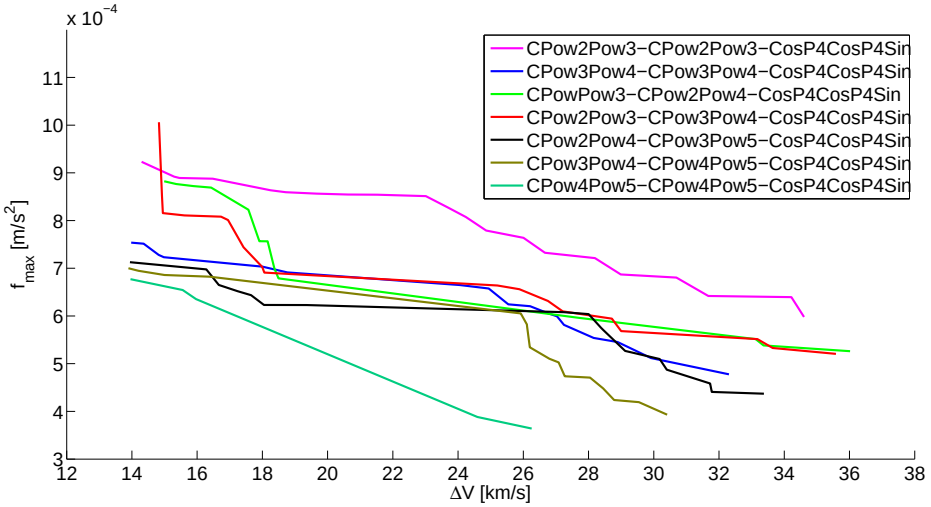


Figure C.13: Pareto fronts of lowest-order polar-angle-driven solutions for different radial-velocity and time-evolution functions for a mission to Tempel1.

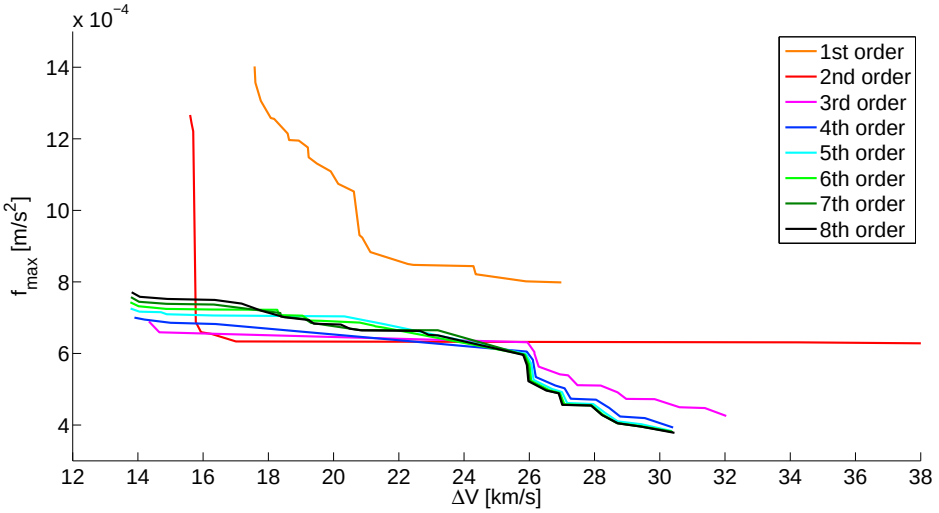


Figure C.14: Pareto fronts of the lowest-order polar-angle-driven solution CPow3Pow4-CPow4Pow5-CosPXCosPXSin for different axial power orders X for a mission to comet Tempel 1.

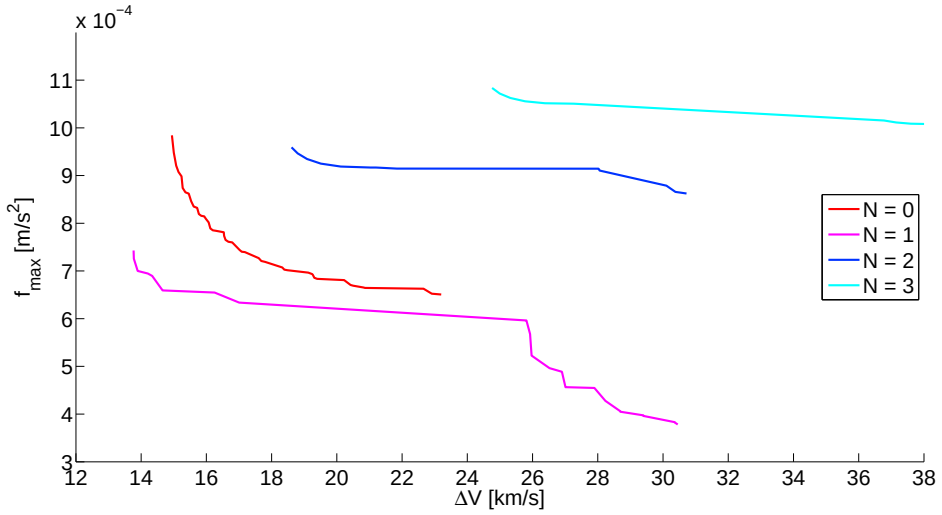


Figure C.15: Pareto fronts of the lowest-order polar-angle-driven solution CPow3Pow4-CPow4Pow5-CosPCosPSin for different number of revolutions N for a mission to comet Tempel 1.

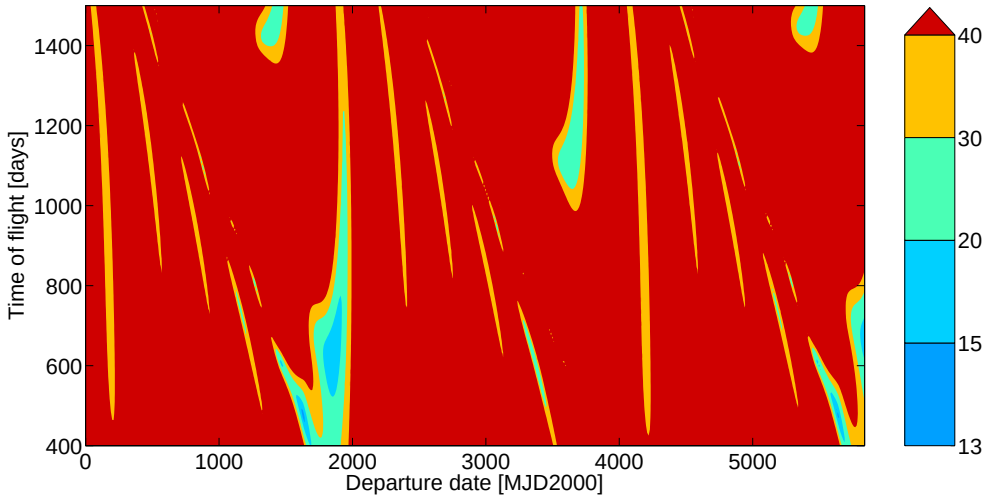


Figure C.16: ΔV plot for the lowest-order polar-angle-driven solution CPow2Pow4-CPow3Pow5-CosP4CosP4Sin for a mission to comet Tempel 1 with $N = 0 - 3$.

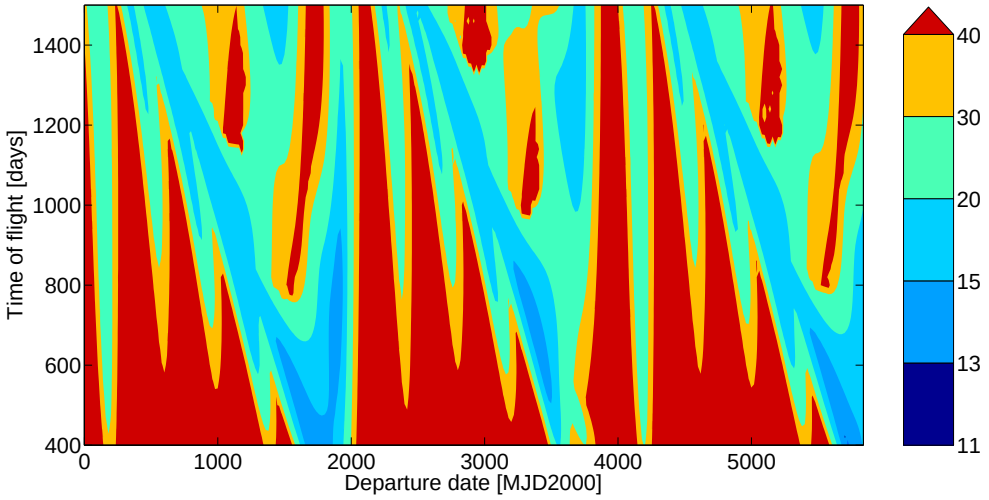
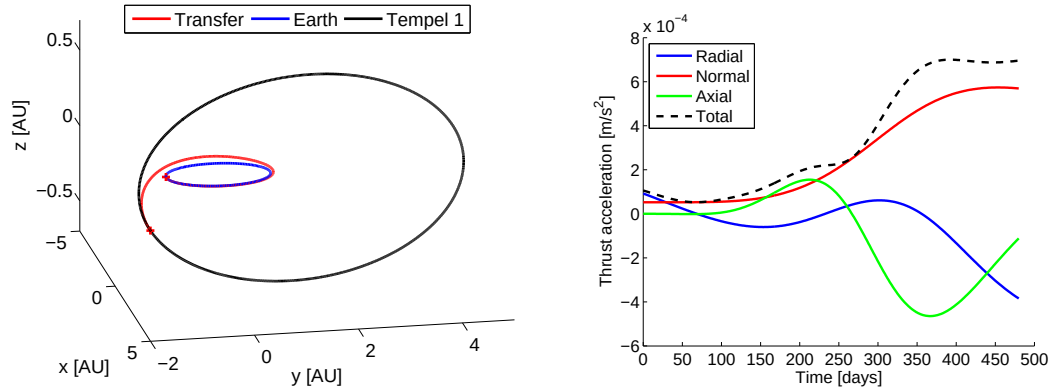
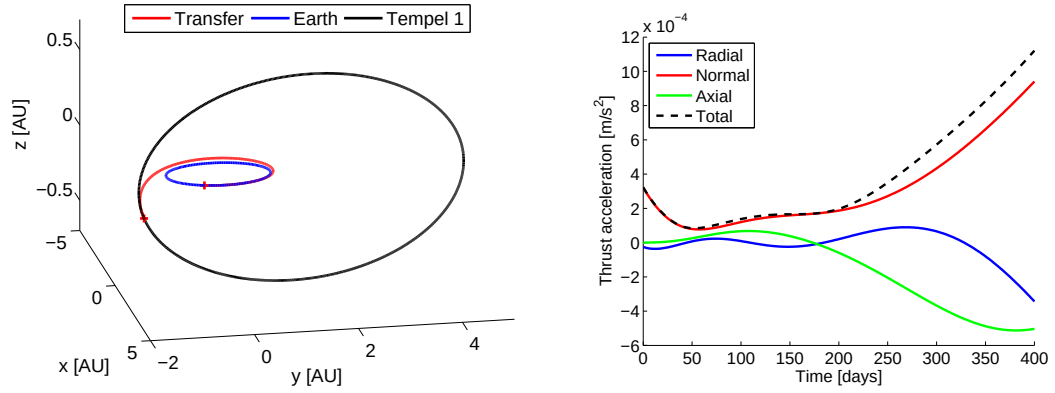


Figure C.17: ΔV plot for the higher-order polar-angle-driven solution CPowPow2Pow3Pow4-CPowPow2Pow3Pow4-CosP3CosP3SinP4CosP4Sin for a mission to comet Tempel 1 with $N = 0-3$.



(a) Trajectory (b) Thrust profile

Figure C.18: Trajectory and thrust profile of best lowest-order polar-angle-driven solution for Earth-Tempel1: $R =$ CPow2Pow3, $T =$ CPow3Pow4 and $Z =$ CosPCosPSin, see Table 11.13.



(a) Trajectory (b) Thrust profile

Figure C.19: Trajectory and thrust profile of best higher-order polar-angle-driven solution for Earth-Tempel1: $R =$ CPowPow2Pow3Pow4, $T =$ CPowPow2Pow3Pow4 and $Z =$ CosP3CosP3SinP4CosP4Sin, see Table 11.14.

Mercury

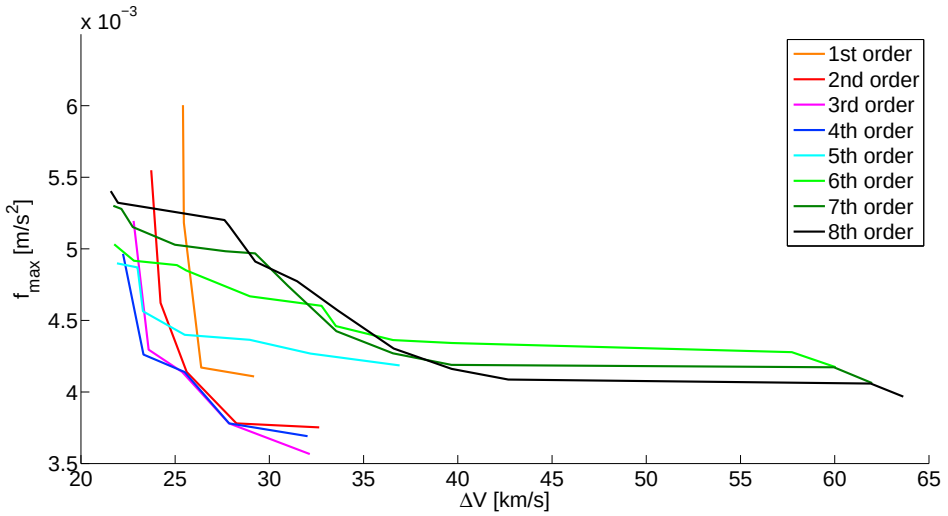


Figure C.20: Pareto fronts of the lowest-order time-driven solution CPowPow2-CPowPow2-CosR5PXCosR5PXSinR5 for different axial power orders X for a mission to Mercury.

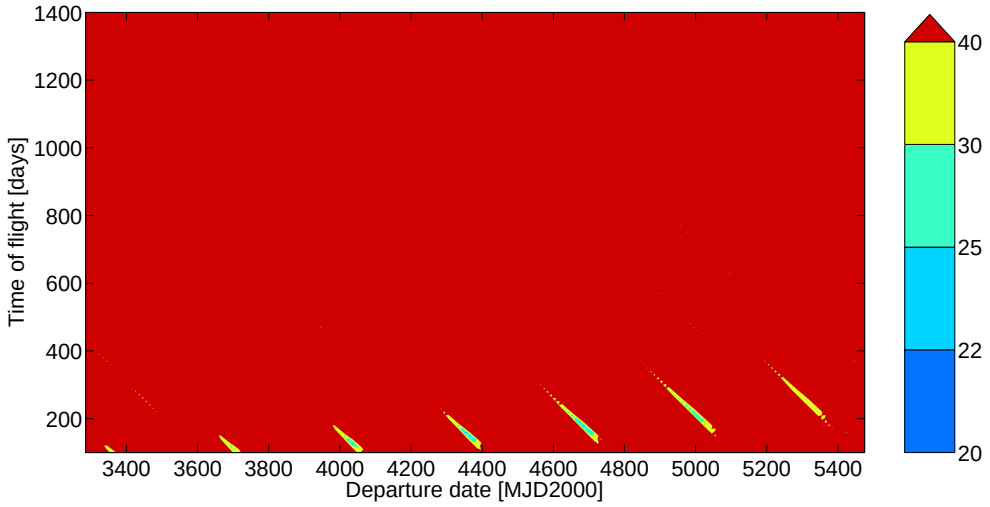


Figure C.21: ΔV plot for the lowest-order time-driven solution CPowPow2-CPowPow2-CosR5P6CosR5P6SinR5 for a mission to Mercury with $N = 0 - 6$.

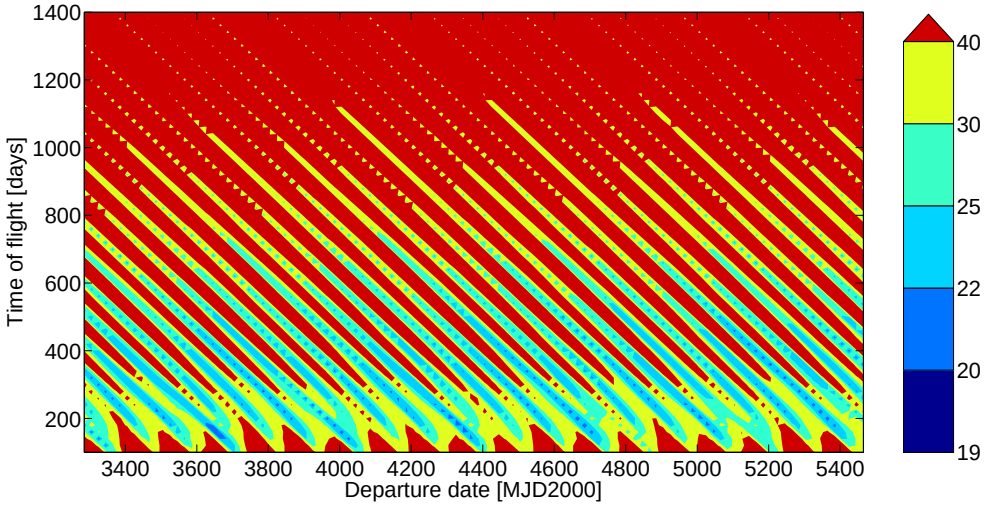


Figure C.22: ΔV plot for the higher-order time-driven solution CPowPow2PSin05PCos05-CPowPow2PSin05PCos05-CosR5P5CosR5P5SinR5P6CosR5P6SinR5 for a mission to Mercury with $N = 0 - 6$.

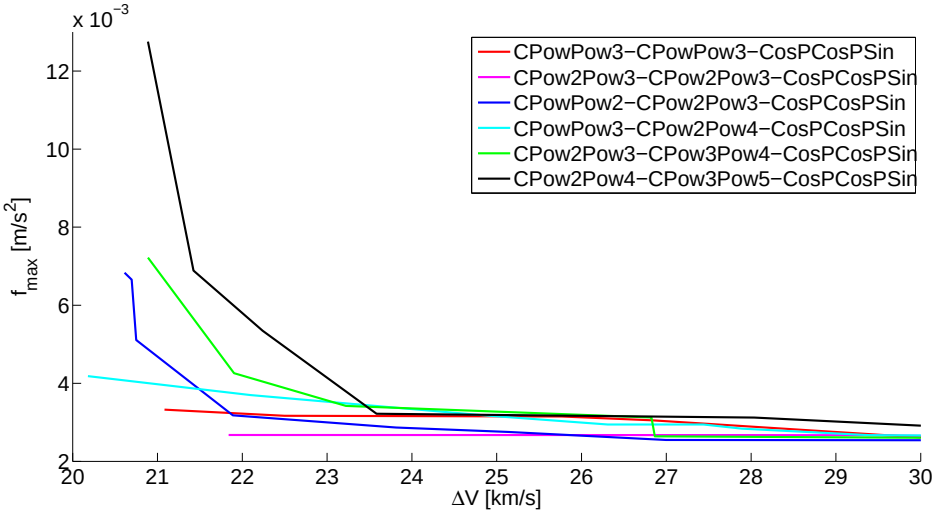


Figure C.23: Pareto fronts of lowest-order polar-angle-driven solutions with different radial-velocity and time-evolution functions for a mission to Mercury.

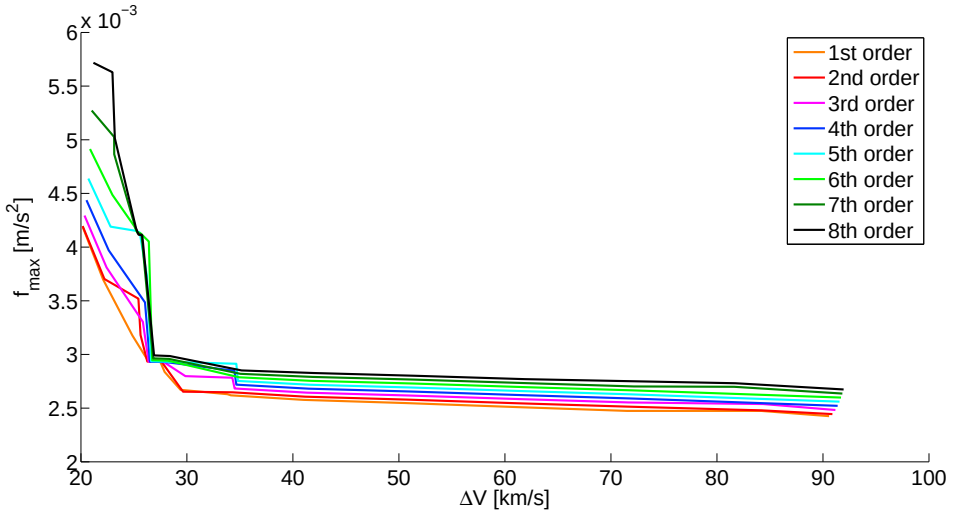


Figure C.24: Pareto fronts of the lowest-order polar-angle-driven solution CPowPow3-CPow2Pow4-CosPXCosPXSin for different axial power orders X for a mission to Mercury.

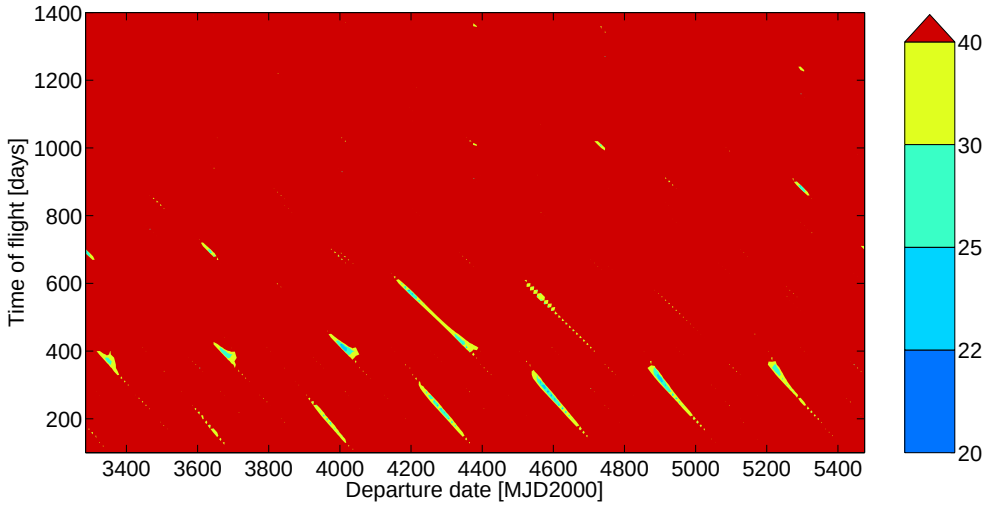


Figure C.25: ΔV plot for the lowest-order polar-angle-driven solution CPowPow3-CPow2Pow4-CosPCosPSin for a mission to Mercury with $N = 0 - 6$.

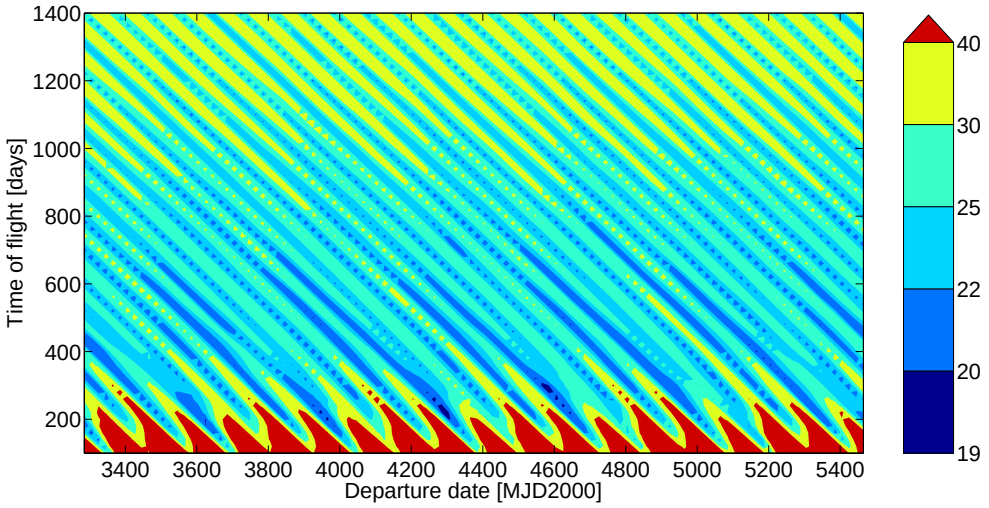
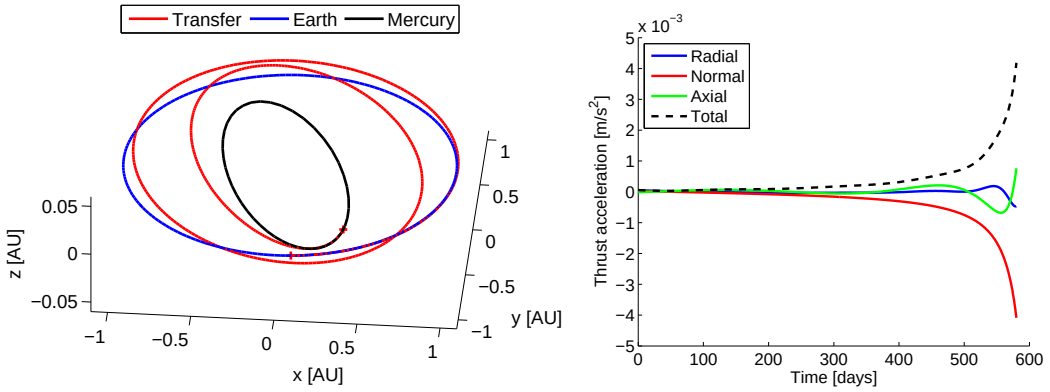
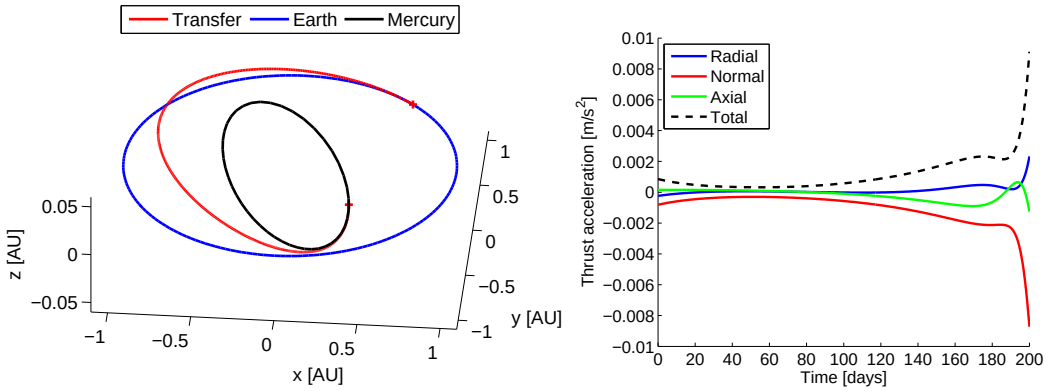


Figure C.26: ΔV plot for the higher-order polar-angle-driven solution CPowPow2Pow3Pow4-CPowPow2Pow3Pow4-CosPCosPSinP2CosP2Sin for a mission to Mercury with $N = 0 - 6$.



(a) Trajectory (b) Thrust profile

Figure C.27: Trajectory and thrust profile of best lowest-order polar-angle-driven solution for Earth-Mercury: R = CPowPow3, T = CPow2Pow4 and Z = CosPCosPSin, see Table 11.18.



(a) Trajectory (b) Thrust profile

Figure C.28: Trajectory and thrust profile of best higher-order polar-angle-driven solution for Earth-Mercury: R = CPowPow2Pow3Pow4, T = CPowPow2Pow3Pow4 and Z = CosP-CosPSinP2CosP2Sin, see Table 11.19.