Optimal Low-Thrust Orbit Transfers Made Easy: A Direct Approach

(Article begins on next page)

01 May 2024

Optimal Low-Thrust Orbit Transfers Made Easy: A Direct Approach

Mirko Leomanni[*], Gianni Bianchini[†], Andrea Garulli [‡], Renato Quartullo [§]

Università di Siena, Siena, 53100, Italy

The optimization of low-thrust, multi-revolution orbit transfer trajectories is often regarded as a difficult problem in modern astrodynamics. In this paper, a flexible and computationally efficient approach is presented for the optimization of low-thrust orbit transfers under eclipse constraints. The proposed approach leverages a new dynamic model of the orbital motion and a Lyapunov-based initial guess generation scheme that is very easy to tune. A multi-objective, single-phase formulation of the optimal control problem is devised, which provides a convenient way to trade off fuel consumption and time of flight. A distinctive feature of such a formulation is that it requires no prior information about the structure of the optimal solution. Simulation results for two benchmark orbit transfer scenarios indicate that minimum-time, minimum-fuel and mixed time/fuel-optimal instances of the control problem can be readily solved via direct collocation, while incurring a significantly lower computational demand with respect to existing techniques.

Nomenclature

$a, e, i, \omega, \Omega, v$	=	Classical orbital elements
p, f, g, h, k, L	=	Modified equinoctial elements
$\rho, e_x, e_y, h_x, h_y, \sigma$	=	Ideal elements
$\mathbf{e}$	=	Relative eccentricity vector
g_0	=	Standard gravity, m/s^2
I_{sp}	=	Specific impulse, s
m	=	Spacecraft mass, kg
$\mathbf{q}$	=	Thrust unit vector
R_e	=	Earth radius, m
R_s	=	Sun radius, m
$\mathbf{r}_{es}$	=	Satellite-Earth vector, m
$\mathbf{r}_s$	=	Earth-Sun vector, m
$\mathbf{r}_{ss}$	=	Satellite-Sun vector, m
s	=	Ideal anomaly, rad
T_{max}	=	Maximum deliverable thrust, N
t	=	Time, s
$\mathbf{u}$	=	Control acceleration, m/s^2
$\mathbf{x}$	=	System state
$\overline{\mathbf{z}}$	=	Target equinoctial elements
α	=	Time/fuel trade-off parameter
γ	=	Relative inclination, rad
ϵ	=	Mesh error tolerance
Δm	=	Mass variation, kg
Δt	=	Time of flight, days
λ_1, λ_2	=	Nodal elements, rad
μ	=	Earth gravitational parameter, m^3/s^2

[*]Research Associate, Department of Information Engineering and Mathematics, leomanni@diism.unisi.it

[†]Professor, Department of Information Engineering and Mathematics, giannibi@diism.unisi.it

[‡]Professor, Department of Information Engineering and Mathematics, garulli@diism.unisi.it

[§]PhD Student, Department of Information Engineering and Mathematics, quartullo@diism.unisi.it

ξ	=	Efficiency factor
ξ_{cut}	=	Cut-off parameter
ϱ_R	=	Radial perturbation, m/s^2
ϱ_T	=	Transverse perturbation, m/s^2
ϱ_N	=	Normal perturbation, m/s^2
ϱ	=	Perturbing acceleration vector, m/s^2
ϱ_{J2}	=	J2 acceleration vector, m/s^2
ζ	=	Throttle command
η	=	Throttle control input
θ_e	=	Angular size of Earth radius, rad
θ_{es}	=	Earth-Sun angle, rad
θ_s	=	Angular size of Sun radius, rad
$\varphi_{az}, \varphi_{el}$	=	Tuning parameters, rad
ψ	=	Shadow function
ψ_ℓ	=	Smoothed shadow function

I.   Introduction

In recent years, low-thrust propulsion technologies such as Electric Propulsion (EP) have become popular as the primary means of propulsion for both planetary and interplanetary space missions. Their main appeal is a fuel efficiency which is ten times higher than that of conventional chemical thrusters. This feature is especially relevant in applications requiring large delta-v increments, a notable case being the orbital transfer about a central body. The low-thrust orbit transfer problem can be stated as the determination of a continuous spacecraft trajectory that satisfies initial and terminal conditions defined along two different orbits, while minimizing fuel expenditure and/or time of flight (TOF). Except for very special cases, this is a nonlinear optimal control problem (OCP) which is extremely hard to solve. Complications arise mainly due to the large time scale of the maneuver and to the bang-off-bang structure of the optimal control policy. An evident indication of such difficulties is that most papers on the subject deal only with a specific objective (either time or fuel optimization) or a specific transfer scenario.

Techniques for addressing the challenging issues in low-thrust trajectory optimization have been investigated for many years. Historically, the solution strategies have been divided into two general categories: indirect methods and direct methods [1]. In indirect methods, necessary conditions for optimality are derived from optimal control theory via variational arguments. The necessary conditions form a boundary-value problem which is solved numerically in order to determine the optimal solution, see, e.g., [2–4]. The primary advantage of indirect methods is that they provide highly accurate solutions. Their main drawback is the very high sensitivity of the numerical solution to the initial guess for the costate variables, which makes these methods only suitable for systems of relatively low dimension. Direct methods, on the other hand, transcribe the continuous-time optimal control problem into a static nonlinear programming problem (NLP) which is solved by a nonlinear optimizer, see, e.g., [5–7]. This alleviates the sensitivity issue, as the solution procedure is much more robust with respect to the selection of the initial trajectory guess. In particular, direct pseudospectral methods (see, e.g., [8–10]) have been applied to low-thrust problems with considerable success. In these methods, the state and control variables are parameterized using polynomials, and the system evolution is approximated via orthogonal collocation. Regardless of the transcription strategy, however, the application of direct methods typically requires the solution of large and computationally demanding NLPs. For a complete overview on these topics, the reader is referred to the dedicated books [11–13].

Despite the advances in optimization techniques and tools, the low-thrust orbit transfer problem is still a computationally difficult one. Given the practical need to ease the computation, a number of studies have focused on approximate or suboptimal solution strategies. In [14, 15], control parametrization and averaging are exploited in order to reduce the size of the optimization problem. Although these approaches are more computationally efficient compared to those based on the exact dynamic model, the accumulated errors may be large, so that the true spacecraft trajectory may fail to reach the target orbit. Lyapunov-based guidance laws such as the Q-law [16, 17] have also been proposed in order to rapidly explore the solution space. These are usually quite robust to modeling errors, but lack optimality due to their heuristic nature. For these reasons, they are commonly employed as an initial guess for subsequent optimization (see, e.g., [18]). In many cases, however, finding an adequate set of tuning parameters for such guidance laws can be as time consuming as the optimization process itself. Along a different line, some efforts have been made to study the impact of different parameterizations of the orbital motion on the computational efficiency of the optimization process, see, e.g., [19, 20]. In particular, it is found in [19] that regularizing the orbital dynamics provides an effective way to speed-up the computation. In this paper, we will show that large computational gains are

possible by working on the parametrization, while retaining the ability to solve the control problem to full optimality.

Another major source of complexity in low-thrust trajectory optimization is due to eclipsing. This is particularly relevant since most of the existing EP system are solar powered and therefore they are unable to operate during solar eclipse periods. Indeed, as pointed out in [21, 22], low-thrust trajectory optimization without concerns for eclipsing is futile when subsequent analysis reveal that a thrusting maneuver has been scheduled in the shadow region. The net effect of solar eclipses is to introduce a state-dependent discontinuity in the control input structure, which constitutes a serious challenge for gradient-based optimization techniques. Unfortunately, this important issue is often disregarded in the literature. In the seminal paper [23], the low-thrust orbit transfer problem with eclipsing is tackled by adopting an indirect method. More recently, the problem has been framed in the context of direct optimization. In particular, two different formulations have demonstrated their utility in applications: single-phase and multi-phase ones. The multi-phase formulation divides the OCP into multiple phases in which the system dynamics are smooth, while ensuring the continuity of the state trajectory among different phases through an appropriate set of event constraints. In this way, discontinuities are removed from the dynamic model and the optimization is cast over the switch times. A limitation of this method is that the structure of the optimal solution must be guessed a priori, which typically involves a rather complex machinery. For instance, it is not trivial to determine whether a phase should be added to the formulation. Multi-phase approaches have been employed in [21] and [24] for the solution of minimum-fuel and minimum-time problems with eclipsing, respectively. The single-phase formulation is defined in the usual way and considers the OCP as a whole. An advantage of this method over multi-phase formulations is that it requires no prior information about the structure of the optimal solution, although some kind of smoothing is needed for the system dynamics. In [22], a single-phase approach is employed in combination with a smoothed eclipse model in order to solve a minimum-fuel problem involving a large number of orbital revolutions. This is done by using hybrid differential dynamic programming [25]. In [26], the minimum-time problem with eclipsing is similarly cast and it is solved via collocation. A suboptimal solution to the minimum-fuel problem, which consists of applying the Q-law in the eclipsed part of the transfer and the optimal control policy in the remaining part, is also discussed.

The aim of this paper is to present a new approach for the optimization of low-thrust orbit transfers, employing a direct collocation method. The work is in the same spirit of [22, 26], but some key features are introduced in the problem definition. First, equinoctial elements are replaced by a novel nonsingular set of *ideal elements* inspired by Hansen's theory (see [27–29]) for the description of the orbital motion. The rationale behind this choice is the observation that the dynamics of the *ideal anomaly* do not directly depend on the perturbing acceleration, as opposed to those of all other anomalies. This makes the ideal anomaly attractive as a basis for regularization (see, e.g., [30,31]). Moreover, ideal elements enjoy a higher level of sparsity of the dynamic model structure compared to other element sets. This is advantageous because the OCP to be solved turns out to be a sparse NLP. As a second contribution, a flexible optimization framework is established, which unifies different types of performance requirements under the same single-phase OCP formulation. The adopted cost function involves a convex combination of TOF and fuel consumption, providing a suitable way to trade off these objectives by means of a scalar parameter. Eclipse effects are taken into account by adapting the smoothing technique in [22] to the the proposed state parametrization. The OCP is transcribed into a sparse NLP by using the commercial package GPOPS–II [32], which implements an *hp*-adaptive Legendre–Gauss–Radau pseudospectral collocation strategy. The NLP is then solved by using a sparse nonlinear optimizer. Furthermore, a new Lyapunov-based guidance scheme that addresses the initial guess generation problem is devised. The guidance scheme features only three assignable parameters, which can be tuned in an intuitive manner.

The proposed optimization architecture is tested on two orbit transfer scenarios taken from the literature: a GTO-GEO transfer and a LEO-GEO transfer. Simulation results show that the method is able to solve minimum-time, mimimum-fuel and mixed time/fuel-optimal problem instances involving many eclipse transitions in a computationally efficient manner. Moreover, the capability of the method to solve large-scale optimization problems on low-power hardware is demonstrated for a realization featuring nearly half a million NLP variables. Overall, the obtained results indicate that the adopted parametrization and optimization scheme allow one to tackle the low-thrust OCP with relative ease in comparison to previous approaches.

The paper is organized as follows. Section II illustrates the new parametrization of the orbital motion and Section III details the dynamic model used for trajectory optimization as well as the eclipse smoothing technique. Section IV discusses the proposed OCP formulation and Section V presents the Lyapunov guidance scheme employed for the initial guess generation. Detailed simulation case studies of the optimization architecture are presented in Section VI for the considered orbit transfer scenarios. Section VII summarizes the main findings of this research and finalizes the paper.

II.   Parametrization of the Orbital Motion

The first step towards the derivation of an optimal orbit transfer strategy is to define a suitable set of parameters describing the orbital motion and to determine how such parameters evolve in response to perturbations. In this section, the perturbed Kepler problem is briefly reviewed and a new parametrization of the orbital motion is presented for this problem. The proposed parametrization is based on a nonsingular set of orbital parameters inspired by Hansen's theory.

A.   Perturbed Kepler Problem

The perturbed Kepler problem amounts to describing the evolution of an orbit in response to a perturbing acceleration that accounts for all contributions other than point mass gravity. The classical solution to this problem is expressed in terms of the orbital elements $\{a, e, i, \omega, \Omega, v\}$, namely the semimajor axis, eccentricity, inclination, argument of periapsis, right ascension of the ascending node, and true anomaly. Let ϱ_R, ϱ_T and ϱ_N be the radial, transverse and normal components of the perturbing acceleration, expressed in the Radial-Transverse-Normal frame centered at the satellite. Whenever nonconservative perturbations such as thrusting are involved, it is customary to describe the evolution of the classical elements through Gauss' variational equations

$$
\begin{aligned}
\dot{a} &= \frac{2a^2}{\sqrt{\mu a(1-e^2)}}\left[e\sin(v)\,\varrho_R + (1+e\cos(v))\,\varrho_T\right] \\[2mm]
\dot{e} &= \sqrt{\frac{a(1-e^2)}{\mu}}\left[\sin(v)\,\varrho_R + \frac{e+(2+e\cos(v))\cos(v)}{(1+e\cos(v))}\varrho_T\right] \\[2mm]
\dot{i} &= \sqrt{\frac{a(1-e^2)}{\mu}}\left[\frac{\cos(v+\omega)}{1+e\cos(v)}\varrho_N\right] \\[2mm]
\dot{\Omega} &= \sqrt{\frac{a(1-e^2)}{\mu}}\left[\frac{\sin(v+\omega)}{(1+e\cos(v))\sin(i)}\varrho_N\right] \\[2mm]
\dot{\omega} &= \sqrt{\frac{a(1-e^2)}{\mu}}\left[-\frac{\cos(v)}{e}\varrho_R + \frac{(2+e\cos(v))\sin(v)}{e(1+e\cos(v))}\varrho_T - \frac{\sin(v+\omega)\cot(i)}{1+e\cos(v)}\varrho_N\right] \\[2mm]
\dot{v} &= \sqrt{\frac{\mu}{a^3}}\frac{(1+e\cos(v))^2}{(1-e^2)^{3/2}} + \sqrt{\frac{a(1-e^2)}{\mu}}\left[\frac{\cos(v)}{e}\varrho_R - \frac{(2+e\cos(v))\sin(v)}{e(1+e\cos(v))}\varrho_T\right]
\end{aligned}
\tag{1}
$$

where μ is the gravitational parameter and the overdot symbol denotes the time derivative.

As seen from Eq. (1), the variational equations for the classical elements are singular at $e = 0$ and $i = 0$, i.e., for circular and equatorial orbits. The standard method to overcome this issue is to employ a set of modified equinoctial elements defined by

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

The dynamics of the modified equinoctial elements are obtained by differentiating (2) with respect to time and using (1), which results in a highly coupled set of differential equations. In particular, the normal component ϱ_N of the perturbing acceleration affects the evolution of all the parameters in (2), except for the semiparameter p. This is undesirable from the point of view of numerical optimization, which often benefits from sparsity in the system model structure [20]. In the following, an alternative set of nonsingular elements is presented to address this issue.

B. Ideal Elements

Let us define the *ideal anomaly s* via the integral relation

$$s = \int_{t_0}^{t} \sqrt{\frac{\mu}{a^3(\tau)}} \frac{[1 + e(\tau)\cos(v(\tau))]^2}{[1 - e^2(\tau)]^{3/2}} \, d\tau \tag{3}$$

where t denotes the actual time and t_0 is the initial time. Notice from (1) that the time derivative of (3) is equal to $\dot{v}$ for the unperturbed motion ($\varrho_R = \varrho_T = \varrho_N = 0$). However, the analogy breaks down in the perturbed case. Based on (2)-(3), the following set parameters is proposed in order to describe the orbital motion

$$
\begin{aligned}
\rho &= p/R_e \\
e_x &= f\cos(L - s) + g\sin(L - s) \\
e_y &= -f\sin(L - s) + g\cos(L - s) \\
h_x &= h\cos(L - s) + k\sin(L - s) \\
h_y &= -h\sin(L - s) + k\cos(L - s) \\
\sigma &= L - s
\end{aligned}
\tag{4}
$$

where R_e is the radius of the central body (the Earth in this study). The parametrization (4) is obtained from the modified equinoctial elements by normalizing the semiparameter p, subtracting the ideal anomaly from the true longitude to obtain the slowly time-varying parameter σ, and rotating both the eccentricity vector $[f\ g]^T$ and the ascending node vector $[h\ k]^T$ by an angle $-\sigma$ about the orbit normal. The latter transformation corresponds to rotating the equinoctial frame by an angle σ about the orbit normal, so as to match the instantaneous orientation of the *Hansen ideal frame* (see, e.g., [33]). Indeed, σ is precisely the in-plane angle between the basis vectors of the Hansel frame and that of the equinoctial frame (in this paper, the initial orientation of the Hansen ideal frame, i.e., the so-called departure frame, is defined by setting $\sigma(0) = L(0)$, so that $s(0) = 0$). One can thus refer to the parameters in (4) as *ideal elements*.

The variational equations for the ideal elements are obtained from (1)-(4) as

$$
\begin{aligned}
\dot{\rho} &= \frac{\sqrt{R_e\rho}}{\sqrt{\mu}\,\kappa_c}\,2\rho\,\varrho_T \\[2mm]
\dot{e}_x &= \frac{\sqrt{R_e\rho}}{\sqrt{\mu}\,\kappa_c}\Big\{\kappa_c\sin(s)\varrho_R + \big[2\kappa_c\cos(s) + \kappa_s\sin(s)\big]\varrho_T\Big\} \\[2mm]
\dot{e}_y &= \frac{\sqrt{R_e\rho}}{\sqrt{\mu}\,\kappa_c}\Big\{-\kappa_c\cos(s)\varrho_R + \big[2\kappa_c\sin(s) - \kappa_s\cos(s)\big]\varrho_T\Big\} \\[2mm]
\dot{h}_x &= \frac{\sqrt{R_e\rho}}{\sqrt{\mu}\,\kappa_c}\big[h_x h_y\sin(s) + \kappa_x\cos(s)\big]\varrho_N \\[2mm]
\dot{h}_y &= \frac{\sqrt{R_e\rho}}{\sqrt{\mu}\,\kappa_c}\big[h_x h_y\cos(s) + \kappa_y\sin(s)\big]\varrho_N \\[2mm]
\dot{\sigma} &= \frac{\sqrt{R_e\rho}}{\sqrt{\mu}\,\kappa_c}\big[h_x\sin(s) - h_y\cos(s)\big]\varrho_N
\end{aligned}
\tag{5}
$$

where

$$
\begin{aligned}
\kappa_c &= 1 + e_x\cos(s) + e_y\sin(s) \\
\kappa_s &= e_x\sin(s) - e_y\cos(s) \\
\kappa_x &= (1 + h_x^2 - h_y^2)/2 \\
\kappa_y &= (1 - h_x^2 + h_y^2)/2
\end{aligned}
\tag{6}
$$

Notice that the right hand side of (5) is independent of the parameter σ. Moreover, the normal component ϱ_N of the perturbing acceleration affects only the parameters h_x, h_y and σ.

The particular structure of (5)-(6) suggests to adopt the ideal anomaly s as the independent integration variable. Such a procedure is commonly known as regularization [31]. Hereafter, the notation $(\cdot)' = d(\cdot)/ds$ will be adopted. From (2)-(4) and (6) one has that

$$t' = \frac{dt}{ds} = \sqrt{\frac{R_e^3}{\mu}}\,\frac{\rho^{3/2}}{\kappa_c^2} := \kappa_t \tag{7}$$

where the actual time t is now a dependent variable, i.e., $t = t(s)$. Therefore, the regularized dynamics of the ideal elements take on the form

$$[\rho' \ e'_x \ e'_y \ h'_x \ h'_y \ \sigma']^T = \kappa_t \, [\dot{\rho} \ \dot{e}_x \ \dot{e}_y \ \dot{h}_x \ \dot{h}_y \ \dot{\sigma}]^T \tag{8}$$

with $[\dot{\rho} \ \dot{e}_x \ \dot{e}_y \ \dot{h}_x \ \dot{h}_y \ \dot{\sigma}]^T$ as in (5)-(6).

III.   System Model

In this Section, the proposed parametrization of the orbital motion is exploited to define the system dynamic model used for trajectory optimization. The model includes thrusting, Earth oblateness, and solar eclipse effects.

Let us define the state vector $\mathbf{x} = [\rho \ e_x \ e_y \ h_x \ h_y \ \sigma]^T$ and the acceleration vector $\varrho = [\varrho_R \ \varrho_T \ \varrho_N]^T$. The vector ϱ accounts for the perturbing acceleration ϱ_{J2} induced by the zonal harmonic J2 of the gravitational potential and for a control acceleration $\mathbf{u}$, i.e.,

$$\varrho = \varrho_{J2} + \mathbf{u} \tag{9}$$

The contribution ϱ_{J2} can be expressed in terms of $\mathbf{x}$ and s as follows

$$\varrho_{J2}(\mathbf{x}, s) = -\frac{3\mu \kappa_c^4 J_2}{2 R_e^2 \rho^4}
\begin{bmatrix}
1 - \dfrac{12(h_x \sin(s) - h_y \cos(s))^2}{(1 + h_x^2 + h_y^2)^2} \\[2ex]
\dfrac{8(h_x \sin(s) - h_y \cos(s))(h_x \cos(s) + h_y \sin(s))}{(1 + h_x^2 + h_y^2)^2} \\[2ex]
\dfrac{4(h_x \sin(s) - h_y \cos(s))(1 - h_x^2 - h_y^2)}{(1 + h_x^2 + h_y^2)^2}
\end{bmatrix} \tag{10}$$

where J_2 is the J2 harmonic coefficient [34]. The regularized dynamics of the state vector $\mathbf{x}$ are obtained from (5)-(10) as follows

$$\mathbf{x}' = \mathbf{f}(\mathbf{x}, s) + \mathbf{G}(\mathbf{x}, s)\mathbf{u} \tag{11}$$

where $\mathbf{f}(\mathbf{x}, s) = \mathbf{G}(\mathbf{x}, s)\varrho_{J2}(\mathbf{x}, s)$ and

$$\mathbf{G}(\mathbf{x}, s) = \frac{R_e^2 \rho^2}{\mu \kappa_c^3}
\begin{bmatrix}
0 & 2\rho & 0 \\
\kappa_c \sin(s) & 2\kappa_c \cos(s) + \kappa_s \sin(s) & 0 \\
-\kappa_c \cos(s) & 2\kappa_c \sin(s) - \kappa_s \cos(s) & 0 \\
0 & 0 & h_x h_y \sin(s) + \kappa_x \cos(s) \\
0 & 0 & h_x h_y \cos(s) + \kappa_y \sin(s) \\
0 & 0 & h_x \sin(s) - h_y \cos(s)
\end{bmatrix}$$

The input vector $\mathbf{u}$ in (11) describes the acceleration generated by the spacecraft propulsion system, which in this work consists of a steerable thruster. More specifically, the three-dimensional vector $\mathbf{u}$ is parameterized as

$$\mathbf{u} = \frac{T_{max}}{m} \zeta \, \mathbf{q} \tag{12}$$

where T_{max} is the maximum deliverable thrust, m is the satellite mass, $\zeta \in [0, 1]$ is the throttle factor, and $\mathbf{q}$ is the unitary thrust direction vector ($\|\mathbf{q}\| = 1$). In this setting, the actual control commands are ζ and $\mathbf{q}$.

The mass variation due to thrusting is accounted for by augmenting system (11)-(12) with the regularized mass flow rate equation

$$m' = -\kappa_t \frac{m \|\mathbf{u}\|}{g_0 I_{sp}} = -\kappa_t \frac{T_{max}}{g_0 I_{sp}} \zeta \tag{13}$$

where g_0 denotes the standard gravity and I_{sp} is the specific impulse. The dynamic model description is completed by the timing equation (7). The full system state vector is then $[\mathbf{x}^T \ m \ t]^T$ and the corresponding regularized dynamic equations are given by (7), (11) and (13).

A.   Eclipse Conditions

A spacecraft is shadowed from the Sun when the angle θ_{es} between the Earth and the Sun, seen from the spacecraft, is smaller than the apparent angular size θ_e of the Earth radius plus the apparent angular size θ_s of the Sun radius. A standard definition for the shadow function is then

$$\psi = \begin{cases} 0 & \text{if} \quad \theta_{es} \leq \theta_e + \theta_s \\ 1 & \text{if} \quad \theta_{es} > \theta_e + \theta_s \end{cases} \tag{14}$$

where $\psi = 0$ indicates that that the satellite is shadowed, while no shadowing takes place for $\psi = 1$. The angles θ_{es}, θ_e and θ_s are given by

$$\begin{aligned} \theta_{es} &= \arccos\left(\frac{\mathbf{r}_{es}^T \mathbf{r}_{ss}}{\|\mathbf{r}_{es}\|\,\|\mathbf{r}_{ss}\|}\right) \\ \theta_e &= \arcsin(R_e/\|\mathbf{r}_{es}\|) \\ \theta_s &= \arcsin(R_s/\|\mathbf{r}_{ss}\|) \end{aligned} \tag{15}$$

where $\mathbf{r}_{es}$ and $\mathbf{r}_{ss}$ are the Earth and the Sun position vectors relative to the spacecraft, and R_s denotes the Sun radius. The satellite-Sun vector $\mathbf{r}_{ss}$ is given by

$$\mathbf{r}_{ss} = \mathbf{r}_s(t) + \mathbf{r}_{es} \tag{16}$$

where $\mathbf{r}_s(t)$ is the ECI position vector of the Sun, which is available as a function of time from ephemeris data. The position vector $\mathbf{r}_{es}$ can be expressed in terms of $\mathbf{x}$ and s as follows

$$\mathbf{r}_{es} = \kappa_r \begin{bmatrix} \cos(s-\sigma)(h_x^2 - h_y^2) + 2\sin(s-\sigma)h_x h_y + \cos(s+\sigma) \\ \sin(s-\sigma)(h_y^2 - h_x^2) + 2\cos(s-\sigma)h_x h_y + \sin(s+\sigma) \\ 2(h_x \sin(s) - h_y \cos(s)) \end{bmatrix} \tag{17}$$

where

$$\kappa_r = -\frac{R_e \rho}{\kappa_c(1 + h_x^2 + h_y^2)}$$

For the purpose of numerical optimization, the step function (14) is smoothed by using a logistic function of the form [22]

$$\psi_\ell = \frac{1}{1 + e^{c\ell}} \tag{18}$$

where $c > 0$ is an assignable gain and $\ell = \theta_e + \theta_s - \theta_{es}$. Notice that such an approximation does not necessarily involve a decrease of modeling accuracy. In fact, eclipse transitions are truly smooth physical phenomena.

IV.   Optimal Control Problem

In this paper, the control objective is to transfer a satellite from a given initial orbit towards a predefined target orbit in finite time. The target orbit is specified in terms of equinoctial elements (see (2)) by introducing the reference state vector

$$\bar{\mathbf{z}} = \begin{bmatrix} \bar{p} \\ \bar{f} \\ \bar{g} \\ \bar{h} \\ \bar{k} \end{bmatrix} = \begin{bmatrix} \bar{a}(1 - \bar{e}^2) \\ \bar{e}\cos(\bar{\Omega} + \bar{\omega}) \\ \bar{e}\sin(\bar{\Omega} + \bar{\omega}) \\ \tan(\bar{i}/2)\cos(\bar{\Omega}) \\ \tan(\bar{i}/2)\sin(\bar{\Omega}) \end{bmatrix} \tag{19}$$

Let s_f be the final value for the integration variable s. According to (4), the orbit matching condition at instant s_f can be formalized as

$$\mathbf{C}\mathbf{x}(s_f) - \mathbf{D}(\sigma(s_f))\bar{\mathbf{z}} = \mathbf{0} \tag{20}$$

where $\mathbf{C} = [\mathbf{I}_{5\times5} \; \mathbf{0}_{5\times1}]$ and

$$\mathbf{D}(\sigma) = \begin{bmatrix} 1/R_e & 0 & 0 & 0 & 0 \\ 0 & \cos(\sigma) & \sin(\sigma) & 0 & 0 \\ 0 & -\sin(\sigma) & \cos(\sigma) & 0 & 0 \\ 0 & 0 & 0 & \cos(\sigma) & \sin(\sigma) \\ 0 & 0 & 0 & -\sin(\sigma) & \cos(\sigma) \end{bmatrix} \tag{21}$$

Thrust limitations due to eclipsing are taken into account in (12) by setting

$$\zeta = \psi_\ell \, \eta \tag{22}$$

and considering $\eta \in [0, 1]$ as a new throttle control input. Substituting (22) into (12) gives the expression for the control input vector applied to system (11)

$$\mathbf{u} = \frac{T_{max}}{m} \psi_\ell \, \eta \, \mathbf{q} \tag{23}$$

where the decision variables are η and $\mathbf{q}$.

For the system model at hand, the time of flight (TOF), expressed in days, is given by

$$\Delta t = \frac{t(s_f) - t(0)}{86400} \tag{24}$$

Moreover, by using (13),(22), and the fact that $0 \le \psi_\ell \le 1$, one can establish the following upper bound on the fuel consumption

$$\Delta m = \int_0^{s_f} \kappa_t \frac{T_{max}}{g_0 \, I_{sp}} \eta \; \mathrm{d}s \tag{25}$$

where $\kappa_t = \kappa_t(\mathbf{x}, s)$ according to (6)-(7). In order to trade off fuel consumption and TOF, we define the performance index

$$J = (1 - \alpha) \, \Delta t + \alpha \, \Delta m \tag{26}$$

where $\alpha \in [0, 1]$ is a predefined constant. The considered optimal control problem is then formulated as

$$
\begin{aligned}
\min_{\mathbf{q}, \eta, s_f} \quad & J = (1 - \alpha) \, \Delta t + \alpha \, \Delta m \\
\text{s.t.} \quad & \mathbf{x}(0) = \mathbf{x}_0, \; m(0) = m_0, \; t(0) = t_0 \\
& \mathbf{x}' = \mathbf{f}(\mathbf{x}, s) + \mathbf{G}(\mathbf{x}, s) \frac{T_{max}}{m} \psi_\ell \, \eta \, \mathbf{q} \\
& m' = -\kappa_t \frac{T_{max}}{g_0 \, I_{sp}} \psi_\ell \, \eta \\
& t' = \kappa_t \\
& \mathbf{C}\mathbf{x}(s_f) - \mathbf{D}(\sigma(s_f))\bar{\mathbf{z}} = \mathbf{0} \\
& L_{min} \le \sigma(s_f) + s_f \le L_{max} \\
& \Delta t_{min} \le \Delta t \le \Delta t_{max} \\
& 0 \le \eta \le 1 \\
& \|\mathbf{q}\| = 1
\end{aligned}
\tag{27}
$$

where $\mathbf{x}_0$, m_0 and t_0 are given initial conditions and $\{L_{min}, \Delta t_{min}\}$, $\{L_{max}, \Delta t_{max}\}$ are prescribed lower and upper bounds for the terminal values of the true longitude and of the TOF (one can assign a fixed terminal longitude $\bar{L}$ and a fixed TOF $\Delta \bar{t}$ by setting $L_{min} = L_{max} = \bar{L}$ and $\Delta t_{min} = \Delta t_{max} = \Delta \bar{t}$). Notice that the final value s_f of the ideal anomaly is itself a decision variable. Problem (27) is in the form of a standard Bolza problem for the nonlinear nonautonomous system defined by the regularized dynamics of the state vector $[\mathbf{x}^T \; m \; t]^T$.

Remark 1. *The upper bound (25) has been employed to avoid including the stiff function $\psi_\ell = \psi_\ell(\mathbf{x}, t, s)$ in the optimization cost. This is done to facilitate numerical optimization. The resulting Δm is an excellent approximation of the actual fuel consumption dictated by (13),(22), due to the minimization with respect to η and the fact that for sufficiently large values of c the shadow function (18) closely approximates the step function.*

Remark 2. *The nonconvex constraint $\|\mathbf{q}\| = 1$ has been identified as a major source of computational issues for the solution of problem (27). For ease of computation, in the software implementation of (27) this constraint is reformulated by introducing the auxiliary decision vector $\mathbf{w}$ and enforcing $\mathbf{q} = \mathbf{w}/\|\mathbf{w}\|$ as well as the convex inequality $\|\mathbf{w}\|^2 \le 1$. The latter is included without loss of generality in order to bound the domain of the OCP.*

A collocation approach is employed to transcribe the continuous-time problem (27) into a static NLP, which is then solved with a suitable nonlinear optimizer. It is worth recalling that, due to the lack of convexity of problem (27), the quality of the solution will be highly dependent of the availability of a reasonable initial guess. In the following, a Lyapunov-based guidance strategy addressing the initial guess generation problem is presented.

V.   Initial Guess Generation

The initial guess basically provides the initialization point for the NLP solver. In order to generate an initial guess for the solution of problem (27), one has to determine a suitable value for s_f as well as a candidate trajectory, defined on the interval $s \in [0, s_f]$, for the state and control components $\{\mathbf{x}(s), m(s), t(s), \mathbf{q}(s), \eta(s)\}$. Technically speaking, the initial guess need not be a feasible solution to problem (27), therefore any trajectory may be used in principle. However, experience suggests that for large-scale optimization problems such as the one considered herein the initial guess must be reasonably close to feasibility, otherwise the optimizer may fail to return a solution [12]. In this respect, the application of feedback control techniques based on Lyapunov theory has seen a considerable success. A Lyapunov guidance scheme inspired by the Q-law [16, 17] is proposed below. It aims at providing a good initial guess for the optimization of low-thrust orbit transfers involving predefined changes in all orbital elements except for the true anomaly.

A.   Lyapunov-based Guidance Scheme

The first step for the derivation of the guidance scheme is to quantify the deviation between the controlled orbit and the target one (note that the parameters in Section II do not quantify such deviation, as they describe the absolute motion). To this aim, we find it convenient to employ a subset of the orbital parameters introduced in [35]. In particular, we will make use of the relative inclination γ (the angle between the orbital planes of the two orbits), and of the angles λ_2 and λ_1 made respectively by the periapsis of the controlled orbit and by the target periapsis, with respect to the relative line of nodes. By using the results in [35] together with (2), (4) and (19), one can construct a nonlinear coordinate transformation $\mathbf{y} = \mathbf{y}(\mathbf{x}, \overline{\mathbf{z}})$ such that

$$\begin{bmatrix} a & e & \gamma & \lambda_1 & \lambda_2 \end{bmatrix}^T = \mathbf{y}(\mathbf{x}, \overline{\mathbf{z}}) \tag{28}$$

Defining the relative eccentricity vector as

$$\mathbf{e} = \begin{bmatrix} e\cos(\lambda_2) - \overline{e}\cos(\lambda_1) \\ e\sin(\lambda_2) - \overline{e}\sin(\lambda_1) \end{bmatrix}, \tag{29}$$

it can be verified that the controlled and the target orbits coincide if $a = \overline{a}$, $\mathbf{e} = \mathbf{0}$ and $\gamma = 0$, where $\overline{a}$ and $\overline{e}$ denote respectively the target semimajor axis and eccentricity. As a scalar measure of the deviation of a from $\overline{a}$, of $\mathbf{e}$ from $\mathbf{0}$, and of γ from 0, we consider the Lyapunov function candidate

$$V(\mathbf{y}) = \frac{k_a}{2}\sqrt{\frac{\mu}{\overline{a}}}\left(\sqrt{\frac{\overline{a}}{a}} - 1\right)^2 + \sqrt{\frac{\mu}{p}}\left[k_e\frac{\|\mathbf{e}\|^2}{2} + k_\gamma\,\tan^2\!\left(\frac{\gamma}{2}\right)\right] \tag{30}$$

where $k_a \geq 0$, $k_e \geq 0$, $k_\gamma \geq 0$ are constant weighting parameters (for notational simplicity, the dependance on constant parameters is not made explicit in the argument of V). The rationale behind the definition of (30) is similar to that leading to the Q-law. While the Q-law attempts to quantify the "best-case quadratic time-to-go" for the maneuver, the specific form of (30) attempts to quantify the delta-v needed to bring the actual orbit to the target orbit. Indeed, it can be easily seen that the dimensional unit of (30) is m/s.

The s-derivative of (30) is evaluated along the trajectory of system (11), neglecting J2 effects (i.e., $\mathbf{f}(\mathbf{x}, s) = \mathbf{0}$). This results in

$$V' = \frac{\partial V(\mathbf{y})}{\partial \mathbf{y}}\frac{\partial \mathbf{y}(\mathbf{x}, \overline{\mathbf{z}})}{\partial \mathbf{x}}\mathbf{G}(\mathbf{x}, s)\,\mathbf{u} := \mathbf{h}^T\,\mathbf{u} \tag{31}$$

Substituting (23) into (31) gives

$$V' = \mathbf{h}^T\frac{T_{max}}{m}\,\psi_\ell\,\eta\,\mathbf{q} \tag{32}$$

In order to achieve $V' \leq 0$, the following functional form is assigned to the initial guess of the thrust direction vector $\mathbf{q}$

$$\mathbf{q} = -\frac{\mathbf{h}}{\|\mathbf{h}\|} \tag{33}$$

Substituting (33) into (32), one gets

$$V' = -\|\mathbf{h}\|\frac{T_{max}}{m}\,\psi_\ell\,\eta \tag{34}$$

so that $V' \leq 0$, as expected.

In (34), the throttle control input $\eta \in [0, 1]$ is yet to be specified. The initial guess of η is specified according to the coasting mechanism described in [17], which employs the *time* derivative of V as an indicator of whether to thrust or coast. The time derivative $\dot{V}$ is obtained from (7) and (34) as

$$\dot{V} = \frac{V'}{\kappa_t} = -\frac{\|\mathbf{h}\|}{\kappa_t} \frac{T_{max}}{m} \psi_\ell \, \eta \tag{35}$$

At each instant s, the maximum value h_{max} and minimum value h_{min} of the gain factor $\|\mathbf{h}\|/\kappa_t = \|\mathbf{h}(\mathbf{x}, s)\|/\kappa_t(\mathbf{x}, s)$ in (35) are predicted over a future time span whose length is equal to one orbital period. This is done by sweeping τ over the interval $[s, \ s + 2\pi]$ in the expression $\|\mathbf{h}(\mathbf{x}, \tau)\|/\kappa_t(\mathbf{x}, \tau)$, while holding the system state $\mathbf{x}$ constant and equal to the present value $\mathbf{x}(s)$. A minor amendment of the method in [17] is implemented so as to effectively cope with eclipse effects. It boils down to discarding eclipse periods occurring in the interval $[s, \ s + 2\pi]$ when computing h_{max} and h_{min} (in other words, h_{max} and h_{min} are evaluated over non-eclipsed orbital arcs). Then, a thrusting efficiency factor is defined as

$$\xi = \frac{\|\mathbf{h}\|/\kappa_t - h_{min}}{h_{max} - h_{min}} \tag{36}$$

A coasting phase is enforced whenever the thrusting efficiency is below a predefined threshold $\xi_{cut} \in [0, 1]$, by setting $\eta = 0$ if $\xi < \xi_{cut}$. Throttle cut-offs due to eclipsing are accounted for by setting $\eta = \psi_\ell$ when $\xi \geq \xi_{cut}$. Summarizing, one has that

$$\eta = \begin{cases} \psi_\ell & \text{if} \quad \xi \geq \xi_{cut} \\ \\ 0 & \text{if} \quad \xi < \xi_{cut} \end{cases} \tag{37}$$

For minimum time problems, it is customary to set $\xi_{cut} = 0$, so as to fire the thruster whenever it is possible. For $\xi_{cut} > 0$ a trade-off is established between TOF and control effort. For additional details about the coasting mechanism, the reader is referred to [17].

The initial guess for the state and control components $\{\mathbf{x}(s), m(s), t(s), \mathbf{q}(s), \eta(s)\}$ is generated by integrating system (7),(11),(13) with the control input (23),(33),(37). Due to the way the feedback control policy is defined, the initial guess gets close to a feasible solution to problem (27) as soon as the Lyapunov function (30) approaches zero. An interesting feature of the proposed guidance scheme is that scaling all the weighting parameters in (30) by the same positive constant does not change the control policy, and hence it does not affect the state trajectory guess. This allows one to restrict the domain of definition of $\{k_a, \ k_e, \ k_\gamma\}$ to the positive orthant of the unit sphere, by means of the spherical coordinate transformation

$$\begin{array}{rcl} k_a & = & \cos(\varphi_{az})\cos(\varphi_{el}) \\ k_e & = & \sin(\varphi_{az})\cos(\varphi_{el}) \\ k_\gamma & = & \sin(\varphi_{el}), \end{array} \tag{38}$$

where the free parameters are $\varphi_{az} \in [0, \pi/2]$ and $\varphi_{el} \in [0, \pi/2]$. In this way, the tuning of the initial guess can be conveniently cast as a two-dimensional search over $\{\varphi_{az}, \varphi_{el}\}$ for minimum time problems or a three-dimensional search over $\{\varphi_{az}, \varphi_{el}, \xi_{cut}\}$ for problems involving fuel optimization. For the case studies presented in Section VI, we adopted a naive search procedure which consists of initializing all the tuning parameters to zero, increasing φ_{az} and φ_{el} until a reasonably low TOF is achieved, and then increasing ξ_{cut} until the desired TOF/fuel trade-off is reached. By doing so, we were able to generate a suitable trajectory guess in few minutes. The use of the parametrization in [35] is instrumental to this purpose, as it allows one to describe the relative motion via a minimal set of variables, which translates into a reduced number of weighting parameters in (30). It is also worth remarking that the control policy (33),(37) is well defined for circular and equatorial orbits, as opposed to the Q-law formulation.

B.   Definition of the Initial Mesh

In order to solve (27) by collocation, one must specify an initial mesh, i.e., a discretization grid for the initial guess. In *hp*-adaptive collocation methods, the mesh is refined during the solution process either by dividing a mesh segment or by increasing the number of collocation points within the segment. Clearly, the size and thus the complexity of the optimization problem will be proportional to the resolution of the initial mesh. Moreover, as with any mesh refinement method, the performance of the *hp* method does depend upon the structure of the initial mesh. For these reasons, the definition of the initial mesh is a critical step of the solution process. The approach proposed herein for the generation of the initial mesh is as follows. First, system (7),(11),(13),(23),(33),(37) is numerically integrated and the solution is collected at equally spaced points in the domain s, unless an eclipse event is detected. The event detection routine keeps track of all eclipse transitions and associates a sample point to each transition. The resulting sample sequence is eventually downsampled by using a method that involves a reduction of the downsample factor in the event proximity.
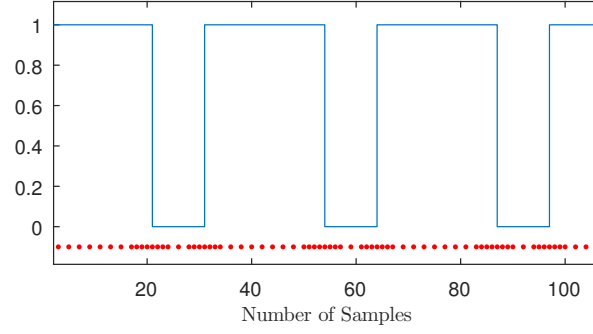
Figure 1. Illustration of the initial mesh generation strategy, showing the profile of the shadow function (14) **together with the location the mesh points.**

This produces a discretization grid which is finer in correspondence of eclipse transitions (see Fig. 1), where the system dynamics are inherently stiff. The transitions of the throttle control input η arising from (37) are not treated as events, as they vary less predictably in the solution process and can be assigned freely by the optimizer.

VI. Numerical Simulations

In this section, the results of numerical simulations are reported for two benchmark orbit transfer scenarios taken from the literature in order to demonstrate the capabilities of the proposed method. The single-phase OCP (27) is solved using the MATLAB optimal control software GPOPS–II [32] in combination with the NLP optimizer IPOPT [36]. GPOPS–II employs an *hp*-adaptive Legendre–Gauss–Radau quadrature orthogonal collocation strategy where the optimal control problem is transcribed into a large sparse NLP, and the NLP is solved on successive mesh iterations until a desired accuracy is achieved. The initial guess for the first mesh iteration is obtained as in Section V. In subsequent iterations, the solver is warm-started with the optimal solution from the previous iteration. In this study, the *hp* mesh refinement strategy described in [37] is employed. Besides the standard features of *hp* methods, the approach in [37] provides the ability to merge mesh segment and to lower the degree of the approximating polynomial, potentially reducing the size of the optimization problem. The mesh refinement accuracy tolerance is set to $\epsilon = 5 \cdot 10^{-6}$, unless otherwise specified. The number of collocation points per mesh segment is allowed to vary from 4 to 6. The IPOPT optimizer is set up with the linear solver `ma57` and an error tolerance of $5 \cdot 10^{-7}$. The first and second derivatives required by IPOPT are computed analytically by the automatic differentiation software ADiGator [38]. The constant c in (18) is set to $c = 298.78$, which according to [22] is a realistic value for Earth-centered transfers. All the simulations have been performed on a laptop equipped with a Intel Core I7-5600U processor and 16 GB of RAM.

A. GTO-GEO Transfer

Herein, we consider the GTO–GEO orbit transfer problem previously solved in [24, 26] via GPOPS-II. The equinoctial elements for the initial GTO and the target GEO are reported in Table 1. The GTO elements refer to an orbit with a perigee of 6563.6 km, an apogee of 42164.3 km, and an inclination of 28.5 deg. The GEO elements refer to a circular, equatorial orbit with a radius of 42165 km. The Julian date at the beginning of the transfer is $JD_0 = 2451625.5$. The propulsion parameters are $T_{max} = 0.31158$ N and $I_{sp} = 1800$ s, while the spacecraft initial mass is $m_0 = 1200$ kg. All planetary constants for this problem are set equal to those in [24, 26] and are not reported here for brevity.

Problem (26)-(27) is set up with $\alpha = 0$ in order to determine the minimum TOF for the transfer. The initial guess for the problem is generated by tuning the guidance scheme with $\varphi_{az} = 7 \cdot \pi/180$ rad, $\varphi_{el} = 45 \cdot \pi/180$ rad and $\xi_{cut} = 0$, resulting in trajectory with a TOF of 120.42 days and in a fuel expenditure of 169.79 kg. The trajectory obtained by solving the OCP is depicted in Fig. 2. It covers approximately 162 revolutions, in terms of the true longitude. The components of the thrust direction vector **q** are shown in Fig. 3. The throttle control input η is always forced to 1 except during eclipse phases. The resulting TOF is 118.74 days and the fuel consumption amounts to 169.38 kg. In

Table 1. Equinoctial elements of the initial and the target orbit for the GTO-GEO transfer

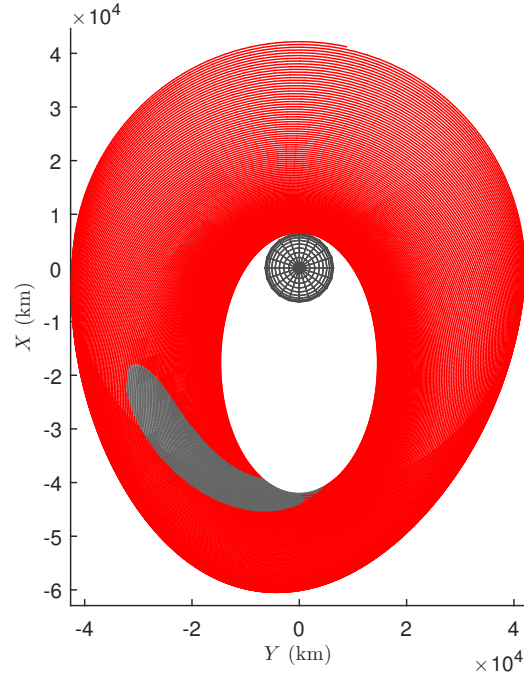| Orbit	p (km)	f	g	h	k
GTO	11359.07	0.7306	0	0.2539676	0
GEO	42165	0	0	0	0

Figure 2. Planar projection of the time-optimal GTO-GEO transfer trajectory: thrust phases are colored red, eclipse phases are colored gray.
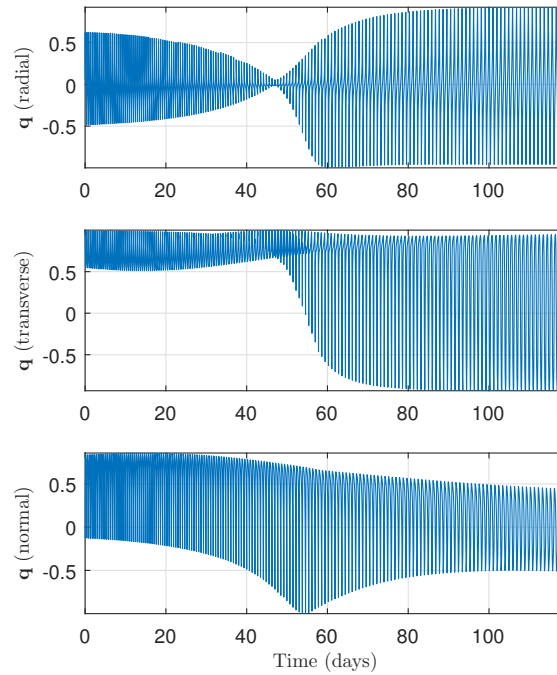

Figure 3. Thrust direction vector profile (interpolated from all non-eclipsed trajectory samples) for the time-optimal GTO-GEO transfer.

this case, they are not far from those provided by the initial guess. The total CPU time for the optimization process is 16 minutes.

The obtained results are compared with those in [24, 26] in Table 2. Both the TOF and the fuel consumption are lower than those in Ref. [24] and practically equal to those in Ref. [26]. Remarkably, the CPU time is one order of magnitude smaller compared to that in Ref. [26]. This is an especially good figure considering that in [26] the mesh error tolerance is set to 10^{-5}, while we employed a tolerance value of $5 \cdot 10^{-6}$. As a result, we ended up solving a problem with approximately twice the number of NLP variables. Since the main differences between the approach

Table 2. Literature comparison for the GTO-GEO transfer

Source	TOF (days)	Fuel (kg)	CPU time (min)
Ref. [24]	121.22	172.23	n/a
Ref. [26]	118.62	169.44	165
This paper	118.74	169.38	16

Table 3. Mesh iteration history for the time-optimal GTO-GEO problem

Iteration	Mesh error	NLP var.	CPU time
1	9.97e-5	162337	317.7 s
2	5.03e-5	127693	95.07 s
3	6.15e-6	129901	158.4 s
4	5.35e-6	130069	164.1 s
5	4.76e-6	130117	183.7 s
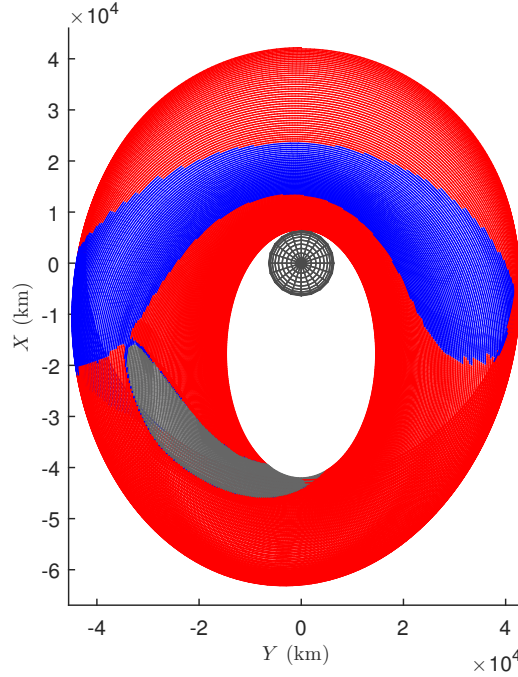
Figure 4. Planar projection of the fuel-optimal GTO-GEO transfer trajectory: thrust phases are colored red, eclipse phases are colored gray, and optimal coast phases are colored blue.

in [26] and the proposed one lie in the parametrization of the dynamic model and in the way the optimization problem is formulated, the observed gain in computational efficiency should be attributed to these two factors. A detailed breakdown of the mesh iteration history is reported in Table 3. Note that the number of variables in the first iteration is actually higher than that in the last iteration, indicating that the initial mesh is on average more dense than the final one. Although this may seem counter-intuitive, it turned out that initializing the solver with a denser initial mesh does result in a lower number of mesh iterations, which in turns leads to a shorter overall CPU time. Table 3 also demonstrates the ability of the mesh refinement algorithm [37] to compress the problem size, thus reducing the computational burden.

A fuel-optimal GTO-GEO transfer with fixed TOF has been simulated by setting $\alpha = 1$ and $\Delta t_{min} = \Delta t_{max} = 129.4$ days in (26)-(27). The initial guess is left unchanged and equal to that of the minimum time problem, so as to assess the capability of the method to locate optimal coasting arcs without prior indication (i.e., to identify the bang-off-bang structure of the fuel-optimal solution). The problem is solved twice using two different mesh error tolerance levels: $\epsilon = 5 \cdot 10^{-6}$ and $\epsilon = 5 \cdot 10^{-7}$. The solution for $\epsilon = 5 \cdot 10^{-6}$ features a fuel consumption of 159.39 kg and takes a CPU time of 25.4 min. It involves 7 mesh iterations, with a number of NLP variables in the final mesh equal to 126132. The solution for $\epsilon = 5 \cdot 10^{-7}$ features a fuel consumption of 159.24 kg and takes a CPU time of 94.6 min. The number of mesh iterations and of NLP variables in the final mesh amount respectively to 11 and 180444. It can be seen that
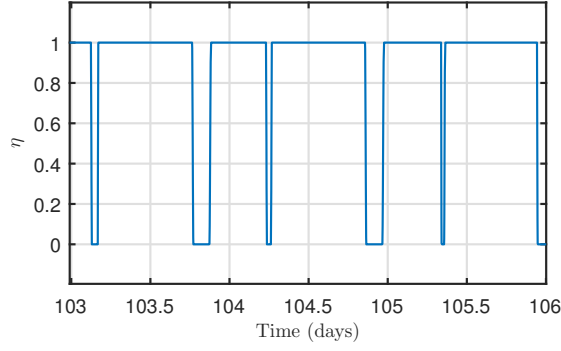
Figure 5. Detail of the throttle control input profile for the fuel-optimal GTO-GEO transfer.

Table 4. Results for the GTO-GEO transfer

Solution type	TOF (days)	Fuel (kg)	CPU time (min)
Time-optimal	118.74	169.38	16
Mixed ($\alpha = 0.5$)	122.62	162.40	36.9
Fuel-optimal	129.4 (fixed)	159.39	25.4
Fuel-optimal ($\epsilon = 5 \cdot 10^{-7}$)	129.4 (fixed)	159.24	94.6

the optimal cost is approximately the same for the two solutions, and that the computational load scales reasonably well with the mesh accuracy. From a qualitative point of view, a lower mesh error tolerance results in a more accurate localization of the coast arcs. The trajectory obtained for $\epsilon = 5 \cdot 10^{-7}$, reported in Fig. 4, covers 165 orbital revolutions and cumulates 420 on-off throttle command transitions. A detail of the throttle control input profile is shown in Fig. 5.

The mixed time/fuel-optimal problem has been investigated by solving (26)-(27) with $\alpha = 0.5$. In this case, the TOF is not assigned and the optimizer has to search for a pareto-optimal solution. The resulting performance figure is: TOF of 122.6 days, fuel consumption of 162.4 kg and CPU time of 36.9 min. The pareto-optimal solution displays a TOF that is close to the time-optimal one, while the fuel expenditure is lowered by approximately 7 kg compared to the time-optimal policy. The GTO-GEO transfer results are summarized in Table 4.

B. LEO-GEO Transfer

Herein, we consider the same LEO–GEO orbit transfer scenario studied in [21, 22]. The equinoctial elements for the initial LEO and the target GEO are reported in Table 5. The LEO elements refer to an orbit with an altitude of 500 km above the Earth surface, an inclination of 28.5 deg, and a right ascension of the ascending node of 180 deg. The GEO elements refer to a circular, equatorial orbit with a radius of about 42241 km. The Julian date at the beginning of the transfer is $JD_0 = 2457377.5$. The propulsion parameters are $T_{max} = 1.445$ N and $I_{sp} = 1849.347748$ s, while the spacecraft initial mass is $m_0 = 1000$ kg. The planetary constants for this problem are set equal to those in [21, 22].

As for the previous case study, we start by evaluating the minimum TOF for the maneuver using $\alpha = 0$. The initial guess for the minimum-time problem is generated by tuning the guidance scheme with $\varphi_{az} = 20 \cdot \pi/180$ rad, $\varphi_{el} = 30 \cdot \pi/180$ rad and $\xi_{cut} = 0$. The resulting trajectory converges in approximately 252 orbital revolutions, leading to a time of flight of 44.96 days and a fuel consumption of 289.9 kg. The much higher number of revolutions compared to the GTO-GEO transfer means that we are going to solve a more difficult optimization problem.

The solution to the time-optimal LEO-GEO problem involves 6 mesh iterations, with a number of NLP variables equal to 252529 in the initial mesh and to 199897 in the final one. The total CPU time taken by the optimization process is 28.3 minutes. The trajectory returned by the optimizer is depicted in Fig. 6. It covers approximately 251

Table 5. Equinoctial elements of the initial and the target orbit for the LEO-GEO transfer

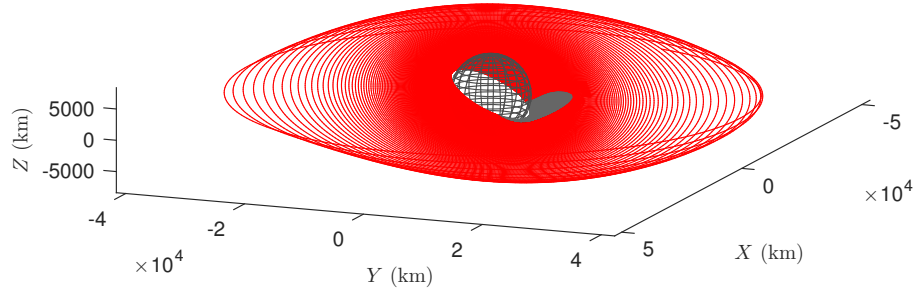| Orbit	p (km)	f	g	h	k
LEO	6878.140	0	0	-0.2539676	0
GEO	42241.095482	0	0	0	0

Figure 6. Time-optimal LEO-GEO transfer trajectory: thrust phases are colored red, eclipse phases are colored gray.
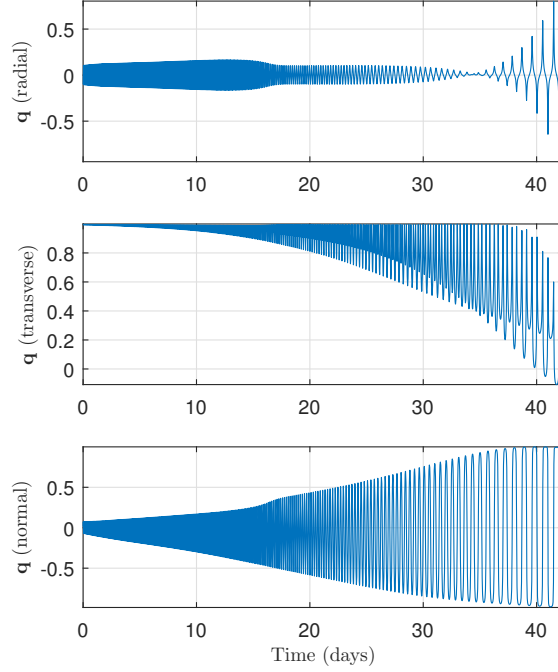

Figure 7. Thrust direction vector profile for the time-optimal LEO-GEO transfer.

Table 6. Literature comparison for the LEO-GEO transfer

Source	TOF (days)	Fuel (kg)	CPU time (min)
Ref. [21]	43.13	281.21	n/a
Ref. [22]	44.48	280.88	31.8[a]
This paper	42.37	276.70	28.3[b]

[a] Dual Intel Xeon E5-2860v3, 24 cores
[b] Intel Core I7-5600U, single core

revolutions in terms of the true longitude. The profile of the thrust direction vector components is shown in Fig. 7. Similarly to the time-optimal GTO-GEO problem, the throttle control input η is always forced to 1 except during eclipses. The optimal TOF is 42.37 days and the fuel consumption amounts to 276.70 kg.

The obtained results are compared with those in [21, 22] in Table 6. In these works, the minimum TOF for the maneuver is estimated via heuristic methods and a minimum-fuel problem is solved in which the TOF is fixed and equal to the estimated TOF. As expected, the minimum TOF resulting from the solution to (26)-(27) with $\alpha = 0$ is lower than that in [21, 22]. Remarkably, the fuel consumption is also lower than that reported in those papers. The total CPU time is close to that obtained in [22] using hybrid differential dynamic programming. Note that we employed a J2-perturbed model while Ref. [22] also models the effect of minor orbital perturbations (J3, J4, third-body). However, the results in [22] are generated on a dual Intel Xeon E5-2860v3 workstation with a significant part of
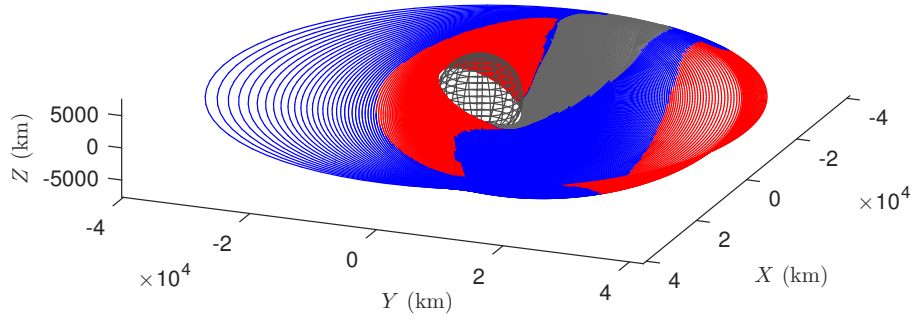
Figure 8. Fuel-optimal LEO-GEO transfer trajectory: thrust phases are colored red, eclipse phases are colored gray, and optimal coast phases are colored blue.
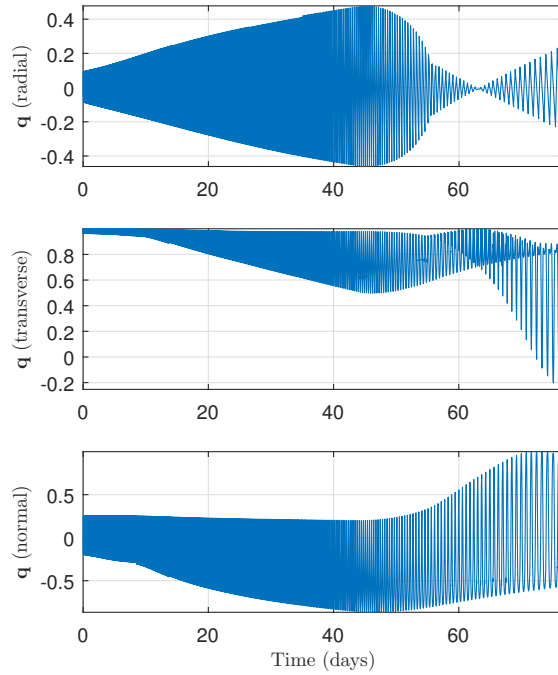

Figure 9. Thrust direction vector profile for the fuel-optimal LEO-GEO transfer.

the workload parallelized over 24 cores, while we employed a laptop processor operated on a single core. Therefore, the computational performance of the proposed solution seems to compare very favorably with that in [22]. The CPU time for Ref. [21] is not reported, because it lists the total computation time only for the last stage of its multi-phase method (based on the average iteration time reported for the previous stages, we estimated a cumulative CPU time of 47.4 minutes).

Finally, in order to challenge the proposed method, we generated a 405 revolution trajectory guess containing many thrust and coast arcs, by tuning the guidance scheme with $\varphi_{az} = 7 \cdot \pi/180$ rad, $\varphi_{el} = 30 \cdot \pi/180$ rad and $\xi_{cut} = 0.3$. This initial guess displays a TOF of 76.7 days and a fuel consumption of 271.01 kg. A minimum fuel problem is set up by using $\alpha = 1$ and constraining the problem TOF to be equal to the guessed TOF. The problem involves 434304 NLP variables on the initial mesh. Problems of this size are reportedly out of reach even for state-of-the-art low-thrust trajectory optimization software such as Mystic [18] (according to [19], computation time limits Mystic to about 250 revolutions for optimized trajectories before switching to the Q-law). We solved the problem twice using the mesh tolerance levels $\epsilon = 5 \cdot 10^{-6}$ and $\epsilon = 5 \cdot 10^{-7}$. The solution for $\epsilon = 5 \cdot 10^{-6}$ features a fuel consumption of 236.28 kg and a CPU time of 170 min. It involves 5 mesh iterations, with a number of NLP variables in the final mesh equal to 290832. The solution for $\epsilon = 5 \cdot 10^{-7}$ features a fuel consumption of 236.29 kg and a CPU time of 311.5 min. The number of mesh iterations and of NLP variables in the final mesh amount respectively to 9 and 364848. The trajectory obtained for $\epsilon = 5 \cdot 10^{-7}$, reported in Fig. 8, covers 405.45 orbital revolutions and cumulates 869 on-off throttle command transitions. The corresponding thrust direction vector profile is shown in Fig. 9.

Table 7. Results for the LEO-GEO transfer

Solution type	TOF (days)	Fuel (kg)	CPU time (min)
Time-optimal	42.37	276.70	28.3
Fuel-optimal	76.7 (fixed)	236.28	170
Fuel-optimal ($\epsilon = 5 \cdot 10^{-7}$)	76.7 (fixed)	236.29	311.5

Note that, for this example, the solution provided by the initial guess is quite far from the optimal one. This demonstrates the capability of the proposed method to effectively explore the solution space. Moreover, it highlights the advantages brought by trajectory optimization with respect to heuristic approaches such as the one in Section V, in terms of achievable performance. The LEO-GEO transfer results are summarized in Table 7.

VII.   Conclusions

A direct approach has been presented for the optimization of low-thrust orbit transfer trajectories under eclipse constraints. A specifically conceived parametrization of the orbital motion has been employed in combination with a suitable eclipse smoothing technique in order to define a flexible single-phase optimal control problem formulation. It has been shown that state-of-the-art pseudospectral collocation algorithms are able to solve this problem effectively. The optimization procedure is complemented by a Lyapunov guidance scheme that can be exploited to generate a reasonable initial guess for the nonlinear solver in short time. The proposed approach is general enough to encompass minimum-time, mimimum-fuel and mixed time/fuel-optimal control problems. Simulations on several relevant missions show that the new parametrization and optimization scheme provide a remarkable improvement in terms of computational efficiency with respect to comparable methods.

References

[1] Betts, J. T., "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 193–207. doi:10.2514/2.4231.

[2] Kechichian, J. A., "Trajectory Optimization Using Nonsingular Orbital Elements and True Longitude," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 5, 1997, pp. 1003–1009. doi:10.2514/2.4147.

[3] Haberkorn, T., Martinon, P., and Gergaud, J., "Low Thrust Minimum-Fuel Orbital Transfer: A Homotopic Approach," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 6, 2004, pp. 1046–1060. doi:10.2514/1.4022.

[4] Zhang, C., Topputo, F., Bernelli-Zazzera, F., and Zhao, Y.-S., "Low-Thrust Minimum-Fuel Optimization in the Circular Restricted Three-Body Problem," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 8, 2015, pp. 1501–1510. doi:10.2514/1.G001080.

[5] Hargraves, C. and Paris, S., "Direct Trajectory Optimization using Nonlinear Programming and Collocation," *Journal of Guidance, Control, and Dynamics*, Vol. 10, No. 4, 1987, pp. 338–342. doi:10.2514/3.20223.

[6] Enright, P. J. and Conway, B. A., "Discrete Approximations to Optimal Trajectories using Direct Transcription and Nonlinear Programming," *Journal of Guidance, Control, and Dynamics*, Vol. 15, No. 4, 1992, pp. 994–1002. doi:10.2514/3.20934.

[7] Betts, J. T., "Very Low-Thrust Trajectory Optimization Using a Direct SQP Method," *Journal of Computational and Applied Mathematics*, Vol. 120, No. 1, 2000, pp. 27 – 40. doi:10.1016/S0377-0427(00)00301-0.

[8] Elnagar, G., Kazemi, M. A., and Razzaghi, M., "The Pseudospectral Legendre Method for Discretizing Optimal Control Problems," *IEEE Transactions on Automatic Control*, Vol. 40, No. 10, 1995, pp. 1793–1796. doi:10.1109/9.467672.

[9] Ross, I. M. and Fahroo, F., "Pseudospectral Knotting Methods for Solving Nonsmooth Optimal Control Problems," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 3, 2004, pp. 397–405. doi:10.2514/1.3426.

[10] Garg, D., Hager, W. W., and Rao, A. V., "Pseudospectral Methods for Solving Infinite-Horizon Optimal Control Problems," *Automatica*, Vol. 47, No. 4, 2011, pp. 829 – 837. doi:10.1016/j.automatica.2011.01.085.

[11] Betts, J. T., *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, Society for Industrial and Applied Mathematics, 2nd ed., 2010. doi:10.1137/1.9780898718577.

[12] Conway, B., *Spacecraft Trajectory Optimization*, Cambridge Aerospace Series, Cambridge University Press, 2010. doi:10.1017/CBO9780511778025.

[13] Kéchichian, J. A., *Applied Nonsingular Astrodynamics: Optimal Low-Thrust Orbit Transfer*, Cambridge Aerospace Series, Cambridge University Press, 2018. doi:10.1017/9781108560061.

[14] Kluever, C. A. and Oleson, S. R., "Direct Approach for Computing Near-Optimal Low-Thrust Earth-Orbit Transfers," *Journal of Spacecraft and Rockets*, Vol. 35, No. 4, 1998, pp. 509–515. doi:10.2514/2.3360.

[15] Gao, Y., "Near-Optimal Very Low-Thrust Earth-Orbit Transfers and Guidance Schemes," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 2, 2007, pp. 529–539. doi:10.2514/1.24836.

[16] Petropoulos, A. E., "Low-thrust Orbit Transfers Using Candidate Lyapunov Functions with a Mechanism for Coasting," *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, Providence, Rhode Island, August 2004. doi:10.2514/6.2004-5089.

[17] Petropoulos, A. E., "Refinements to the Q-law for the Low-thrust Orbit Transfers," *AAS/AISS Space Flight Mechanics Meeting*, Copper Mountain, Colorado, January 2005.

[18] Whiffen, G., "Mystic: Implementation of the Static Dynamic Optimal Control Algorithm for High-Fidelity, Low-Thrust Trajectory Design," *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, Keystone, Colorado, August 2006. doi:10.2514/6.2006-6741.

[19] Aziz, J. D., Parker, J. S., Scheeres, D. J., and Englander, J. A., "Low-Thrust Many-Revolution Trajectory Optimization via Differential Dynamic Programming and a Sundman Transformation," *The Journal of the Astronautical Sciences*, Vol. 65, 2018, pp. 205–228. doi:10.1007/s40295-017-0122-8.

[20] Junkins, J. L. and Taheri, E., "Exploration of Alternative State Vector Choices for Low-Thrust Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 42, No. 1, 2019, pp. 47–64. doi:10.2514/1.G003686.

[21] Betts, J. T., "Optimal Low–Thrust Orbit Transfers With Eclipsing," *Optimal Control Applications and Methods*, Vol. 36, No. 2, 2015, pp. 218–240. doi:10.1002/oca.2111.

[22] Aziz, J., Scheeres, D., Parker, J., and Englander, J., "A Smoothed Eclipse Model for Solar Electric Propulsion Trajectory Optimization," *Transactions of the Japan Society for Aeronautical and Space Sciences, Aerospace Technology Japan*, Vol. 17, No. 2, 2019, pp. 181–188. doi:10.2322/tastj.17.181.

[23] Ferrier, C. and Epenoy, R., "Optimal Control for Engines with Electro-Ionic Propulsion Under Constraint of Eclipse," *Acta Astronautica*, Vol. 48, No. 4, 2001, pp. 181 – 192. doi:10.1016/S0094-5765(00)00158-2.

[24] Graham, K. F. and Rao, A. V., "Minimum-Time Trajectory Optimization of Low-Thrust Earth-Orbit Transfers with Eclipsing," *Journal of Spacecraft and Rockets*, Vol. 53, No. 2, 2016, pp. 289–303. doi:10.2514/1.A33416.

[25] Lantoine, G. and Russell, R. P., "A Hybrid Differential Dynamic Programming Algorithm for Constrained Optimal Control Problems. Part 1: Theory," *Journal of Optimization Theory and Applications*, Vol. 154, No. 2, 2012, pp. 382–417. doi:10.1007/s10957-012-0039-0.

[26] Shannon, J. L., Ozimek, M. T., Atchison, J. A., and Hartzell, C. M., "Q-Law Aided Direct Trajectory Optimization of Many-Revolution Low-Thrust Transfers," *Journal of Spacecraft and Rockets*, Vol. 57, No. 4, 2020, pp. 672–682. doi:10.2514/1.A34586.

[27] Hansen, P. A., *Auseinandersetzung einer zweckmässigen Methode zur Berechnung der absoluten Störungen der kleinen Planeten*, Abhandlungen der mathematisch-physischen Classe der königlich sächsischen Gesellschaft der Wissenschaften, dritter Band, bei S. Hirzel, Leipzig; dritte Abhandlung im fünften Band, 1859.

[28] Musen, P., *A Modified Hansen's Theory as Applied to the Motion of Artificial Satellites*, Vol. 492, National Aeronautics and Space Administration, 1960.

[29] Deprit, A., "Ideal Elements for Perturbed Keplerian Motions," *Journal of Research of the National Bureau of Standards - B. Mathematical Sciences*, Vol. 79B, No. 1, 1975, pp. 1–15.

[30] Junkins, J. and Singla, P., "How Nonlinear Is It? A Tutorial on Nonlinearity of Orbit and Attitude Dynamics," *Journal of the Astronautical Sciences*, Vol. 52, No. 1-2, 2004, pp. 7–60.

[31] Roa, J., *Regularization in Astrodynamics: Applications to Relative Motion, Low-Thrust Missions, and Orbit Propagation*, Ph.D. thesis, Technical University of Madrid, 2016.

[32] Patterson, M. A. and Rao, A. V., "GPOPS-II: A MATLAB Software for Solving Multiple-Phase Optimal Control Problems Using Hp-Adaptive Gaussian Quadrature Collocation Methods and Sparse Nonlinear Programming," *ACM Transactions on Mathematical Software (TOMS)*, Vol. 41, No. 1, 2014, pp. 1–37. doi:10.1145/2558904.

[33] Jochim, E. F. M., "The Significance of the Hansen Ideal Space Frame," *Astronomische Nachrichten*, Vol. 333, No. 8, 2012, pp. 774–783. doi:10.1002/asna.201111711.

[34] Vallado, D. A., *Fundamentals of Astrodynamics and Applications*, Springer-Verlag New York, 2nd ed., 2007, pp. 903-905.

[35] Leomanni, M., Garulli, A., Giannitrapani, A., and Quartullo, R., "Satellite Relative Motion Modeling and Estimation via Nodal Elements," *Journal of Guidance, Control, and Dynamics*, Vol. 43, No. 10, 2020, pp. 1904–1914. doi:10.2514/1.G005186.

[36] Wächter, A. and Biegler, L. T., "On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming," *Mathematical programming*, Vol. 106, No. 1, 2006, pp. 25–57. doi:10.1007/s10107-004-0559-y.

[37] Liu, F., Hager, W. W., and Rao, A. V., "Adaptive Mesh Refinement Method for Optimal Control Using Decay Rates of Legendre Polynomial Coefficients," *IEEE Transactions on Control Systems Technology*, Vol. 26, No. 4, 2018, pp. 1475–1483. doi:10.1109/TCST.2017.2702122.

[38] Weinstein, M. J. and Rao, A. V., "Algorithm 984: ADiGator, a Toolbox for the Algorithmic Differentiation of Mathematical Functions in MATLAB Using Source Transformation via Operator Overloading," *ACM Transactions on Mathematical Software*, Vol. 44, No. 2, 2017. doi:10.1145/3104990.