(Preprint) AAS 22-750

EMBEDDED HOMOTOPY FOR CONVEX LOW-THRUST TRAJECTORY OPTIMIZATION WITH OPERATIONAL CONSTRAINTS

Christian Hofmann[*] **and Francesco Topputo**[†]

A novel homotopic approach for convex low-thrust trajectory optimization is developed to improve convergence, accuracy, and computational effort compared to state-of-the-art methods. The homotopy is included in the optimization process where the homotopic parameter is updated continuously based on the constraint violation. This method is applied to a low-fidelty model that is gradually transformed into a high-fidelty model with n-body dynamics, solar radiation pressure, and a real thruster model. In addition, engine shutdown constraints are considered where the thruster is required to remain off for certain periods. The effectiveness of the approach is assessed in several fuel-optimal transfers and compared to a state-of-the-art solver.

INTRODUCTION

Trajectory design is of key importance for all space missions. Typically, trajectories with minimum fuel consumption are sought due to the high launch cost and limited propellant mass [1]. Even though this is already a challenging task, the current trend toward more autonomy aims at shifting this task on board [2]. Ideally, the spacecraft shall determine its trajectory autonomously on the fly, therefore reducing operational costs significantly and reducing the dependence from ground. Yet, designing a trajectory requires solving a nonlinear optimal control problem. As analytical solutions exist only for special cases, numerical methods are most commonly used to determine a solution. Direct and indirect methods are the state-of-the-art techniques to solve the low-thrust trajectory optimization problem [3]. Both methods require a decent initial guess, and due to the relatively small convergence domain, non-convergence is often one of the biggest issues, especially for onboard applications. Therefore, sequential convex programming (SCP) has been identified as a promising method due to its rapid speed and high reliability [4, 5]. Generally, infeasible initial guesses often suffice, and the computational effort can be significantly lower compared to standard nonlinear programming solvers. Instead of solving the nonlinear problem directly, a sequence of convex problems is solved until the original, nonlinear constraints are satisfied. Yet, such methods often suffer from poor accuracy and convergence in case of more complex dynamical models [6]. Due to the requirement that all constraints need to be convex, nonlinear dynamics are usually linearized using a first-order Taylor series. This often works well for simple problems, but becomes critical in case of more complex models [7]. When trying to solve highly nonlinear problems directly, convergence

---

[*] Ph.D. Candidate, Department of Aerospace Science and Technology, Politecnico di Milano, 20156 Milan, Italy. Email: christian.hofmann@polimi.it

[†] Professor, Department of Aerospace Science and Technology, Politecnico di Milano, 20156 Milan, Italy. Email: francesco.topputo@polimi.it

is often only achieved if a good initial guess is provided. This is the reason why SCP is mostly used with two-body dynamics only [5, 8]. Methods based on nonlinear programming techniques seem to be a more natural choice when more complex dynamical systems are considered. Only very few examples in the literature consider higher-fidelity models such as the circular restricted three-body problem [6]. In that work, however, the SCP approach was combined with a nonlinear programming method to deal with the highly nonlinear dynamics and to achieve convergence. In [7], a homotopic approach was incorporated in the sequential convex programming method to compute low-thrust trajectories when additional perturbations are present. A sequence of simpler problems is solved first, and some homotopic parameter is used to model the path to the desired, more complex problem. Even though it was demonstrated that the success rate can be increased with this approach, the number of iterations is considerably larger. The reason is that each optimization problem is solved to full optimality. This approach is similar to indirect methods where a homotopy from the smooth energy- to the discontinuous fuel-optimal problem is performed [9, 10]. The step size of this homotopic path, however, is often fixed. Potential useful information from previous iterations is not taken into account.

In this work, the step size is updated in each iteration based on the constraint violation, therefore allowing for smaller or larger steps depending on the progress of the algorithm. This technique is incorporated into a SCP algorithm where a homotopy from simple two-body dynamics toward a more sophisticated model with additional perturbations is performed. N-body dynamics, solar radiation pressure, variable specific impulse and maximum thrust, and operational constraints are considered. In particular, no-thrust periods are taken into account where the engine needs to remain off for certain periods. Even though such constraints are crucial for real space missions, they are rarely included in the trajectory optimization process [11]. The developed approach is assessed in fuel-optimal transfers to two asteroids and compared with a state-of-the-art nonlinear programming solver.

The paper is structured as follows. Section II describes the optimal control problem and discretization technique. The homotopic approach is explained in Section III, and the results are presented in Section IV. Section V concludes this paper.

OPTIMAL CONTROL PROBLEM AND FIRST-ORDER-HOLD DISCRETIZATION

We briefly state the nonlinear optimal control problem and the transformation into a convex problem. Moreover, the first-order-hold method is addressed that is used to discretize the problem in this work.

Optimal Control Problem

The equations of motion for the two-body model read in Cartesian coordinates:

$$\dot{\mathbf{r}}(t) = \mathbf{v}(t) \tag{1}$$

$$\dot{\mathbf{v}}(t) = -\frac{\mu\,\mathbf{r}(t)}{r(t)^3} + \frac{\mathbf{T}(t)}{m(t)} \tag{2}$$

$$\dot{m}(t) = -\frac{\|\mathbf{T}(t)\|_2}{g_0\, I_{\text{sp}}} \tag{3}$$

where $\mathbf{r}(t) \in \mathbb{R}^{3\times1}$ ($r = \|\mathbf{r}(t)\|_2$), $\mathbf{v}(t) \in \mathbb{R}^{3\times1}$, and $m(t) \in \mathbb{R}$ denote the position, velocity, and mass of the spacecraft, respectively. μ is the gravitational constant of the primary body, $\mathbf{T}(t) \in$

$\mathbb{R}^{3\times 1}$ the thrust vector, g_0 the gravitational acceleration at sea level, and I_{sp} the specific impulse. In this work, the initial (subscript 0) and final (subscript f) final boundary conditions are fixed (except for the final mass that is free):

$$\mathbf{r}(t_0) = \mathbf{r}_0, \quad \mathbf{v}(t_0) = \mathbf{v}_0, \quad m(t_0) = m_0 \tag{4}$$

$$\mathbf{r}(t_f) = \mathbf{r}_f, \quad \mathbf{v}(t_f) = \mathbf{v}_f \tag{5}$$

We decouple states and controls using a change of variables [12]:

$$\Gamma(t) := \frac{\|\mathbf{T}(t)\|_2}{m(t)}, \quad \boldsymbol{\tau}(t) := \frac{\mathbf{T}(t)}{m(t)}, \quad z(t) := \ln m(t) \tag{6}$$

The equations of motion in the new variables are then

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) = \begin{bmatrix} \mathbf{v}(t) \\ -\mu\,\mathbf{r}(t)/r(t)^3 + \boldsymbol{\tau}(t) \\ -\Gamma(t)/(g_0\, I_{\text{sp}}) \end{bmatrix} \tag{7}$$

with $\mathbf{x} = \left[\mathbf{r}^\top, \mathbf{v}^\top, z\right]^\top$ being the states and $\mathbf{u} = \left[\boldsymbol{\tau}^\top, \Gamma\right]^\top$ the controls. The non-convex thrust magnitude constraint $0 \leq \|\mathbf{T}(t)\|_2 \leq T_{\text{max}}$ is linearized about the reference $\bar{z}$:

$$0 \leq \Gamma(t) \leq T_{\text{max}} \mathrm{e}^{-\bar{z}} \left(1 - z(t) + \bar{z}(t)\right) \tag{8}$$

The minimum-fuel problem is then given by [5]

$$\begin{aligned}
\underset{\mathbf{u}(t)}{\text{minimize}} \quad & -z(t_f) + \lambda_\nu \, \|\boldsymbol{\nu}(t)\|_1 + \lambda_\eta \, \max(0, \eta(t)) && \text{(9a)} \\
\text{subject to:} \quad & \dot{\mathbf{x}}(t) = \mathbf{A}(\bar{\mathbf{x}}(t))\,\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{q}(\bar{\mathbf{x}}(t), \bar{\mathbf{u}}(t)) + \boldsymbol{\nu}(t) && \text{(9b)} \\
& \Gamma(t) \leq T_{\text{max}} \mathrm{e}^{-\bar{z}(t)} \left(1 - z(t) + \bar{z}(t)\right) + \eta(t) && \text{(9c)} \\
& \|\boldsymbol{\tau}(t)\|_2 \leq \Gamma(t) && \text{(9d)} \\
& \|\mathbf{x}(t) - \bar{\mathbf{x}}(t)\|_1 \leq R && \text{(9e)} \\
& \mathbf{r}(t_0) = \mathbf{r}_0, \ \mathbf{v}(t_0) = \mathbf{v}_0, \ z(t_0) = z_0 && \text{(9f)} \\
& \mathbf{r}(t_f) = \mathbf{r}_f, \ \mathbf{v}(t_f) = \mathbf{v}_f && \text{(9g)}
\end{aligned}$$

with

$$\mathbf{A}(\bar{\mathbf{x}}(t)) := \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\bar{\mathbf{x}}(t)}, \ \mathbf{B} := \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\bar{\mathbf{u}}(t)}, \ \mathbf{q}(\bar{\mathbf{x}}(t), \bar{\mathbf{u}}(t)) := \mathbf{f}(\bar{\mathbf{x}}(t), \bar{\mathbf{u}}(t)) - \mathbf{A}(\bar{\mathbf{x}}(t))\,\bar{\mathbf{x}}(t) - \mathbf{B}\bar{\mathbf{u}}(t) \tag{10}$$

Slack variables $\boldsymbol{\nu}(t)$ and $\eta(t) \geq 0$ are included in the linearized dynamics in Eq. (9b) and the performance index in Eq. (9a) to handle artificial infeasibility [5]. Equation (9d) is obtained by relaxing the constraint $\|\boldsymbol{\tau}(t)\|_2 = \Gamma(t)$ [12]. The trust-region constraint in Eq. (9e) with radius R is added to avoid artificial unboundedness and keep the linearized dynamics close to the reference solution.

First-Order-Hold Discretization

In the first-order-hold method, state and control trajectories are divided into $N-1$ equidistant segments

$$t_0 = t_1 < t_2 < \cdots < t_N = t_f \tag{11}$$

with N being the number of discretization points. The control history $\mathbf{u}(t)$ is approximated using piecewise affine functions:

$$\mathbf{u}(t) = \underbrace{\frac{t_{k+1} - t}{t_{k+1} - t_k}}_{=:\,\lambda_-(t)}\,\mathbf{u}_k + \underbrace{\frac{t - t_k}{t_{k+1} - t_k}}_{=:\,\lambda_+(t)}\,\mathbf{u}_{k+1} = \lambda_-(t)\,\mathbf{u}_k + \lambda_+(t)\,\mathbf{u}_{k+1}, \qquad t \in [t_k, t_{k+1}] \tag{12}$$

The discretized dynamics are then given by [13]

$$\mathbf{x}_{k+1} = \mathbf{A}_k\,\mathbf{x}_k + \mathbf{B}_k^-\,\mathbf{u}_k + \mathbf{B}_k^+\,\mathbf{u}_{k+1} + \mathbf{q}_k + \boldsymbol{\nu}_k \tag{13}$$

where

$$\mathbf{A}_k = \mathbf{\Phi}(t_{k+1}, t_k) \tag{14a}$$

$$\mathbf{B}_k^- = \mathbf{A}_k \int_{t_k}^{t_{k+1}} \mathbf{\Phi}^{-1}(t, t_k)\,\mathbf{B}(t)\,\lambda_-(t)\,\mathrm{d}t \tag{14b}$$

$$\mathbf{B}_k^+ = \mathbf{A}_k \int_{t_k}^{t_{k+1}} \mathbf{\Phi}^{-1}(t, t_k)\,\mathbf{B}(t)\,\lambda_+(t)\,\mathrm{d}t \tag{14c}$$

$$\mathbf{q}_k = \mathbf{A}_k \int_{t_k}^{t_{k+1}} \mathbf{\Phi}^{-1}(t, t_k)\,\mathbf{q}(t)\,\mathrm{d}t \tag{14d}$$

The state transition matrix $\mathbf{\Phi}$ satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{\Phi}(t, t_0) = \mathbf{A}(t)\,\mathbf{\Phi}(t, t_0), \qquad \mathbf{\Phi}(t_0, t_0) = \mathbf{1} \tag{15}$$

where $\mathbf{1}$ is the identity matrix. Integrating the state transition matrix in Eq. (15) along with the nonlinear dynamics in Eq. (7) and the integrands in Eqs. (14b)–(14d) simultaneously yields the matrices $\mathbf{A}_k \in \mathbb{R}^{7\times 1}$, $\mathbf{B}_k^- \in \mathbb{R}^{7\times 4}$, $\mathbf{B}_k^+ \in \mathbb{R}^{7\times 4}$, and the vector $\mathbf{q}_k \in \mathbb{R}^{7\times 1}$. The reference state at $t \in [t_k, t_{k+1}]$ is computed using

$$\bar{\mathbf{x}}(t) = \bar{\mathbf{x}}_k + \int_{t_k}^{t} \mathbf{f}(\mathbf{x}(\xi), \mathbf{u}(\xi))\,\mathrm{d}\xi \tag{16}$$

EMBEDDED HOMOTOPIC APPROACH FOR HIGH-FIDELITY MODELS

A homotopic approach is a popular choice for indirect methods to solve complex optimal control problems. Due to the small convergence domain, a series of easier (often smooth) problems is solved first where the solution of each subproblem is used as the initial guess for the next problem. This process continues until the desired (often non-smooth) problem is eventually solved. The problems are connected by the homotopic parameter $\varepsilon \in [0, 1]$ that is successively decreased by $\Delta\varepsilon$ from $\varepsilon = 1$ (corresponds to an easier problem) to $\varepsilon = 0$ (corresponds to the problem that is to be solved). In general, each subproblem is solved to full optimality and the step size $\Delta\varepsilon$ is fixed and defined by the user [7]. Often, a small step size is chosen to avoid non-convergence. This, however, may result in a large number of homotopic steps and thus, iterations and CPU time increase. Even if a larger step size is chosen and then reduced by some heuristic when the solver fails, this would require additional (and unwanted) iterations and CPU time. It is therefore desirable to adjust the step size dynamically based on the progress of the algorithm. In this work, we use the maximum constraint violation c_{max} as a measure to decide when and to what extent to update ε. Instead of solving each subproblem to full feasibility and optimality, ε is adjusted within each major SCP iteration. Figure 1 illustrates two homotopic paths h_1 and h_2 that relate the homotopic parameter to the maximum constraint violation. c_{max} gives an indication about the progress of the algorithm in terms of feasibility, and a large value suggests that the current solution is far from a feasible point. Therefore, we want to solve an easier problem to drive the algorithm toward feasibility, and ε should be selected to be close to 1. When c_{max} decreases, the complexity of the model can be increased and hence, ε is reduced according to the homotopic path. This procedure continues until the desired problem with $\varepsilon = 0$ is solved. The boundaries of the homotopic path can be chosen based on the maximum constraint violation of the initial guess c_{guess} and the desired feasibility tolerance c_{tol}. This approach is flexible because for a simple optimization problem (or if a good initial guess is provided), c_{max} decreases quickly and the algorithm computes larger step sizes that result in faster convergence. Likewise, if a problem is more difficult to solve, c_{guess} varies only slowly, and consequently, $\Delta\varepsilon$ will also be small to stay within the feasible region.

This approach is used to gradually increase the fidelity of the dynamical model as explained in the following sections.

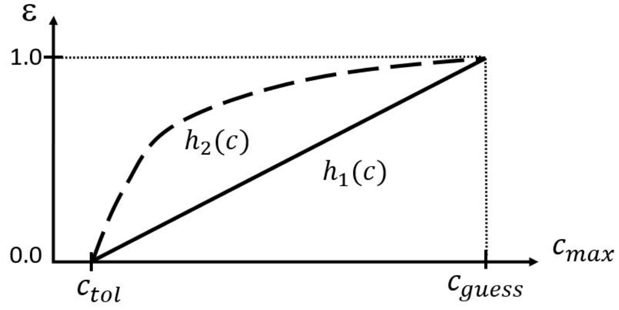


Figure 1: Two examples of smooth homotopy paths $h_1(c)$ and $h_2(c)$ that define the relationship between the homotopic parameter ε and the maximum constraint violation c_{max}.

N-Body Dynamics and Solar Radiation Pressure

Instead of solving the problem with n-body dynamics and solar radiation pressure (SRP) directly, we include the homotopic parameters $\varepsilon_{\text{nbody}}$ and ε_{SRP} in the dynamical model to obtain:

$$\mathbf{f}(\mathbf{x},\mathbf{u}) = \begin{bmatrix} \mathbf{v} \\ -\mu\,\mathbf{r}/r^3 \\ 0 \end{bmatrix} + (1-\varepsilon_{\text{nbody}}) \begin{bmatrix} \mathbf{0} \\ \mathbf{a}_{\text{nbody}} \\ 0 \end{bmatrix} + (1-\varepsilon_{\text{SRP}}) \begin{bmatrix} \mathbf{0} \\ \mathbf{a}_{\text{SRP}} \\ 0 \end{bmatrix} + \mathbf{B}\mathbf{u} \tag{17}$$

where $\mathbf{a}_{\text{nbody}}$ and $\mathbf{a}_{\text{SRP}}$ are the additional accelerations caused by the gravitational pull of other bodies in the solar system and the solar radiation pressure, respectively. $\mathbf{Bu}$ is the control term (see Eq. (9b)). When the constraint violation is large, ε is close to one, and the perturbing acceleration hence small. When $\varepsilon = 0$, the full perturbation is considered. That way, convergence improves as the nonlinear term is only added step by step.

A simple cannonball model is used for the solar radiation pressure where the projected area A_{SC} of the spacecraft is assumed constant, and the acceleration is given by [14]

$$\mathbf{a}_{\text{SRP}} = \frac{S_{\text{Sun}} C_R A_{\text{SC}}}{m} \frac{\mathbf{r}}{r^3} \tag{18}$$

with

$$S_{\text{Sun}} = \frac{L_{\text{Sun}}}{4\pi c} \tag{19}$$

$$L_{\text{Sun}} = 4\pi C_{\text{Sun}} \text{AU}^2 \tag{20}$$

C_R is the reflectivity coefficient of the spacecraft, S_{Sun} the solar pressure constant, L_{Sun} the luminosity of the Sun, and C_{Sun} the solar constant. c and AU denote the speed of light and astronomical unit, respectively. Note that the mass m in the denominator of Eq. (18) is replaced by e^z in the implementation due to the change of variables according to Eq. (6).

With regard to the n-body dynamics, the $n = 10$ perturbations of the barycenters of Mercury, Venus, Earth, Moon, Mars, Jupiter, Saturn, Uranus, Neptune, and Pluto are considered in this work. The perturbing acceleration is calculated as follows:

$$\mathbf{a}_{\text{nbody}} = \sum_{i}^{n} \mu_i \left(\frac{\mathbf{r}_{\text{sat},i}}{r_{\text{sat},i}^3} - \frac{\mathbf{r}_i}{r_i^3} \right) \tag{21}$$

μ_i is the gravitational constant of the ith body, $\mathbf{r}_i$ is the position of the ith body with respect to the Sun, and $\mathbf{r}_{\text{sat},i} = \mathbf{r}_i - \mathbf{r}$ denotes the position of the ith body with respect to the spacecraft. The time-dependent positions $\mathbf{r}_i$ of the perturbing bodies can be obtained using the software SPICE [15].

Variable Specific Impulse and Maximum Thrust

With regard to the variable specific impulse I_{sp} and maximum thrust $T_{\max}$ homotopy, a standard approach is to solve the problem with fixed I_{sp} and $T_{\max}$, and then use this as the initial guess to solve another optimization problem with the real thruster model. This is not ideal due to two reasons: first, the solver might not be able to find a solution even when the solution of the constant I_{sp} and $T_{\max}$ case is provided as the initial guess for the real model. This is especially true if the constant and real thruster models differ a lot. Second, such an approach would require solving another optimization problem with a potentially significantly different thruster curve, and therefore more computational effort and time. We propose two modifications:

1. A smooth representation of the input power P_{in} over $T_{\max}$ curve.
2. A homotopy from an easier to the real thruster model to improve convergence.

As $T_{\max}$ depends on the input power, the saturation logic of a real thruster is given by [16]

$$T_{\max} = \begin{cases} T_{\max}(P_{\text{in,max}}) & \text{if } P_{\text{in}}(r) > P_{\text{in,max}} \\ T_{\max}(P_{\text{in}}(r)) & \text{if } P_{\text{in,min}} \le P_{\text{in}}(r) \le P_{\text{in,max}} \\ 0 & \text{if } P_{\text{in}}(r) < P_{\text{in,min}} \end{cases} \tag{22}$$

where r is the distance to the Sun, and $P_{\text{in,min}}$ and $P_{\text{in,max}}$ denote the minimum and maximum input power, respectively. As this results in a non-smooth curve, we propose the following smooth T_{max} function:

$$T_{\text{max}} = \begin{cases} T_{\text{max}}(P_{\text{in,max}}) & \text{if } P_{\text{in}}(r) > P_{\text{in,max}} + \rho \\ g(P_{\text{in}}(r)) & \text{if } P_{\text{in,max}} - \rho \leq P_{\text{in}}(r) \leq P_{\text{in,max}} + \rho \\ \frac{1}{2}T_{\text{max}}(P_{\text{in}}(r)) \left[\tanh\left(\frac{P_{\text{in}}(r) - P_{\text{in,min}}}{\rho}\right) + 1\right] & \text{if } P_{\text{in}}(r) < P_{\text{in,max}} - \rho \end{cases} \tag{23}$$

with some function $g(P_{\text{in}}(r))$ that depends on the order of the polynomial of the P_{in}-T_{max} curve, and a smoothing parameter ρ. This function allows us to adjust the smoothness of the curve and the maximum available thrust at the same time. This way we can not only compute analytical derivatives, but also control the complexity of the problem. This is done by shifting the P_{in}-T_{max} curve along the x-axis, which increases the maximum available thrust for a given input power as shown in Fig. 2. It is expected that the problem is easier to solve if more thrust is available. At the same time, ρ is adjusted to smooth the curve, thus enhancing convergence as abrupt changes of T_{max} values are avoided. When the constraint violation reduces, the complexity is increased by gradually shifting the curve back to the original one and decreasing the smoothness. Linear maps $\mathcal{M}_{\Delta P}$, $\mathcal{M}_{\rho}$ are defined to describe the relationship between the shifting of the power curve ΔP, the smoothing ρ, and the homotopic parameter $\varepsilon \in [0, 1]$:

$$\mathcal{M}_{\Delta P} : \varepsilon \longmapsto \Delta P_{\text{min}} + \frac{\Delta P_{\text{max}} - \Delta P_{\text{min}}}{\varepsilon_{\text{max}} - \varepsilon_{\text{min}}} (\varepsilon - \varepsilon_{\text{min}}) \tag{24a}$$

$$\mathcal{M}_{\rho} : \varepsilon \longmapsto \rho_{\text{min}} + \frac{\rho_{\text{max}} - \rho_{\text{min}}}{\varepsilon_{\text{max}} - \varepsilon_{\text{min}}} (\varepsilon - \varepsilon_{\text{min}}) \tag{24b}$$

where P_{min}, P_{max}, ρ_{min}, and ρ_{max} are constants that depend on the problem. ε_{min} and ε_{max} are the lower and upper bound of the homotopic parameter, typically 0 and 1.

Remark: The matrix $\mathbf{B}$ in Eq. (9b) becomes now $\mathbf{B}(\mathbf{x})$ due to the dependence on the distance to the Sun. Instead of linearizing this term, we use the approximation $\mathbf{B}(\mathbf{x}) \approx \mathbf{B}(\bar{\mathbf{x}})$. Numerical simulations suggest that this is often advantageous for convergence. Likewise, the maximum available thrust depends on r, and we approximate it using $T_{\text{max}}(r) \approx T_{\text{max}}(\bar{r})$.

Remark: Even though different homotopic parameters can be defined for each perturbation, we combine them into one such that $\varepsilon := \max\left(\varepsilon_{\text{nbody}}, \varepsilon_{\text{SRP}}, \varepsilon_{T_{\text{max}}}\right)$. This reduces the number of parameters and complexity of the algorithm.

No-Thrust Periods

We focus on operational constraints where the thruster has to remain off for certain periods. This is crucial for real space missions as thrusting for several weeks or even months is often not feasible due to mission constraints such as duty cycles. We briefly describe how thruster off periods can be included in a first-order hold discretization method by extending the approach in [17]. No-thrust periods Δt_{off} in a given trajectory segment $[t_k, t_{k+1}]$ are defined as follows:

$$[t_i, t_i + \Delta t_{\text{off},i}], \quad \forall i = 1, \ldots, n \tag{25}$$

Without loss of generality, we require them to lie strictly within the segment, i.e. $t_i > t_k$ and $t_i + \Delta t_{\text{off},i} < t_{k+1}$. Therefore,

$$\mathbf{u}(t) = 0, \qquad \forall t \in [t_i, t_i + \Delta t_{\text{off},i}] \tag{26}$$

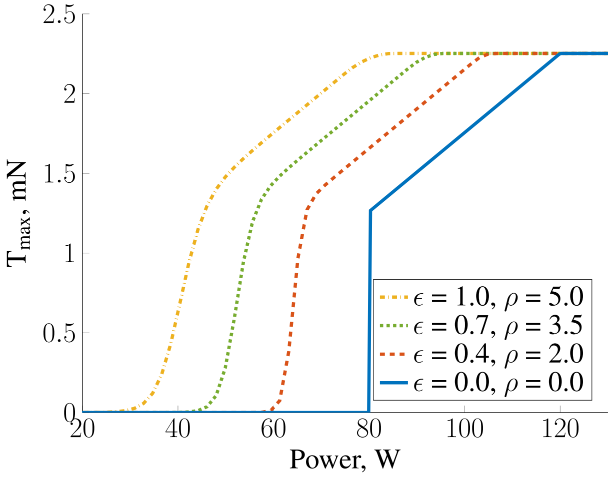


Figure 2: Illustration of shifting and smoothing the power vs. $T_{\max}$ curve.

and

$$\mathbf{u}(t) = \tilde{\lambda}_{-}(t)\,\mathbf{u}_k + \tilde{\lambda}_{+}(t)\,\mathbf{u}_{k+1}, \qquad \forall t \notin [t_i, t_i + \Delta t_{\text{off},i}] \tag{27}$$

where $\tilde{\lambda}_{-}(t)$ and $\tilde{\lambda}_{+}(t)$ are the factors that define the control interpolation. Assuming that there is one no-thrust period in a segment, we have

$$[t_k, t_k + \Delta t_{\text{on}}] \longrightarrow \text{thrust} \tag{28}$$
$$[t_1, t_1 + \Delta t_{\text{off}}] \longrightarrow \text{no thrust} \tag{29}$$
$$[t_2, t_{k+1}] \longrightarrow \text{thrust} \tag{30}$$

where $t_1 \coloneqq t_k + \Delta t_{\text{on}}$ and $t_2 \coloneqq t_1 + \Delta t_{\text{off}}$. The modified dynamics are then given by

$$\mathbf{x}(t_1) = \mathbf{A}(t_1, t_k)\,\mathbf{x}(t_k) + \mathbf{B}^{-}(t_1, t_k)\,\mathbf{u}(t_k) + \mathbf{B}^{+}(t_1, t_k)\,\mathbf{u}(t_1) + \mathbf{q}(t_1, t_k) \tag{31}$$
$$\mathbf{x}(t_2) = \mathbf{A}(t_2, t_1)\,\mathbf{x}(t_1) + \mathbf{q}(t_2, t_1) \tag{32}$$
$$\mathbf{x}(t_{k+1}) = \mathbf{A}(t_{k+1}, t_2)\,\mathbf{x}(t_2) + \mathbf{B}^{-}(t_{k+1}, t_2)\,\mathbf{u}(t_2) + \mathbf{B}^{+}(t_{k+1}, t_2)\,\mathbf{u}(t_{k+1}) + \mathbf{q}(t_{k+1}, t_2) \tag{33}$$

Substituting Eqs. (31) and (32) into Eq. (33) yields the state at t_{k+1} when no-thrust periods are considered. The matrices $\mathbf{A}(\cdot,\cdot)$, $\mathbf{B}^{-}(\cdot,\cdot)$ and $\mathbf{B}^{+}(\cdot,\cdot)$ and the vector $\mathbf{q}(\cdot,\cdot)$ can be obtained using the relations in Eqs. (14b)–(14d).

Remark: It is straightforward to include an arbitrary number of no-thrust periods within one segment using the same procedure.

NUMERICAL SIMULATIONS

The performance of the developed homotopic approach in terms of success rate, CPU time, and obtained final mass is compared with a standard SCP method without any homotopy [5] and the state-of-the-art optimal control software GPOPS-II [18] in combination with the Sparse Nonlinear Optimizer (SNOPT) [19]. Moreover, it is investigated how SCP performs when no-thrust periods are included. All simulations are performed in MATLAB on an Intel Core i7-8565 1.80 GHz Laptop with four cores and 16 GB of RAM. With regard to SCP, the numerical integration of Eq. (7)

and Eqs. (14b)–(14d) is performed using a mex function. The open-source Embedded Conic Solver (ECOS) [20] is used to solve the second-order cone program within SCP. We compute fuel-optimal trajectories from the Sun-Earth Lagrange point L_2 (SEL$_2$) to the asteroids 2000 SG344 and Dionysus. Relevant parameters for the transfers and algorithms are given in Tables 1 and 2. Physical constants are given in Table 3.

Table 1: Simulation values for the transfers from SEL$_2$ to the asteroids 2000 SG344 and Dionysus [16, 21].

Parameter	SEL$_2$ - 2000 SG344	SEL$_2$ - Dionysus
Initial epoch	04-Feb-2024 12:00:00 UTC	23-Dec-2012 00:00:00 UTC
Initial position $\mathbf{r}_0$, AU	$[-0.70186065, 0.70623244, -3.51115 \times 10^{-5}]^\top$	$[-0.023941014, 0.99325372, -3.02763 \times 10^{-5}]^\top$
Initial velocity $\mathbf{v}_0$, VU	$[-0.73296949, -0.71590485, 4.40245 \times 10^{-5}]^\top$	$[-1.02637347, -0.02809721, 1.98538 \times 10^{-6}]^\top$
Initial mass m_0, kg	22.6	4000
Final position $\mathbf{r}_f$, AU	$[0.41806795, 0.82897114, -0.00143382]^\top$	$[-2.04061782, 2.05179130, 0.55428895]^\top$
Final velocity $\mathbf{v}_f$, VU	$[-0.96990332, 0.43630220, -0.00123381]^\top$	$[-0.14231932, -0.45108800, 0.01894690]^\top$
Final mass m_f, kg	free	free
Min. input power $P_{\text{in,min}}$, W	90	62.5
Max. input power $P_{\text{in,max}}$, W	120	1000
Max. thrust $T_{\max}$, N	2.2519×10^{-3}	0.5
Max. specific impulse $I_{\text{sp,max}}$, s	3067	3000
Spacecraft area A_{SC}, m^2	0.05	100
Reflectivity coefficient C_R	1.3	1.3
Time of flight t_f, days	700	3534

Table 2: Parameters of the algorithms.

Parameter	Value
Feasibility tol. ε_c	10^{-6}
Optimality tol. ε_ϕ	10^{-4}, 10^{-5}
Max. iterations	1500, 3000

Table 3: Physical constants in all simulations.

Parameter	Value
Gravitational const. μ	1.3271244×10^{11} km^3/s^2
Gravitational accel. g_0	9.80665×10^{-3} km/s^2
Length unit AU	$1.495978707 \times 10^{8}$ km
Velocity unit VU	$\sqrt{\mu/\text{AU}}$ km/s
Time unit TU	AU/VU s
Acceleration unit ACU	VU/TU km/s^2
Mass unit MU	m_0

N-body dynamics, solar radiation pressure, and a real thruster model with variable specific impulse and maximum thrust are considered. In particular, the following thruster model is used for the transfer to 2000 SG344 [16]:

$$T_{\text{max}}(P_{\text{in}}) = a_0 + a_1 P_{\text{in}} + a_2 P_{\text{in}}^2 + a_3 P_{\text{in}}^3 + a_4 P_{\text{in}}^4 \tag{34a}$$

$$I_{\text{sp}}(P_{\text{in}}) = b_0 + b_1 P_{\text{in}} + b_2 P_{\text{in}}^2 + b_3 P_{\text{in}}^3 + b_4 P_{\text{in}}^4 \tag{34b}$$

$$P_{\text{in}}(r) = c_0 + c_1 r + c_2 r^2 + c_3 r^3 + c_4 r^4 \tag{34c}$$

with $a_0 = -0.7253$ mN, $a_1 = 0.02481$ mN/W, $a_2 = a_3 = a_4 = 0$, $b_0 = 2652$ s, $b_1 = -18.123$ s/W, $b_2 = 0.3887$ s/W^2, $b_3 = -0.00174$ s/W^3, $b_4 = 0$, and $c_0 = 840.11$ W, $c_1 = -1754.3$ W/AU, $c_2 = 1625.01$ W/AU^2, $c_3 = -739.87$ W/AU^3, $c_4 = 134.45$ W/AU^4, and r in AU. For Dionysus, we use a modified version of the model presented in [22]:

$$T_{\text{max}}(P_{\text{in}}) = \tilde{a}_0 + \tilde{a}_1 P_{\text{in}} \tag{35a}$$

$$I_{\text{sp}}(P_{\text{in}}) = I_{\text{sp,max}} \tag{35b}$$

$$P_{\text{in}}(r) = \frac{1}{r^2} \tag{35c}$$

where $\tilde{a}_1 = 0.1069$ N, $\tilde{a}_2 = 3.9307 \times 10^{-4}$ N/W, and r in AU.

Assessment of the Embedded Homotopic Approach

With regard to GPOPS-II, we choose different numbers of nodes and optimality tolerances to account for potential discrepancies in the performance if an inappropriate value is selected. We consider 5 and 10 nodes per segment for each transfer, and choose 30 and 15 segments for 2000 SG344, respectively, and 50 and 25 segments for Dionysus, respectively. This way, the total number of nodes is 150 (2000 SG344) and 250 (Dionysus), respectively. A total of 101 (2000 SG344) and 301 simulations (Dionysus) is performed. A simple perturbed cubic interpolation is used to generate infeasible initial guesses that neither satisfy the dynamical nor the endpoint constraints [23]. The numbers of revolutions of the initial guesses range from 1.6 to 2.6 for 2000 SG344, and 4 to 7 for Dionysus. Note that we do not include no-thrust periods in this simulation to ensure a fair comparison with GPOPS-II.

The results are shown in Figs. 3 and 4. SCP is the standard SCP method without homotopy, SCP$_{\text{H}}$ the homotopic approach. The notation GPOPS$_i^k$ refers to the number of nodes $i \in \{5, 10\}$ per segment, and $k \in \{10^{-4}, 10^{-5}\}$ to the optimality tolerance for GPOPS-II. Even though we use 10^{-4} for SCP, selecting a smaller value would not change the results as feasiblity is the main factor for SCP. With regard to the 2000 SG344 transfer, the benefit of a homotopic approach becomes clear in Fig. 3a. When the nonlinear problem is solved directly with a SCP method, the success rate is only 61 %, whereas SCP$_{\text{H}}$ is able to converge in more than 76 % of the cases. Remarkably, only a low success rate of approximately 31 - 36 % is obtained with GPOPS-II regardless of the number of nodes and optimality tolerance. The reason is that this example is particularly challenging due to the large minimum input power of 90 W, hence making a homotopic approach highly beneficial. According to Fig. 3b, the CPU time is lowest for SCP, followed by SCP$_{\text{H}}$ which requires more iterations. GPOPS-II and SNOPT as a nonlinear programming solver require more computational effort and therefore more time. The final mass, however, is similar for all methods (cf. Fig. 3c). Moreover, all methods yield an excellent accuracy as the propagation error (i.e. the difference between the optimized final state and the state obtained when integrating the dynamics with the obtained controls) is of the order of few kilometers only.

Although the standard SCP method is able to converge in 73% of the simulations of the Dionysus transfer, this number can be increased to 83% when including the embedded homotopic approach (see Fig. 4a). Depending on the optimality threshold, GPOPS-II achieves a success rate of only 64% ($\varepsilon_\phi = 10^{-4}$) and 50% ($\varepsilon_\phi = 10^{-5}$). Clearly, GPOPS-II struggles to find an optimal solution when the tolerance is small. However, if a larger value is chosen, the obtained final masses can be considerably lower as shown in Fig. 4c, also compared to SCP. With regard to the mean CPU time per simulation, it is evident from Fig. 4b that both SCP methods require up to one order of magnitude fewer seconds to converge than GPOPS-II. Even though the CPU time per iteration is similar for SCP and SCP_H, the latter requires twice as many iterations and therefore, also twice as much CPU time. The propagation error is of the order 10^1 to 10^2 km, therefore again being very small.

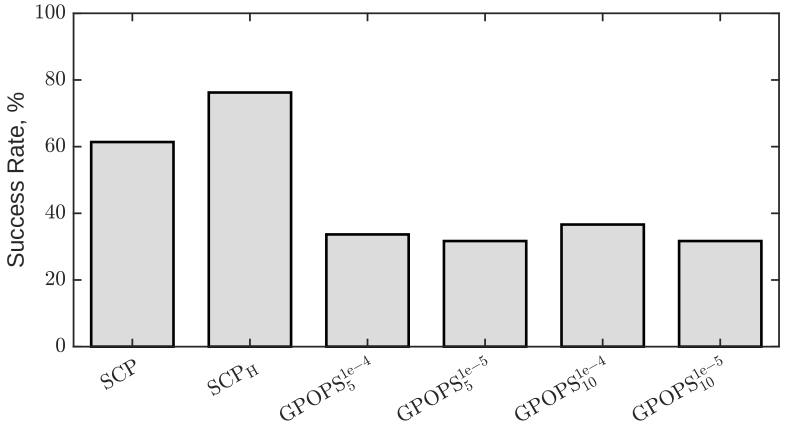
Even though a homotopic approach requires in general more iterations, the simulations confirm that convergence can be improved by gradually increasing the complexity of the dynamical model. Due to the rapid speed of SCP, the additional iterations may not be as relevant when compared to methods that usually require more computational effort, such as nonlinear programming solvers. Remarkably, the number of iterations can be reduced considerably with an embedded approach compared to the method in [7] where each optimization problem is solved to full optimality.

Performance With No-Thrust Periods

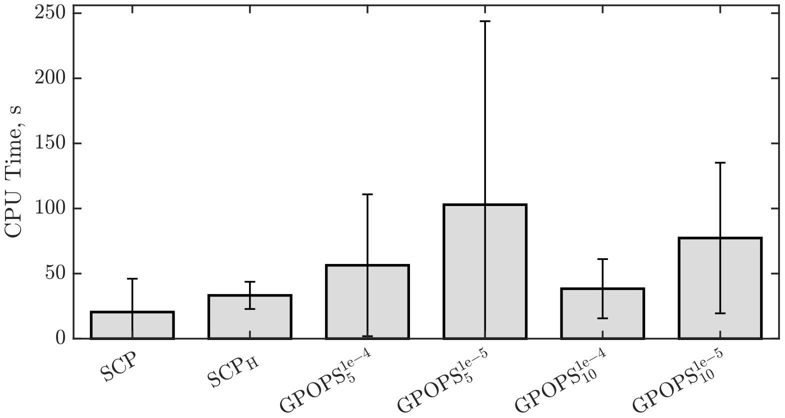
We briefly assess the solutions obtained with SCP when including no-thrust periods as per Section III. In this context, we consider periods of two (2000 SG344) and three (Dionysus) days per segment where the thruster has to remain off. Typical profiles of the thrust magnitude are shown in Figs. 5a and 5b. The dotted black line refers to the maximum available thrust, and the vertical lines correspond to the on and off switches. Clearly, the thrust magnitude follows the available maximum thrust, being either zero or taking the maximum value that depends on the distance to the Sun. Note that the available thrust is zero between 175 and 250 days in Fig. 5a because the input power drops below the required minimum value of 90 W. With regard to the Dionysus transfer, it is evident that the available maximum thrust decreases considerably over time as the spacecraft gets farther away from the Sun. Due to the long transfer time, there are many no-thrust periods, making this problem difficult to solve. The corresponding trajectories are depicted in Figs. 6a and 6b. The discontinuous thrust arcs in red show the no-thrust periods. Compared to the transfer to 2000 SG344 which requires nearly two revolutions, more than five revolutions are needed to reach Dionysus. Despite the higher complexity when no-thrust constraints are considered, the CPU time and number iterations do not increase significantly. If SCP is able to determine an optimal solution for the problem without no-thrust constraints, it is in general also able to converge successfully when such constraints are considered using the homotopic approach (provided that a solution exists).

CONCLUSION

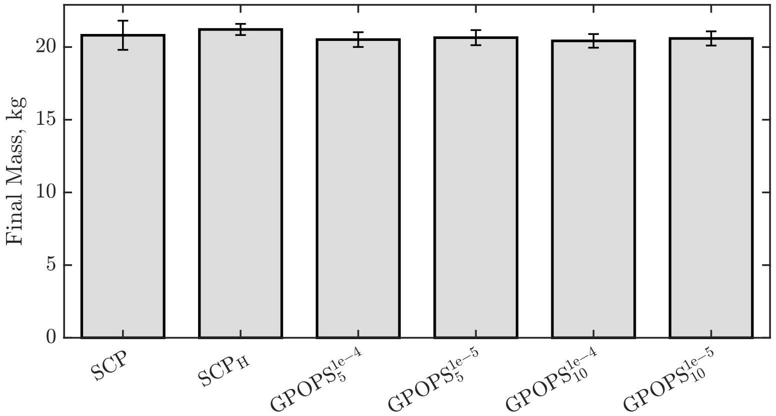
This work combines the sequential convex programming method with an embedded homotopic approach to solve the low-thrust trajectory optimization problem for high-fidelity models. The simulations show that increasing the complexity of the dynamical model only step by step can increase convergence. Despite the larger number of iterations that is required to reach convergence, the rapid speed of the proposed method makes it an excellent alternative to nonlinear programming solvers even for highly nonlinear problems. As no-thrust constraints can directly be included in the optimization process without a significant increase in computational effort and decrease in convergence,



(a) Comparison of success rate.

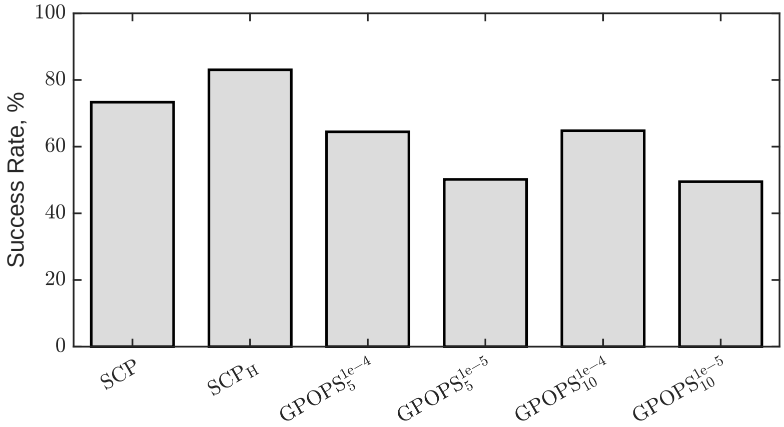


(b) Comparison of mean CPU time.

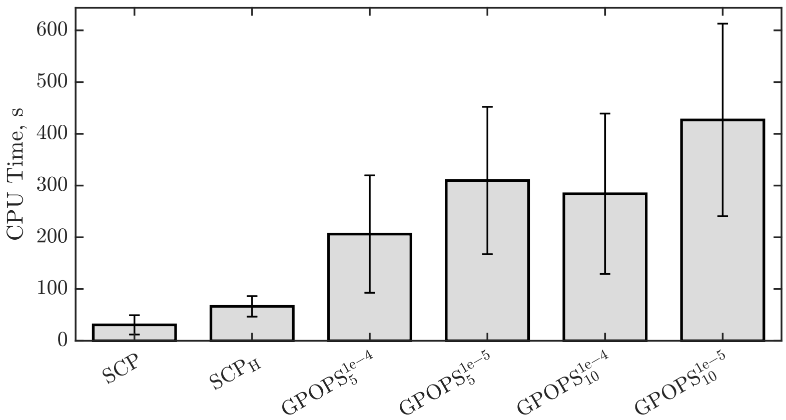


(c) Comparison of mean final mass.

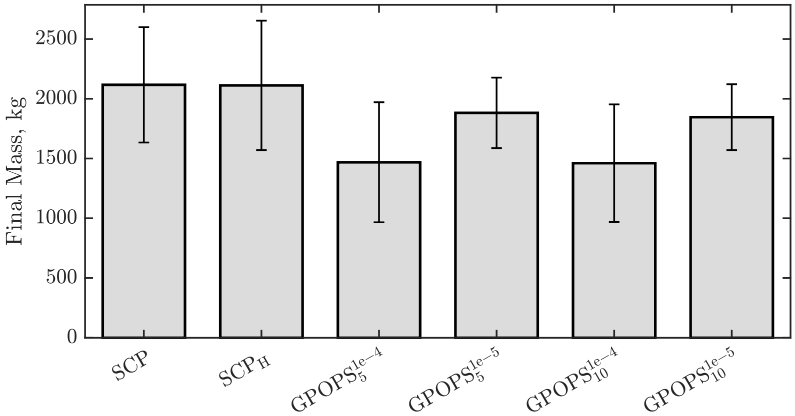
Figure 3: Comparison of SCP, SCP$_\mathrm{H}$ and GPOPS-II in terms of convergence, CPU time, and final mass for the 2000 SG344 transfer. Mean values are shown, the error bars refer to the standard deviation $\pm 1\sigma$.



(a) Comparison of success rate.

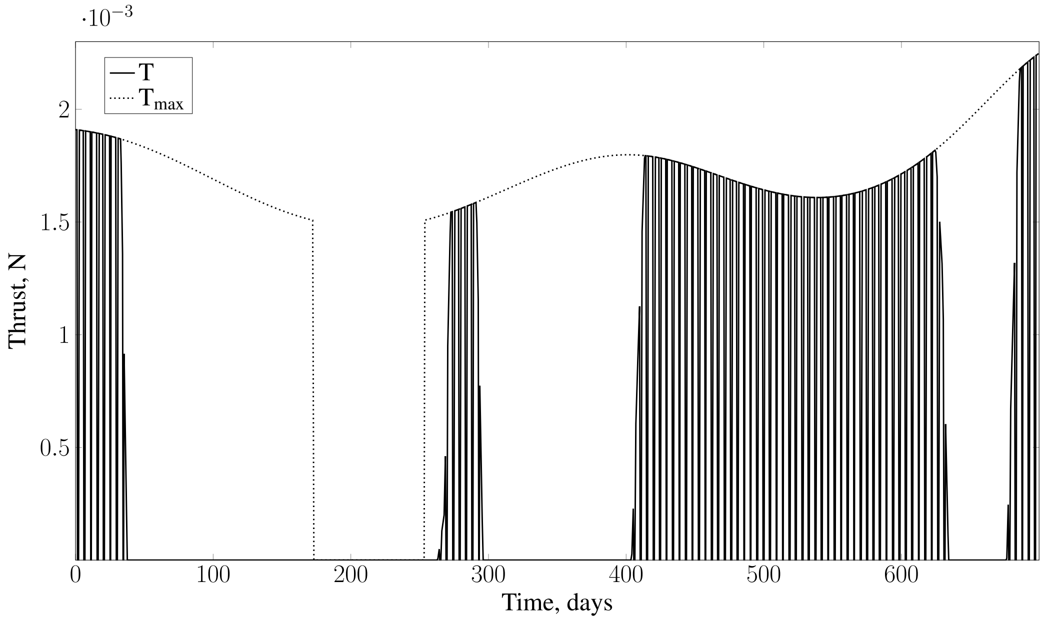


(b) Comparison of mean CPU time.

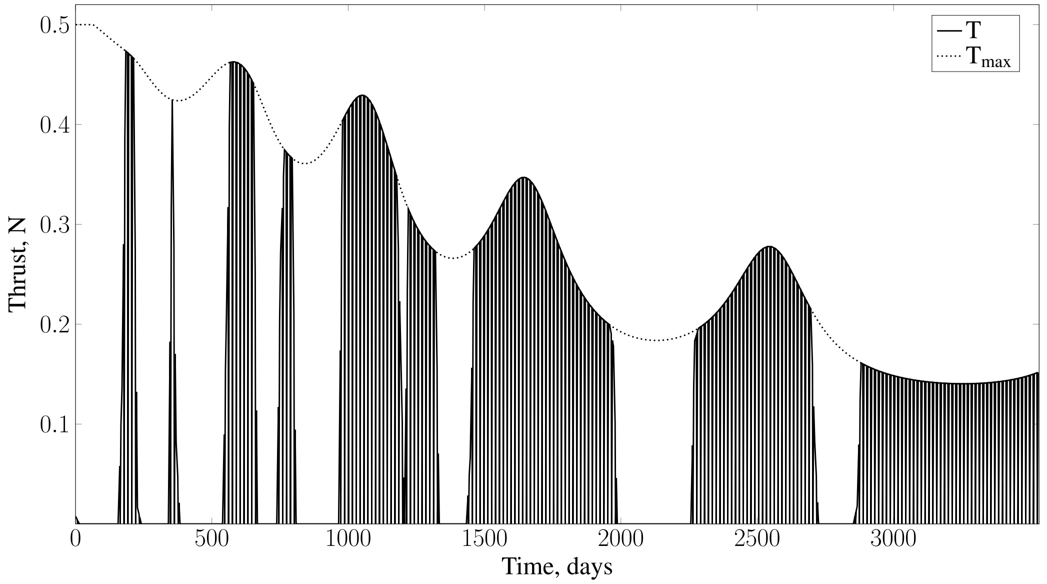


(c) Comparison of mean final mass.

Figure 4: Comparison of SCP, SCP$_\mathrm{H}$ and GPOPS-II in terms of convergence, CPU time, and final mass for the Dionysus transfer. Mean values are shown, the error bars refer to the standard deviation $\pm 1\sigma$.

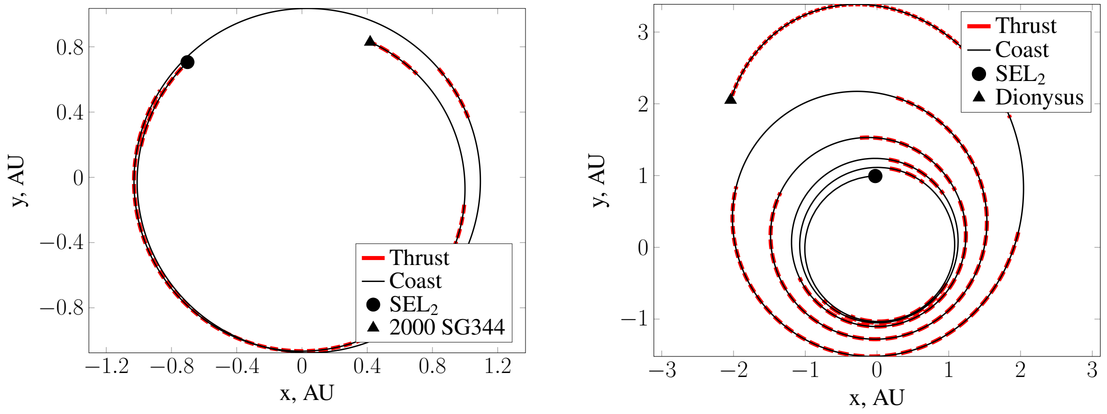


(a) Thrust profile for transfer to 2000 SG344.



(b) Thrust profile for transfer to Dionysus.

Figure 5: Typical thrust profiles for the transfers to 2000 SG344 and Dionysus when no-thrust periods are considered.



(a) Trajectory for 2000 SG344 transfer.

(b) Trajectory for Dionysus transfer.

Figure 6: Typical trajectories for transfers to 2000 SG344 and Dionysus when no-thrust periods are considered.

this work can be considered another step toward computing more mission-compliant trajectories using convex programming techniques.

REFERENCES

[1] T. Reynolds, D. Malyuta, M. Mesbahi, B. Açıkmeşe, and J. M. Carson, "A Real-Time Algorithm for Non-Convex Powered Descent Guidance," *AIAA Scitech 2020 Forum*, 2020, 10.2514/6.2020-0844.

[2] M. B. Quadrelli, L. J. Wood, J. E. Riedel, M. C. McHenry, M. Aung, L. A. Cangahuala, R. A. Volpe, P. M. Beauchamp, and J. A. Cutts, "Guidance, Navigation, and Control Technology Assessment for Future Planetary Science Missions," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 7, 2015, pp. 1165–1186, 10.2514/1.G000525.

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

[4] Y. Mao, M. Szmuk, and B. Açıkmeşe, "A Tutorial on Real-time Convex Optimization Based Guidance and Control for Aerospace Applications," *2018 Annual American Control Conference (ACC)*, 2018, pp. 2410–2416, 10.23919/ACC.2018.8430984.

[5] C. Hofmann and F. Topputo, "Rapid Low-Thrust Trajectory Optimization in Deep Space Based On Convex Programming," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 7, 2021, pp. 1379–1388, 10.2514/1.G005839.

[6] Y. Kayama, K. C. Howell, M. Bando, and S. Hokamoto, "Low-Thrust Trajectory Design with Convex Optimization for Libration Point Orbits," *AAS/AIAA Space Flight Mechanics Meeting*, February 2021. AAS Paper 21-231.

[7] C. Hofmann and F. Topputo, "Pseudospectral Convex Low-Thrust Trajectory Optimization in a High-Fidelity Model," *AAS/AIAA Astrodynamics Specialist Conference*, August 2021. AAS Paper 21-678.

[8] Z. Wang and M. J. Grant, "Minimum-Fuel Low-Thrust Transfers for Spacecraft: A Convex Approach," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 54, No. 5, 2018, pp. 2274–2290, 10.1109/TAES.2018.2812558.

[9] B. Pan, P. Lu, X. Pan, and Y. Ma, "Double-Homotopy Method for Solving Optimal Control Problems," *Journal of Guidance, Control, and Dynamics*, Vol. 39, Jun. 2016, pp. 1706–1720, 10.2514/1.G001553.

[10] F. Jiang, H. Baoyin, and J. Li, "Practical Techniques for Low-Thrust Trajectory Optimization with Homotopic Approach," *Journal of Guidance, Control, and Dynamics*, Vol. 35, Aug. 2012, pp. 245–258, 10.2514/1.52476.

[11] F. Jia, D. Qiao, H. Han, and X. Li, "Efficient optimization method for variable-specific-impulse low-thrust trajectories with shutdown constraint," *Science China Technological Sciences*, Vol. 65, No. 3, 2022, p. 581–594, 10.1007/s11431-021-1949-0.

[12] B. Açıkmeşe and S. R. Ploen, “Convex Programming Approach to Powered Descent Guidance for Mars Landing,” *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 5, 2007, pp. 1353–1366, 10.2514/1.27553.

[13] W. J. Rugh, *Linear System Theory*. Prentice-Hall, 2 ed., 1996.

[14] D. P. Lubey and D. J. Scheeres, “Identifying and Estimating Mismodeled Dynamics via Optimal Control Policies and Distance Metrics,” *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 5, 2014, pp. 1512–1523, 10.2514/1.G000369.

[15] C. Acton, N. Bachman, B. Semenov, and E. Wright, “A look towards the future in the handling of space science mission geometry,” *Planetary and Space Science*, Vol. 150, 2018, pp. 9–12. Enabling Open and Interoperable Access to Planetary Science and Heliophysics Databases and Tools, 10.1016/j.pss.2017.02.013.

[16] F. Topputo, Y. Wang, G. Giordano, V. Franzese, H. Goldberg, F. Perez-Lissi, and R. Walker, “Envelop of Reachable Asteroids by M-ARGO CubeSat,” *Advances in Space Research*, Vol. 67, No. 12, 2021, pp. 4193–4221, 10.1016/j.asr.2021.02.031.

[17] J. M. Carson and B. Açıkmeşe, “A Model Predictive Control Technique with Guaranteed Resolvability and Required Thruster Silent Times for Small-Body Proximity Operations,” *AIAA Guidance, Navigation and Control Conference and Exhibit*, 8 2006. AIAA Paper 2006-6780.

[18] M. A. Patterson and A. V. Rao, “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.

[19] P. E. Gill, W. Murray, and M. A. Saunders, “SNOPT: An SQP algorithm for large-scale constrained optimization,” *SIAM review*, Vol. 47, No. 1, 2005, pp. 99–131.

[20] A. Domahidi, E. Chu, and S. Boyd, “ECOS: An SOCP Solver for Embedded Systems,” *European Control Conference*, Zurich, Switzerland, 2013, pp. 3071–3076, doi: 10.23919/ECC.2013.6669541.

[21] E. Taheri, I. Kolmanovsky, and E. Atkins, “Enhanced Smoothing Technique for Indirect Optimization of Minimum-Fuel Low-Thrust Trajectories,” *Journal of Guidance, Control, and Dynamics*, Vol. 39, No. 11, 2016, 10.2514/1.G000379.

[22] E. Taheri, J. L. Junkins, I. Kolmanovsky, and A. Girard, “A Novel Approach for Optimal Trajectory Design With Multiple Operation Modes of Propulsion System, Part 1,” *Acta Astronautica*, Vol. 172, 2020, pp. 151–165, 10.1016/j.actaastro.2020.02.042.

[23] C. Hofmann, A. C. Morelli, and F. Topputo, “On the Performance of Discretization and Trust-Region Methods for On-Board Convex Low-Thrust Trajectory Optimization,” *AIAA Scitech 2020 Forum*, 2022, 10.2514/6.2022-1892.