(Preprint) AAS 21-678

PSEUDOSPECTRAL CONVEX LOW-THRUST TRAJECTORY OPTIMIZATION IN A HIGH-FIDELITY MODEL

Christian Hofmann* **and Francesco Topputo†**

The low-thrust minimum-fuel trajectory optimization problem is solved in a high-fidelity model using convex optimization. The problem is convexified and discretized with the Radau pseudospectral method. We apply a homotopic approach and successively increase the complexity and accuracy of the model to enhance convergence properties. Solar radiation pressure, variable specific impulse and maximum thrust, and the perturbation of other bodies are considered. A homotopy from the minimum-energy to the minimum-fuel problem is added. The performance is assessed in minimum-fuel interplanetary transfers to two asteroids. Moreover, the developed approach is integrated into a closed-loop guidance simulation. The numerical simulations show that the sequential convex programming approach is able to solve highly nonlinear optimization problems to high accuracy with poor initial guesses.

INTRODUCTION

Highly efficient propulsion systems such as electric propulsion have become an important development during the past years. Even though they provide only low thrust, their high specific impulse enables considerable savings of propellant. This enables new space missions, but also requires new trajectory design techniques as the thrusting periods increase and therefore, the assumption of instantaneous impulses does not hold anymore. Although extensive research has been conducted in the field of low-thrust trajectory optimization,[1] it was only recently that autonomous guidance and control has attracted more and more attention.[2,3] The current state of the art is to determine trajectories in an open-loop fashion on ground and update the state and control histories of the spacecraft accordingly. This approach, however, results in high operational costs and is therefore not consistent with the overall goal to reduce the cost of future missions.[3] Moreover, it limits the mission design and can pose risks as the dependency on a communication link to Earth might impede time-critical acting during a mission. It is therefore desirable to limit the interaction with Earth to a minimum and shift the decision making on board. It is expected that this is the only way to enable a robust and sustainable exploration and exploitation of space.[4]

Planning the trajectory requires solving a nonlinear optimal control problem (OCP). In contrast to indirect methods that suffer from convergence issues when a poor initial guess is provided,[5] direct methods are often preferred, especially when high-fidelity models are used.[6] The infinite-dimensional problem is discretized into a finite-dimensional nonlinear parameter optimization problem. There are several approaches to solve the resulting nonlinear program (NLP), ranging from

*Ph.D. Candidate, Department of Aerospace Science and Technology, Politecnico di Milano, Via La Masa 34, 20156 Milan, Italy. Email: christian.hofmann@polimi.it

†Full Professor, Department of Aerospace Science and Technology, Politecnico di Milano, Via La Masa 34, 20156 Milan, Italy. Email: francesco.topputo@polimi.it

single and multiple shooting to direct collocation.[5] As solving the NLP directly often requires high computational power and a decent initial guess, sequential convex programming (SCP) techniques have become a popular alternative in the past few years.[7] The problem is convexified and then iteratively solved until the solution satisfies the original nonlinear constraints. Many applications, such as powered descent landing,[8] nonlinear model predictive control,[9] collision avoidance and robotics,[10] have successfully applied convex programming techniques. The use in low-thrust trajectory optimization has also become more popular. For example, the minimum-time- and minimum-fuel problems have recently been solved with a convex approach.[11,12] Tang et al. used SCP to generate an initial guess for an indirect method.[13] More complex interplanetary trajectories have been computed with an adaptive pseudospectral method.[14] Moreover, Hofmann et al. assessed the real-time performance of a SCP method on an ARM processor for interplanetary transfers.[15]
Low-thrust trajectory optimization problems that consider higher fidelity models (for example, third body perturbations, solar radiation pressure, realistic thruster model) have been solved with indirect ([16,17]) and direct, NLP-based methods ([18–20]). With regard to convex programming techniques, the accuracy and convergence can be poor for highly nonlinear applications due to the successive linearization. Therefore, most researchers considered only simple two-body dynamics. The literature on solving more complex models, for example the circular-restricted three-body problem using convex programming, is scarce.[21] Moreover, none of the previous works addressed additional perturbations such as solar radiation pressure or the gravitational forces of other bodies. Even though simple two-body dynamics suffice for many applications (especially preliminary design), the error can be large in several cases. An example is ESA's M-ARGO mission to an asteroid where the spacecraft departs from the Sun-Earth Lagrange point L_2. It was shown that neglecting the gravitational influence of the Earth results in different final masses and a large error during the first part of the trajectory when the spacecraft is still close to Earth.[22,23] Even though highly accurate models are often desirable, the computational effort can increase significantly. This becomes even more challenging for real-time applications with limited computational power; a tradeoff is necessary and simpler models are often preferred. In this work, we exploit the advantages of convex programming approaches and incorporate more complex dynamical models. A homotopy is performed to gradually increase the complexity of the model. Third-body perturbations, solar radiation pressure and thruster models with a variable specific impulse and maximum thrust are considered. We use an adaptive Radau pseudospectral method to solve the resulting optimization problem and run a multitude of simulations to assess the effectiveness of the approach. The high-fidelity model is incorporated into a closed-loop guidance simulation to assess the potential onboard capability.
The paper is structured as follows. Section II describes the optimal control problem and the adaptive Radau discretization method. In Section III, the homotopy process to more complex models is explained. Section IV addresses the closed-loop guidance simulation and Section V presents the results of the numerical simulations. Section VI concludes this paper.

PROBLEM FORMULATION

We present the optimal control problem in space flight and explain the convexification to obtain a convex subproblem. The adaptive Radau pseudospectral method is used to discretize the infinite-dimensional OCP.

Optimal Control Problem and Convexification

Considering a two-body model with the Sun as the primary, the equations of motion in Cartesian coordinates are given by

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \Rightarrow \begin{bmatrix} \dot{\mathbf{r}} \\ \dot{\mathbf{v}} \\ \dot{m} \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ -\mu \mathbf{r}/r^3 + \mathbf{T}/m \\ -\left\|\mathbf{T}\right\|/(g_0 I_{\text{sp}}) \end{bmatrix} \tag{1}$$

where $\mathbf{r}$, $\mathbf{v}$, and m are the position, velocity and mass, respectively. μ is the gravitational constant, $\mathbf{T}$ is the thrust vector, g_0 is the gravitational acceleration at sea level and I_{sp} the specific impulse. $\|\cdot\|$ refers to the 2-norm if not stated otherwise. The fuel-optimal control problem in space flight is to minimize

$$J_0 = -m(t_f) \tag{2}$$

subject to

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{3a}$$

$$0 \leq \|\mathbf{T}\| \leq T_{\text{max}} \tag{3b}$$

$$\mathbf{x}(t_0) = \mathbf{x}_0, \quad \mathbf{x}(t_f) = \mathbf{x}_f \tag{3c}$$

$$\mathbf{x}_l \leq \mathbf{x} \leq \mathbf{x}_u, \quad \mathbf{u}_l \leq \mathbf{u} \leq \mathbf{u}_u \tag{3d}$$

The subscripts $(\cdot)_0$ and $(\cdot)_f$ refer to initial and final values, respectively. Eq. (3d) defines the lower (subscript l) and upper bounds (subscript u) for states and controls, respectively. Introducing the new variable T and defining the states and controls as $\mathbf{x} = [\mathbf{r}, \mathbf{v}, m]^\top$ and $\mathbf{u} = [\mathbf{T}, T]^\top$, we rewrite Eq. (3) to obtain

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\mathbf{r}} \\ \dot{\mathbf{v}} \\ \dot{m} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{v} \\ -\mu \mathbf{r}/r^3 \\ 0 \end{bmatrix}}_{\mathbf{p}(\mathbf{x})} + \underbrace{\begin{bmatrix} \mathbf{0}_{3\times 4} \\ \mathbf{1}_{3\times 3}\ 1/m \quad \mathbf{0}_{3\times 1} \\ \mathbf{0}_{1\times 3} \quad -1/(g_0 I_{\text{sp}}) \end{bmatrix}}_{\mathbf{B}(\bar{\mathbf{x}})} \begin{bmatrix} \mathbf{T} \\ T \end{bmatrix} = \mathbf{p}(\mathbf{x}) + \mathbf{B}(\mathbf{x})\mathbf{u} \tag{4}$$

where $\mathbf{1}$ denotes the identity matrix. We set $\mathbf{f}(\mathbf{x}, \mathbf{u}) \approx \mathbf{p}(\mathbf{x}) + \mathbf{B}(\bar{\mathbf{x}})\mathbf{u}$ and linearize Eq. (4) only partially at $\bar{\mathbf{x}}$ so that the dynamics are independent of the previous control $\bar{\mathbf{u}}$:

$$\mathbf{f}(\mathbf{x}, \mathbf{u}) \approx \mathbf{A}(\bar{\mathbf{x}})\mathbf{x} + \mathbf{B}(\bar{\mathbf{x}})\mathbf{u} + \mathbf{q}(\bar{\mathbf{x}}) \tag{5}$$

Numerical simulations show that this technique can improve the convergence considerably.[24] $\mathbf{A}(\bar{\mathbf{x}}) = \left.\frac{\partial \mathbf{p}}{\partial \mathbf{x}}\right|_{\mathbf{x}=\bar{\mathbf{x}}}$ denotes the Jacobian matrix evaluated at the reference point $\bar{\mathbf{x}}$ and $\mathbf{q}(\bar{\mathbf{x}}) = \mathbf{p}(\bar{\mathbf{x}}) - \mathbf{A}(\bar{\mathbf{x}})\bar{\mathbf{x}}$ refers to the constant part of the linearization.
The second-order cone program then reads:

$$\min \quad -m(t_f) + \lambda \left\|\boldsymbol{\nu}\right\|_1 \tag{6}$$

subject to

$$\dot{\mathbf{x}} = \mathbf{A}(\bar{\mathbf{x}})\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{q}(\bar{\mathbf{x}}) + \boldsymbol{\nu} \tag{7a}$$

$$\|\mathbf{T}\| \leq T \tag{7b}$$

$$0 \leq T \leq T_{\text{max}} \tag{7c}$$

$$\|\mathbf{x} - \bar{\mathbf{x}}\|_1 \leq r_{\text{tr}} \tag{7d}$$

$$\mathbf{x}(t_0) = \mathbf{x}_0, \quad \mathbf{x}(t_f) = \mathbf{x}_f \tag{7e}$$

$$\mathbf{x}_l \leq \mathbf{x} \leq \mathbf{x}_u, \quad \mathbf{u}_l \leq \mathbf{u} \leq \mathbf{u}_u \tag{7f}$$

Note that we added a virtual control $\boldsymbol{\nu}$ in Eq. (7a) to avoid *artificial infeasibility* that may arise due to the linearization. To ensure that this slack parameter is only active when needed, we penalize it in the cost function with a sufficiently large parameter λ. Eq. (7b) is the relaxed version of the thrust constraint $\|\mathbf{T}\| = T$. Apparently, a feasible solution that satisfies $\|\mathbf{T}\| = T$ is also a feasible solution of the relaxed problem. Moreover, it can be shown that the optimal solution of the relaxed problem is also an optimal solution of the original problem, and we have $\|\mathbf{T}\| = T$ and $T = 0$ or $T = T_{\text{max}}$.[25] The trust-region constraint in Eq. (7d) is needed to keep the linearization valid.

Radau Pseudospectral Method

Pseudospectral methods offer several favorable properties, including spectral convergence for smooth problems and the possibility to determine the costates using the obtained Lagrange multipliers.[26] As interplanetary trajectories often last several years, many discretization nodes might be needed to accurately approximate the states and controls. As a consequence, using a single interpolating polynomial of high degree would result in a dense problem that requires large computational cost. In this paper, we apply an adaptive method where the trajectory is divided into several segments and the states and controls are approximated with polynomials of different degrees (and therefore, arbitrary number of nodes). Instead of approximating the dynamics at the roots of the flipped Legendre–Radau polynomial,[14,27] we use the standard Legendre–Radau points that are defined in the pseudospectral time domain $\hat{t}_i^{(k)} \in [-1, 1)$. Therefore, the final point of each segment is not a collocation point. Still, we include it in the state approximation in this work. With regard to the controls, note that the final control at t_f is not obtained in the solution. The relationship between the physical time t_i and pseudospectral time $\hat{t}_i$ is given by[28]

$$t_i^{(k)} = t_{N_k}^{(k-1)} + \frac{t_{N_k}^{(k)} - t_0^{(k)}}{2}\hat{t}_i^{(k)} + \frac{t_{N_k}^{(k)} + t_0^{(k)}}{2} \qquad \text{for } i = 0, 1, ..., N_k, \quad k = 1, ..., K \tag{8}$$

with $t_{N_k}^{(0)} = 0$. $(\cdot)_i^{(k)}$ refers to the ith node of the kth segment. Given $N_k - 1$ collocation points and defining the discretized states and controls as $\mathbf{X}$ and $\mathbf{U}$, respectively, the dynamics are approximated as[29]

$$\begin{aligned} \sum_{j=0}^{N_k} D_{ij}^{(k)} \mathbf{X}_j^{(k)} &= \frac{t_{N_k}^{(k)} - t_0^{(k)}}{2} \mathbf{f}(\mathbf{X}_i^{(k)}, \mathbf{U}_i^{(k)}) \\ &= \frac{t_{N_k}^{(k)} - t_0^{(k)}}{2} \left[\mathbf{A}(\bar{\mathbf{X}}_{\mathbf{i}}^{(\mathbf{k})}) \mathbf{X}_i^{(k)} + \mathbf{B}(\bar{\mathbf{X}}_{\mathbf{i}}^{(\mathbf{k})}) \mathbf{U}_i^{(k)} + \mathbf{q}(\bar{\mathbf{X}}_i^{(k)}) + \boldsymbol{\nu}_i^{(k)} \right] \end{aligned} \tag{9}$$

with the differentiation matrix $\mathbf{D}$ and $i = 0, 1, ..., N_k - 1$. As the final points are not collocation points, the condition $\mathbf{X}_{N_k}^{(k)} = \mathbf{X}_0^{(k+1)}$ must be satisfied for all segments $k < K$. This constraint is implicitly taken into account as the final node of a segment and the initial node of the next segment are considered the same. Defining $\mathbf{Y} \equiv [\mathbf{X}, \mathbf{U}, \boldsymbol{\nu}]^\top$, the discretized convex dynamics can be written in standard form $\mathbf{M}\,\mathbf{Y} = \mathbf{b}$ where

$$\mathbf{M} = \left[\text{diag}(\hat{\mathbf{D}}^{(k)}),\ \text{diag}(\hat{\mathbf{B}}^{(k)}),\ \mathbf{1} \right] \tag{10}$$

with the diagonal entries

$$\hat{\mathbf{D}}^{(k)} = \begin{bmatrix} D_{00}^{(k)}\mathbf{1}_{n_x} - \Delta^{(k)}\mathbf{A}_0^{(k)} & D_{01}^{(k)}\mathbf{1}_{n_x} & \cdots & D_{0N_k}^{(k)}\mathbf{1}_{n_x} \\ D_{10}^{(k)}\mathbf{1}_{n_x} & D_{11}^{(k)}\mathbf{1}_{n_x} - \Delta^{(k)}\mathbf{A}_1^{(k)} & \cdots & D_{1N_k}^{(k)}\mathbf{1}_{n_x} \\ \vdots & \vdots & & \vdots \\ D_{N_k-1,1}^{(k)}\mathbf{1}_{n_x} & \cdots & & D_{N_k-1,N_k}^{(k)}\mathbf{1}_{n_x} - \Delta^{(k)}\mathbf{A}_{N_k-1}^{(k)} \end{bmatrix} \tag{11}$$

and

$$\hat{\mathbf{B}}^{(k)} = \begin{bmatrix} -\Delta^{(k)}\mathbf{B}_0^{(k)} & & & \mathbf{0} \\ & -\Delta^{(k)}\mathbf{B}_1^{(k)} & & \\ & & \ddots & \\ \mathbf{0} & & & -\Delta^{(k)}\mathbf{B}_{N_k-1}^{(k)} \end{bmatrix} \tag{12}$$

The notation

$$\Delta^{(k)} = \frac{t_{N_k}^{(k)} - t_0^{(k)}}{2} \tag{13}$$

and $\mathbf{A}_j^{(k)} = \mathbf{A}(\bar{\mathbf{X}}_j^{(k)})$ ($\mathbf{B}$ and $\mathbf{q}$ are defined similarly) was introduced for conciseness. $n_x \in \mathbb{R}^{7\times 1}$ is the number of states.

LOW-THRUST TRAJECTORY OPTIMIZATION IN A HIGH-FIDELITY MODEL

As adding high-fidelity models significantly increases the nonlinearity, a standard SCP algorithm would have difficulties to converge for complex problems. Therefore, we perform a homotopy-like procedure and gradually increase the complexity of the problem. The following steps can combined arbitrarily depending on the considered model:

- If needed, a homotopy from the smooth minimum-energy to the non-smooth minimum-fuel problem is performed.
- We gradually add the perturbing force caused by solar radiation pressure. As it is relatively small, our rationale is that the solution will not change significantly; only a small acceleration is added to the dynamics. The solution of the previous run is used as the initial guess to obtain a refined solution in only few iterations.
- The real maximum thrust and specific impulse curves are approximated with piecewise affine functions. The solution to this simplified problem is then used as the initial guess to solve the problem with the real thruster model.
- Solving the full n-body problem would deteriorate the convergence as the dynamics are highly nonlinear. Therefore, the value of the gravitational parameter of the desired perturbing body is gradually increased from 0 to the final value. This way, the nonlinearity increases only slightly in each iteration and convergence is not impeded. We successively add other bodies and proceed in a similar way.

Minimum-Energy to Minimum-Fuel Homotopy

For the minimum-energy to minimum-fuel homotopy, the control structure is relaxed and the performance index changes to[30]

$$J_0 = \int_{t_0}^{t_f} \left[T - \epsilon_{\text{energy}} T\left(1-T\right)\right] \, \mathrm{d}t = \int_{t_0}^{t_f} T\left(1-\epsilon_{\text{energy}}\right) \, \mathrm{d}t + \int_{t_0}^{t_f} \epsilon_{\text{energy}} T^2 \, \mathrm{d}t \tag{14}$$

where $\epsilon_{\text{energy}} \in [0,1]$ is a smoothing parameter. The smooth energy-optimal problem ($\epsilon_{\text{energy}} = 1$) is solved first and ϵ is gradually decreased until the fuel-optimal problem ($\epsilon_{\text{energy}} = 0$) is obtained.
Remark: Minimizing $\int_{t_0}^{t_f} T \, \mathrm{d}t$ is equivalent to Eq. (2).
Even though the objective function in Eq. (14) is convex, we need to reformulate the quadratic part in standard form. The discretized objective is

$$J_0 = \left(1-\epsilon_{\text{energy}}\right) \sum_{i}^{N} \Delta_i U_{T,i} w_i + \underbrace{\mathbf{U}_T^\top \mathbf{Q} \mathbf{U}_T}_{J_{\text{quad}}} \tag{15}$$

where N is the number of collocation points and $\mathbf{Q}$ is a diagonal matrix with entries

$$Q_{ii} = \epsilon_{\text{energy}} \Delta_i w_i \tag{16}$$

Δ_i, $U_{T,i}$ and w_i denote the ith entries of the concatenated time transformation factors $\mathbf{\Delta}$ [see Eq. (13)], thrust magnitudes $\mathbf{U}_T$ and Radau integration weights $\mathbf{w}$, respectively. We introduce the slack variable $y \in \mathbb{R}^1$ to rewrite the quadratic part of Eq. (15) as[31]

$$J_{\text{quad}} = y \tag{17}$$

with the additional constraint $\mathbf{U}_T^\top \mathbf{Q} \mathbf{U}_T \leq y$. This epigraph form is equivalent to minimizing $\mathbf{U}_T^\top \mathbf{Q} \mathbf{U}_T$ directly. As the matrix $\mathbf{Q}$ is positive definite, the Cholesky decomposition $\mathbf{Q} = \mathbf{L}^\top \mathbf{L}$ is used to reformulate this constraint as $\mathbf{z}^\top \mathbf{z} \leq y$ and $\mathbf{z} = \mathbf{L}^\top \mathbf{U}_T$, $\mathbf{z}$ being a slack variable. The quadratic constraint can now be reformulated as a second-order cone constraint and the optimization problem for the energy-to-fuel homotopy reads

$$\min_{U_T, y, \boldsymbol{\nu}} \quad (1-\epsilon) \sum_{i}^{N} \Delta_i U_{T,i} w_i + y + \lambda \left\| \boldsymbol{\nu} \right\|_1 \tag{18}$$

subject to

$$\left\| \begin{bmatrix} 2\mathbf{L}^\top \mathbf{U}_T \\ y - 1 \end{bmatrix} \right\| \leq y + 1 \tag{19}$$

and the discretized form of Eq. (7).

Remark: Due to the structure of $\mathbf{Q}$, the diagonal entries L_{ii} of $\mathbf{L}^\top$ reduce to $L_{ii} = \sqrt{\epsilon_{\text{energy}} \Delta_i w_i}$.

Solar Radiation Pressure

The equations of motion considering the solar radiation pressure are given as

$$\begin{bmatrix} \dot{\mathbf{r}} \\ \dot{\mathbf{v}} \\ \dot{m} \end{bmatrix}_{\text{SRP}} = \mathbf{f} + \begin{bmatrix} \mathbf{0} \\ \mathbf{a} \\ 0 \end{bmatrix}_{\text{SRP}} \tag{20}$$

where $\mathbf{f}$ refers to the remaining part of the dynamics. Moreover,

$$\mathbf{a}_{\text{SRP}} = \frac{S_{\text{Sun}} C_R A_{\text{SC}}}{m} \frac{\mathbf{r}}{r^3} \tag{21}$$

$$S_{\text{Sun}} = \frac{L_{\text{Sun}}}{4\pi c} \tag{22}$$

$$L_{\text{Sun}} = 4\pi C_{\text{Sun}} \text{AU}^2 \tag{23}$$

A simple cannonball model is used where the projected area A_{SC} of the spacecraft is assumed constant.[32] 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.

We introduce the homotopic parameter ϵ_{SRP} to gradually add the acceleration caused by the solar radiation pressure. Eq. (20) then reads

$$\begin{bmatrix} \dot{\mathbf{r}} \\ \dot{\mathbf{v}} \\ \dot{m} \end{bmatrix}_{\text{SRP}} = \mathbf{f} + \epsilon_{\text{SRP}} \begin{bmatrix} \mathbf{0} \\ \mathbf{a} \\ 0 \end{bmatrix}_{\text{SRP}} \tag{24}$$

The problem is first solved with $\epsilon_{\text{SRP}} = 0$ (SRP is not taken into account) and gradually increased until $\epsilon_{\text{SRP}} = 1$ (full influence of SRP is considered). The solution of the previous optimization problem is used as the initial guess for the new one. Eq. (24) is then partially linearized as described in Section II. Even though the additional $\mathbf{a}_{\text{SRP}}$ increases the nonlinearity, this smoothing approach is a simple means to improve convergence in SCP as the perturbation is only added step by step.

Variable Maximum Thrust and Specific Impulse

In a real thruster model, the maximum thrust $T_{\max}(P_{\text{in}}(r))$ and specific impulse $I_{\text{sp}}(P_{\text{in}}(r))$ depend on the input power $P_{\text{in}}(r)$, which in turn is a function of the distance to the Sun. These functions are often described by nth order polynomials. This increases complexity and results in additional nonlinear constraints. Without loss of generality, we consider two different thruster models in this work (see Figs. 1 and 2a):

Model 1:[17]

$$T_{\max,1}(r) = \frac{\hat{a}_0}{r} \frac{\hat{a}_1 + \frac{\hat{a}_2}{r} + \frac{\hat{a}_3}{r^2}}{1 + \hat{a}_4 r + \hat{a}_5 r^2} \tag{25a}$$

$$I_{\text{sp},1}(r) = \frac{\hat{b}_0}{r} \frac{\hat{b}_1 + \frac{\hat{b}_2}{r} + \frac{\hat{b}_3}{r^2}}{1 + \hat{b}_4 r + \hat{b}_5 r^2} \tag{25b}$$

with $\hat{a}_0 = 0.3375$ N, $\hat{a}_1 = 1.621$ AU, $\hat{a}_2 = -0.108$ AU^2, $\hat{a}_3 = -0.0117$ AU^3, $\hat{a}_4 = 0.0108$ 1/AU, $\hat{a}_5 = -0.0013$ 1/AU^2 and $\hat{b}_0 = 3375$ s, $\hat{b}_1 = 1.621$ AU, $\hat{b}_2 = -0.108$ AU^2, $\hat{b}_3 = -0.0117$ AU^3, $\hat{b}_4 = 0.0108$ 1/AU, $\hat{b}_5 = -0.0013$ 1/AU^2 and r in AU. The coefficients of the real thruster model are modified values of the ones used in Taheri et al.[17]

Model 2:[20]

$$T_{\max,2}(P_{\text{in}}) = \tilde{a}_0 + \tilde{a}_1 P_{\text{in}} + \tilde{a}_2 P_{\text{in}}^2 + \tilde{a}_3 P_{\text{in}}^3 + \tilde{a}_4 P_{\text{in}}^4 \tag{26a}$$

$$I_{\text{sp},2}(P_{\text{in}}) = \tilde{b}_0 + \tilde{b}_1 P_{\text{in}} + \tilde{b}_2 P_{\text{in}}^2 + \tilde{b}_3 P_{\text{in}}^3 + \tilde{b}_4 P_{\text{in}}^4 \tag{26b}$$

$$P_{\text{in},2}(r) = \tilde{c}_0 + \tilde{c}_1 r + \tilde{c}_2 r^2 + \tilde{c}_3 r^3 + \tilde{c}_4 r^4 \tag{26c}$$

with $\tilde{a}_0 = -0.7253$ mN, $\tilde{a}_1 = 0.02481$ mN/W, $\tilde{a}_2 = 0$, $\tilde{a}_3 = 0$, $\tilde{a}_4 = 0$, $\tilde{b}_0 = 2652$ s, $\tilde{b}_1 = -18.123$ s/W, $\tilde{b}_2 = 0.3887$ s/W^2, $\tilde{b}_3 = -0.00174$ s/W^3 and $\tilde{c}_0 = 840.11$ W, $\tilde{c}_1 = -1754.3$ W/AU, $\tilde{c}_2 = 1625.01$ W/AU^2, $\tilde{c}_3 = -739.87$ W/AU^3, $\tilde{c}_4 = 134.45$ W/AU^4.

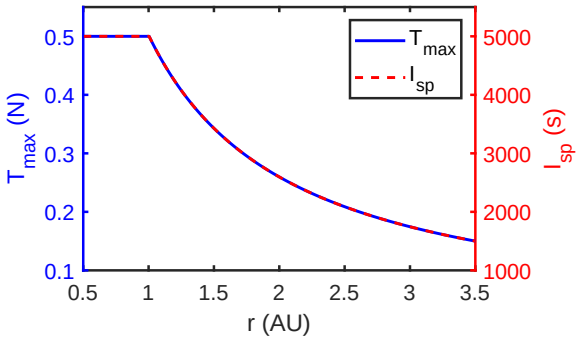
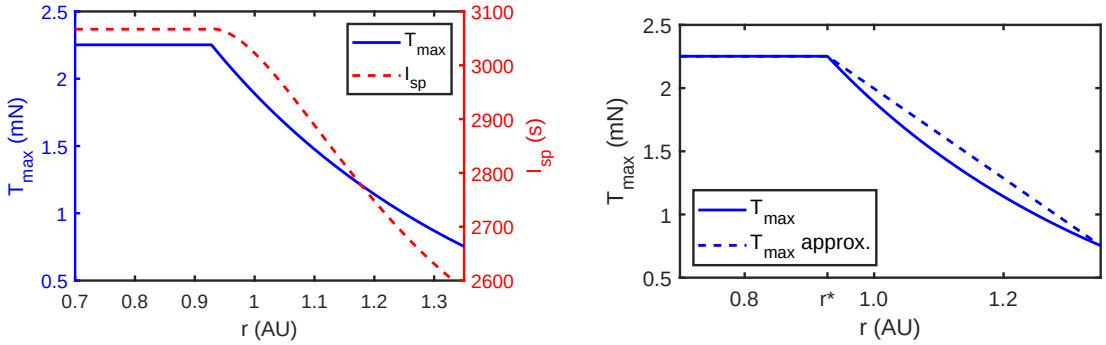


Figure 1: $T_{\text{max},1}(r)$ and $I_{\text{sp},1}(r)$ curves.



(a) $T_{\text{max},2}(r)$ and $I_{\text{sp},2}(r)$ curves.

(b) Approximation of $T_{\text{max},2}(r)$.

Figure 2: Thruster model 2 and its approximation with an affine function.

As the input power depends on the distance to the Sun (note the similarity to a r^{-2} power law), we propose to approximate the real thruster model with piecewise affine functions at first. Approximating $T_{\max}$ as

$$T_{\max}(r) \approx ar + b \tag{27}$$

with constants a and b, and substituting it into Eq. (3b) yields

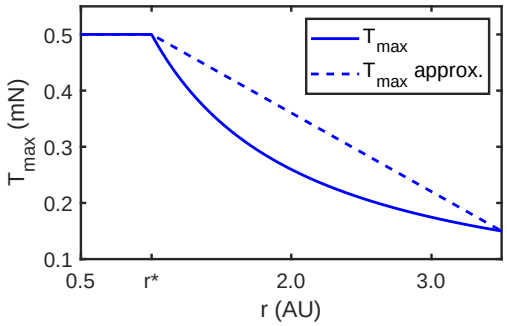
$$T - ar - b \leq 0 \tag{28}$$

As the slope a is always negative in our examples, we use $\|\mathbf{r}\|_2 = r$ to rewrite Eq. (28) to obtain the second-order cone constraint

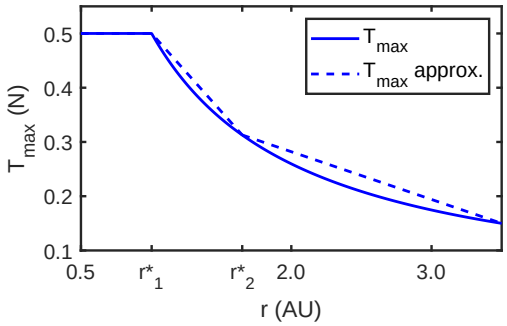
$$\|\mathbf{r}\|_2 \leq \frac{b}{|a|} - \frac{1}{|a|}T \tag{29}$$

This enhances the convergence properties as the constraint

$$T \leq T_{\max}(r) \iff T \leq T_{\max}(P_{\text{in}}) \tag{30}$$



(a) Approximation of $T_{\text{max},1}(r)$ using an affine function.



(b) Approximation of $T_{\text{max},1}(r)$ using a piecewise affine function.

Figure 3: $T_{\text{max},1}(r)$ approximations using an affine and piecewise affine function.

is now convex and no additional slack variables or linearizations are needed. Figs. 3a and 2b show the affine approximations of the T_{max} curves of model 1 and 2. In general, we try to ensure that the affine function lies above the nonlinear curve. This way, more thrust is available and the problem is often easier to solve. We have the relationship

$$T_{\text{max}}(r) = \begin{cases} T_{\text{max}} & r \leq r^* \\ ar + b & r > r^* \end{cases} \tag{31}$$

where r^* depends on the thruster model. In implementation, the case $r \leq r^*$ is implicitly handled through the upper bound T_{max} on T. Therefore, no case-by-case analysis is needed.
This approach can easily be extended to piecewise affine functions. In this paper, the case with two sub-functions is considered as this represents an acceptable approximation of the original nonlinear curve (see Fig. 3b):

$$T_{\text{max}}(r) = \begin{cases} T_{\text{max}} & r \leq r_1^* \\ a_1 r + b_1 & r_1^* < r \leq r_2^* \\ a_2 r + b_2 & r_2^* < r \end{cases} \tag{32}$$

This requires a distinction of cases. We propose to use $\bar{r}$ from the previous iteration to determine which sub-function is to be used.
The solution to this simplified problem serves as the initial guess to solve the problem with the real thruster model. Instead of linearizing the upper bound of constraint Eq. (30) about a reference point $\bar{\mathbf{x}}$, we approximate $T_{\text{max}}(r)$ as

$$T_{\text{max}}(r) \approx T_{\text{max}}(\bar{r}) \tag{33}$$

where $\bar{r}$ from the previous iteration is used. Our simulations suggest that this is often advantageous in terms of convergence.
Due to the partial linearization of the dynamics in Eq. (5), $I_{\text{sp}}(r)$ in $\mathbf{B}(\bar{\mathbf{x}})$ is computed similarly using $\bar{r}$.

Perturbation of Other Bodies

The dynamics of the n-body problem can be written as

$$\begin{bmatrix} \dot{\mathbf{r}} \\ \dot{\mathbf{v}} \\ \dot{m} \end{bmatrix}_{\text{nbody}} = \mathbf{f} + \begin{bmatrix} \mathbf{0} \\ \mathbf{a} \\ 0 \end{bmatrix}_{\text{nbody}} \tag{34}$$

where $\mathbf{a}_{\text{nbody}}$ is the additional acceleration caused by the gravitational pull of other bodies in the solar system:

$$\mathbf{a}_{\text{nbody}} = \sum_{i}^{n} \epsilon_{i,\text{nbody}} \mu_i \left(\frac{\mathbf{r}_{\text{sat},i}}{r_{\text{sat},i}^3} - \frac{\mathbf{r}_i}{r_i^3} \right) \tag{35}$$

In this work, we consider the $n = 9$ perturbations of the barycenters of Mercury, Venus, Earth, Mars, Jupiter, Saturn, Uranus, Neptune and Pluto. 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 are obtained using the software SPICE.[33] Note that Eq. (35) already contains the homotopic parameter $\epsilon_{i,\text{nbody}}$ that adjusts the contribution of each perturbing body individually. It is gradually increased from 0 (no contribution) to 1 (full gravitational influence considered). This way, the perturbing bodies can be added successively until the full n-body problem is obtained. Eq. (35) is linearized and added to the optimization problem as described in Section II. Our rationale is that the solver is more likely to find a feasible solution when increasing the nonlinearity (and hence, the linearized term) only step by step than solving the full problem directly. Considering the real thruster model together with the perturbations caused by the solar radiation pressure and other bodies yields the high-fidelity model.

CLOSED-LOOP GUIDANCE PROCESS

The closed-loop guidance process is shown in Fig. 4. The guidance is determined in a closed-loop fashion where the reference trajectory is recomputed in predefined intervals using the SCP method described in Sections II and III. After each reoptimization, the equations of motion are propagated with the obtained controls for some time interval until a new reference is to be determined. This process continues until the target is reached. To account for uncertainties and other errors, the propagated state is perturbed and then used as the new initial state. As the spacecraft advances in its flight, the number of collocation points is decreased in segments that meet a certain accuracy threshold. The interested reader is referred to Hofmann et al.[23] for details about the closed-loop guidance simulation.

NUMERICAL SIMULATIONS

We calculate fuel-optimal trajectories from the Sun-Earth Lagrange point L_2 (SEL2) to the asteroids Dionysus and 2000 SG344. The open-source Embedded Conic Solver (ECOS)[34] is used to solve the second-order cone program. Physical constants, scaling factors and SCP parameters are given in Tables 1 and 2. The trust-region based SCP algorithm is described in Hofmann et al.[15]
For the homotopic approach, every subproblem is solved to full optimality (maximum constraint violation ϵ_c and change of cost ϵ_ϕ are below some threshold) or until the maximum number of iterations or step tolerance is reached. If not stated otherwise, a simple cubic interpolation is used to generate the (poor) initial guess.

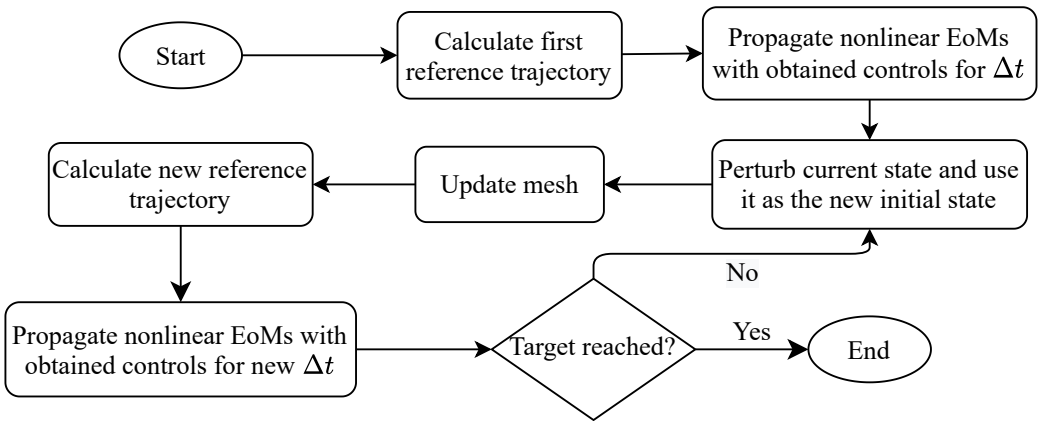


Figure 4: Flowchart of the closed-loop guidance simulation.[23]

Table 1: Physical constants and scaling factors.

Parameter	Value
Gravitational const. μ	1.32712×10^{11} km^3/s^2
Gravitational accel. g_0	9.80665×10^{-3} km/s^2
Solar constant C_{Sun}	1361 W/m^2
Speed of light c	299792458 m/s
Length unit LU = AU	1.49597×10^8 km
Velocity unit VU	$\sqrt{\mu/\text{LU}}$ km/s
Time unit TU	LU/VU s
Acceleration unit ACU	VU/TU km/s^2
Mass unit MU	m_0

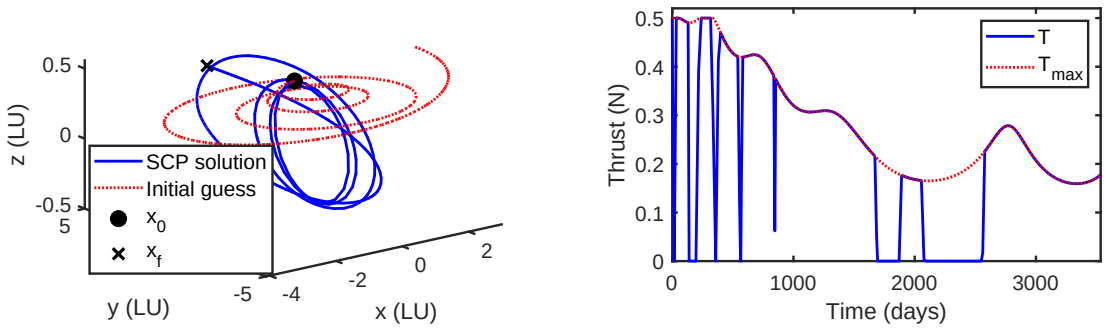
Table 2: Parameters of the SCP algorithm.[15]

Parameter	Value
Penalty weight λ	100
Penalty weight μ	100
Trust region r_0	100
ρ_0, ρ_1, ρ_2	$0.01, 0.25, 0.9$
α	1.5
β	1.5
ϵ_c	10^{-5}, 10^{-6}
ϵ_ϕ	10^{-4}

Assessment of Homotopic Approach

For each dynamical model described in Section III, we calculate 350 fuel-optimal trajectories to asteroid Dionysus with poor initial guesses to assess the performance of the homotopic approach. This example is considered challenging due to the long time of flight and large changes in orbital elements; several revolutions are required to reach the target. Moreover, the maximum thrust and specific impulse decrease significantly over time as the distance between spacecraft and Sun increases. The initial guesses are generated using a simple shape-based cubic interpolation method where we impose non-integer values in the range [3.0, 6.5] on the number of revolutions. This way, neither the dynamics nor the final boundary condition are satisfied. Two example trajectories and thrust profiles for the high-fidelity model are illustrated in Figs. 5 and 6. It is evident that the initial guesses are far from the optimal solutions. The characteristic bang-off-bang control structure confirms that optimal solutions were found that comply with the real thruster model. We compare the number of converged cases, average iterations and final masses, including the $\pm 1\sigma$ standard deviation. A maximum constraint violation tolerance of $\epsilon_c = 10^{-5}$ is used. The parameters for the simulations are given in Table 3.

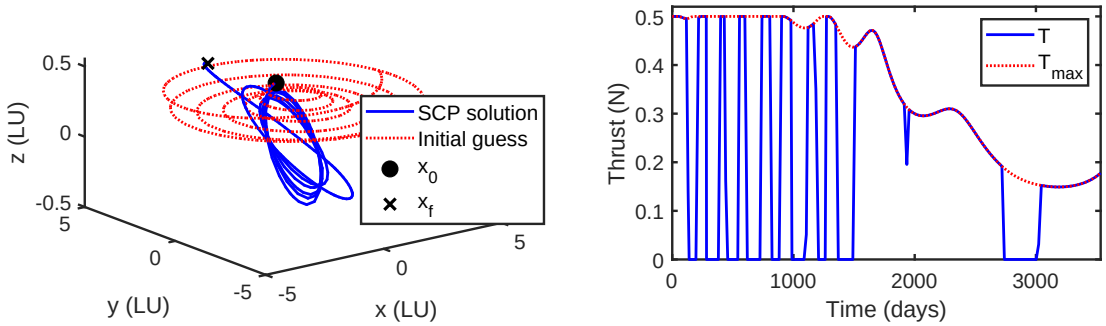
Throughout this section, $2b$ refers to the fuel-optimal problem with two-body dynamics, constant maximum thrust and specific impulse, and no additional perturbations. The notation X $\rightarrow$ Y indi-



(a) SCP trajectory and initial guess.

(b) Thrust magnitude and $T_{\max}(r)$.

Figure 5: Trajectory and thrust magnitude for a transfer to asteroid Dionysus with three revolutions.



(a) SCP trajectory and initial guess.

(b) Thrust magnitude and $T_{\max}(r)$.

Figure 6: Trajectory and thrust magnitude for a transfer to asteroid Dionysus with six revolutions.

cates that problem X is solved first and then used as an initial guess for problem Y.

Energy-to-Fuel Homotopy. We perform 15 homotopic steps with $\epsilon_{\text{energy}} = [1.0, 0.95, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1, 0.01, 0.001, 0.0]$ for the energy-to-fuel homotopy. Only two-body dynamics with constant $T_{\max}$ and I_{sp} and no perturbations are considered. The results are shown in Table 4. Even though solving the fuel-optimal problem directly yields a high convergence rate of 89 %, performing a homotopy from the energy-optimal problem slightly increases this figure. As we solve each step to full optimality, the number of iterations is considerably larger due to the 15 intermediate optimizations. One remedy could be to reduce the number of steps or limit the number of iterations during the homotopy. Nevertheless, the average run time of few minutes on a standard Laptop seems to be acceptable for such a large number of iterations.

Solar Radiation Pressure Homotopy. The results for six homotopic steps $\epsilon_{\text{SRP}} = [0.0001, 0.001, 0.01, 0.1, 0.5, 1.0]$ are given in Table 5 where SRP refers to solving the problem directly with a cubic interpolation guess. The figures are similar to the previous case: solving the problem directly already yields a high percentage of converged cases of 89 %, but adding a homotopy increases it to 95 % at the cost of more iterations and hence, computational time. Although the SRP homotopy finds the maximum final masses, all values are similar.

Table 3: Simulation values for the SEL_2 to Dionysus transfer.[35]

Parameter	Value
Initial epoch	23-Dec-2012 00:00:00 UTC
Initial mass $m(t_0)$	4000 kg
Final position $\mathbf{r}_f$	$[-2.0406, 2.0518, 0.5543]^\top$ LU
Final velocity $\mathbf{v}_f$	$[-0.1423, -0.4511, 0.0189]^\top$ VU
Final mass $m(t_f)$	free
$T_{\max}$, I_{sp} (fixed I_{sp})	0.32 N, 3000 s
$T_{\max}$, $I_{\text{sp,max}}$ (variable I_{sp})	0.5 N, 5000 s
Thruster model	1
Time of flight t_f	3534 days
Projected spacecraft area A_{SC}	100 m^2
Reflectivity coefficient C_R	1.3
Total number of collocation points	250
Number of segments	25

Table 4: Results of energy-to-fuel homotopy.

Simulation	Converged	Converged (%)	Avg. iter.	Avg. $m(t_f)$ (kg)
Fuel-opt.	312	89	70.1 ± 20.1	1805 ± 428
Homotopy	328	94	784.9 ± 104.0	2116 ± 223

Variable Thrust and Specific Impulse Homotopy. The variable I_{sp} homotopy consists of approximating the curves with affine and piecewise affine functions first before using the real thruster model. As the maximum thrust and specific impulse decrease considerably with the distance to the Sun in the chosen model, the algorithm was only able to find optimal solutions in 80 % of the cases when considering the real thruster model directly (Var. I_{sp}^{b}, see Table 6). The convergence rate increases by approximately eight percent when (piecewise) linear approximations are included. It can therefore be an effective means to find feasible solutions in case of complex transfers or when poor initial guesses are provided. As mentioned in Section III, using $T_{\max}(r) \approx T_{\max}(\bar{r})$ instead of linearizing the upper bound of Eq. (30) improves the convergence rate, cf. I_{sp}^{a} and I_{sp}^{b}. Note that the final masses of the homotopic approach are lower than in the other cases. The reason is that the former found additional solutions that correspond to lower final masses, thus reducing the average value.

Table 5: Results of the SRP homotopy.

Simulation	Converged	Converged (%)	Avg. iter.	Avg. $m(t_f)$ (kg)
SRP	312	89	68.8 ± 18.4	1800 ± 432
2b $\rightarrow$ SRP	316	90	88.9 ± 29.5	1796 ± 440
2b $\rightarrow$ SRP homotopy	338	95	256.3 ± 116.5	1839 ± 481

Table 6: Results of the variable maximum thrust and specific impulse homotopy.

Simulation	Converged	Converged (%)	Avg. iter.	Avg. $m(t_f)$ (kg)
Var. I_{sp}[a]	251	72	74.1 ± 14.1	2148 ± 391
Var. I_{sp}[b]	279	80	80.9 ± 25.2	1807 ± 461
2b $\rightarrow$ var. I_{sp}	296	85	134.5 ± 35.5	1783 ± 485
2b $\rightarrow$ var. I_{sp} homotopy	309	88	237.4 ± 80.9	1712 ± 495

[a] Linearizing the upper bound of Eq. (30).
[b] Using $T_{\max}(r) \approx T_{\max}(\bar{r})$.

N-body Homotopy. With regard to the n-body problem, it is apparent that this is difficult to solve as the dynamics are highly nonlinear. The results are presented in Table 7. Remarkably, despite the poor initial guesses, SCP was still able to find optimal solutions in 78 % of the cases when solving the n-body problem directly (cf. $N\text{-}body$ in the first row). When solving the n-body problem directly without any homotopy using the solution of $2b$ as the initial guess, the convergence increases by more than ten percent. Still, incorporating a homotopic approach results in more converged cases (310 vs. 323 and 334). Also note that because the spacecraft departs close to Earth at the SEL_2 point, adding the gravitational influence of the Earth separately allows us to determine a few more optimal solutions (see the case $2b \rightarrow 3b$ hom. $\rightarrow$ n-body hom.). We used $\epsilon_{\text{nbody}} = [0.0001, 0.001, 0.01, 0.1, 0.5, 1.0]$ for all simulations. The discrepancy in the final masses is again because of the additional solutions that were found.

Table 7: Results of the n-body homotopy.

Simulation	Converged	Converged (%)	Avg. iter.	Avg. $m(t_f)$ (kg)
N-body	272	78	72.4 ± 16.4	1874 ± 398
2b $\rightarrow$ n-body	310	89	107.1 ± 22.8	1825 ± 424
2b $\rightarrow$ n-body homotopy	323	92	202.6 ± 95.1	1852 ± 436
2b $\rightarrow$ 3b hom. $\rightarrow$ n-body hom.	334	95	303.7 ± 177.9	1809 ± 475

Comparison with a Nonlinear Programming Solver

We compare SCP with a nonlinear programming approach in terms of convergence, iterations and computational time. We use the DIRETTO (DIREct collocation Tool for Trajectory Optimization),[36] a tool developed at Politecnico di Milano that is based on a Hermite–Simpson collocation method, together with IPOPT.[37] As the NLP solver is not able to solve the Dionysus transfer without a decent initial guess, we choose a simpler transfer from SEL_2 to the near-Earth asteroid 2000 SG344, which is a possible target of ESA's Miniaturised Asteroid Remote Geophysical Observer (M-ARGO) mission.[20] The simulation values are given in Table 8. We reduce the the maximum constraint violation tolerance to $\epsilon_c = 10^{-6}$. The initial guesses are generated as follows:

- Guess 1: cubic interpolation that does neither satisfy the dynamics nor the final boundary condition. We perform 100 simulations where the number of revolutions of the initial guess ranges from 1.6 to 2.6. It is considered a poor initial guess.

- Guess 2: propagating the nonlinear dynamics for t_f with constant tangential thrust. We run again 100 simulations. For each guess, a thrust magnitude $T \in [0, T_{\max}]$ is chosen. Such an initial guess is considered relatively good as the dynamics are satisfied and the propagated final state is often close to the target state.

Table 8: Simulation values for the SEL_2 to asteroid 2000 SG344 transfer.[20]

Parameter	Value
Initial epoch	04-Feb-2024 12:00:00 UTC
Initial mass $m(t_0)$	22.6 kg
Final mass $m(t_f)$	free
Maximum thrust $T_{\max}$	2.2519 mN
Maximum specific impulse $I_{\text{sp,max}}$	3067 s
Thruster model	2
Time of flight t_f	700 days
Projected spacecraft area A_{SC}	0.05 m^2
Reflectivity coefficient C_R	1.3
Total number of collocation points	150
Number of segments	15

The converged cases and average number of iterations, CPU time and final mass are reported in Table 9. In contrast to SCP that converged in 64 % of the cases, the NLP solver was not able to find optimal solutions for the poor initial guesses at all (cf. *Guess 1*). When a decent initial guess is provided (*Guess 2*), however, the NLP method determined 69 optimal solutions out of 100. Still, it required on average almost two times as many iterations as SCP which failed in two cases, resulting in a considerably higher CPU time (112.6 vs. 14.5 seconds). The final masses are instead very similar. The results confirm that a NLP method requires a decent initial guess whereas a poor guess often suffices for SCP.

Table 9: Comparison of SCP and NLP for transfers to asteroid 2000 SG344.

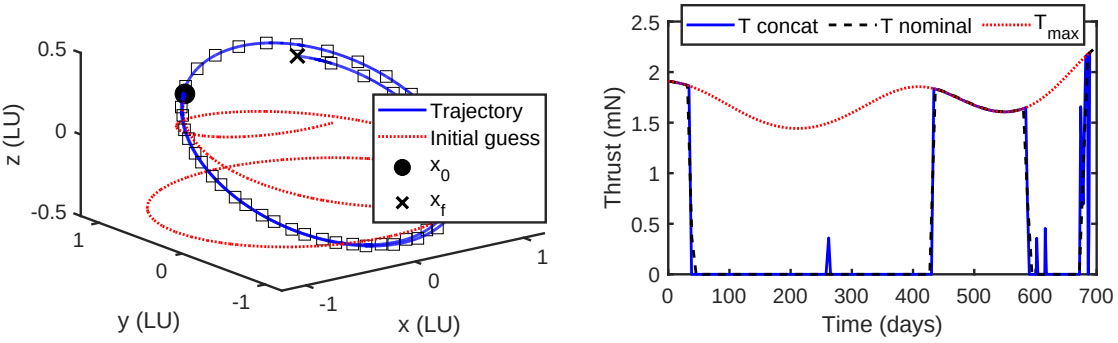
Simulation		Converged	Converged (%)	Iterations	CPU time (s)	$m(t_f)$ (kg)
Guess 1	SCP	64	64	44.3 ± 34.3	18.7 ± 14.7	21.29 ± 0.42
	NLP	0	0	—	—	—
Guess 2	SCP	98	98	36.5 ± 29.1	14.5 ± 11.2	21.44 ± 0.16
	NLP	69	69	66.3 ± 36.6	112.6 ± 59.7	21.45 ± 0.01

Closed-Loop Guidance Simulation

We extend the approach in Hofmann et al.[23] and perform the closed-loop guidance simulation to asteroid 2000 SG344 using a high-fidelity model. A poor initial guess is provided that requires a homotopic approach to find an optimal solution in the first iteration. As this is a simple transfer with few revolutions only, it suffices to solve the two-body problem first and then directly proceed to the high-fidelity model. In all subsequent reoptimizations, no additional homotopy is applied as the previous solutions serve as decent initial guesses. The random position and velocity perturbances

are -10^4 to 10^4 km and -10^{-3} to 10^{-3} km/s, respectively. The trajectory is recomputed every 14 days, hence resulting in a total of 50 reference trajectories to be calculated given the time of flight of 700 days. Other simulation values are given in Table 8.

The concatenated propagated trajectory (that is, the trajectory flown by the spacecraft) and the initial guess are shown in Fig. 7a. The squares indicate the positions where a new reference was computed. Apparently, the poor initial guess does neither satisfy the dynamics nor the final boundary condition. Fig. 7b illustrates the thrust magnitudes where *T nominal* corresponds to the obtained control history in the first iteration. Both curves match well despite some small spikes that occur during the closed-loop guidance; a finer mesh is likely to eliminate those. Apart from that, the real thruster behavior was captured accurately as the thrust magnitude either follows the $T_{\max}$ curve or is zero. The number of iterations and computational time are only high for the first iteration (see Fig. 8) due to the required homotopy. Despite the high-fidelity model, computing all other reference trajectories required only two iterations and 0.5 seconds on average, thus making the performance similar to the two-body simulation in Hofmann et al.[23] As we consider additional perturbations in our simulation, the propagation error is considerably lower (order of 10^0 km or less). The obtained final mass of 21.49 kg after reaching the target is almost identical to the nominal one.



(a) Trajectory flown by the spacecraft.

(b) Thrust magnitudes.

Figure 7: Trajectory flown by the spacecraft and thrust magnitude for the closed-loop guidance simulation.

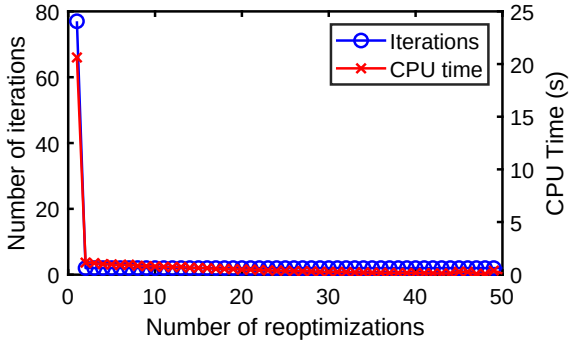


Figure 8: Iterations and CPU time for each optimization in the closed-loop guidance.

CONCLUSION

The low-thrust fuel-optimal problem was addressed using a pseudospectral SCP method. It was shown that high-fidelity models such as solar radiation pressure, variable maximum thrust and specific impulse, and the gravitational influence of other bodies can be solved with standard SCP techniques. A homotopic approach was proposed that improves the convergence rate when more complex models are considered. Thousands of transfers to an asteroid demonstrate the excellent robustness even in highly nonlinear applications.

This work has demonstrated that SCP can outperform a standard NLP solver even when more complex dynamical models are considered. The high robustness against poor initial guesses make it an appealing choice whenever a high convergence rate is desired. Apart from artificial intelligence, numerical optimization is the second important class of methods for onboard guidance and control. Especially SCP has proven in several occasions that it is suitable for real-time applications. The simulations suggest that a high-fidelity model can in general be integrated into a closed-loop guidance scheme without a significant increase of computational effort. For problems with few revolutions, the performance is comparable to the low-fidelity model, except for the accuracy that can be improved considerably. This is a promising result as SCP is capable of solving also highly nonlinear problems in a short amount of time.

REFERENCES

[1] "Spacecraft Trajectory Optimization: A Review of Models, Objectives, Approaches and Solutions," *Progress in Aerospace Sciences*, Vol. 102, 2018, pp. 76–98, 10.1016/j.paerosci.2018.07.007.

[2] P. Lu, "Introducing Computational Guidance and Control," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 2, 2017, pp. 193–193, 10.2514/1.G002745.

[3] 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.

[4] C. N. Dennehy, "Autonomous GN&C," 11th International ESA Conference on Guidance, Navigation & Control Systems, Keynote talk, 2021.

[5] 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.

[6] B. A. Conway, "A Survey of Methods Available for the Numerical Optimization of Continuous Dynamic Systems," *Journal of Optimization Theory and Applications*, Vol. 152, No. 2, 2012, pp. 271 – 306, 10.1007/s10957-011-9918-z.

[7] X. Liu, P. Lu, and B. Pan, "Survey of Convex Optimization for Aerospace Applications," *Astrodynamics*, Vol. 1, No. 1, 2017, pp. 23–40, https://doi.org/10.1007/s42064-017-0003-8.

[8] 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.

[9] D. Morgan, S.-J. Chung, and F. Y. Hadaegh, "Model Predictive Control of Swarms of Spacecraft Using Sequential Convex Programming," *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 6, 2014, pp. 1725–1740, 10.2514/1.G000218.

[10] Y. Chen, M. Cutler, and J. P. How, "Decoupled Multiagent Path Planning via Incremental Sequential Convex Programming," *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 5954–5961, 10.1109/ICRA.2015.7140034.

[11] 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.

[12] Z. Wang and M. J. Grant, "Optimization of Minimum-Time Low-Thrust Transfers Using Convex Programming," *Journal of Spacecraft and Rockets*, Vol. 55, No. 3, 2018, doi: 10.2514/1.A33995.

[13] G. Tang, F. Jiang, and J. Li, "Fuel-Optimal Low-Thrust Trajectory Optimization Using Indirect Method and Successive Convex Programming," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 54, No. 4, 2018, pp. 2053–2066, 10.1109/TAES.2018.2803558.

[14] 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.

[15] C. Hofmann and F. Topputo, "Toward On-Board Guidance of Low-Thrust Spacecraft in Deep Space Using Sequential Convex Programming," *AAS/AIAA Space Flight Mechanics Meeting*, February 2021. AAS Paper 21-350.

[16] M. Saghamanesh, E. Taheri, and H. Baoyin, "Systematic Low-Thrust Trajectory Design to Mars Based on a Full Ephemeris Modeling," *Advances in Space Research*, Vol. 64, No. 11, 2019, pp. 2356–2378, 10.1016/j.asr.2019.08.013.

[17] 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.

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

[19] J. Englander, J. M. Knittel, K. Williams, D. Stanbridge, and D. H. Ellison, "Validation of a Low-Thrust Mission Design Tool Using Operational Navigation Software," *AAS/AIAA Space Flight Mechanics Meeting*, February 2017. AAS Paper 17-204.

[20] 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.

[21] 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.

[22] A. Mereta and D. Izzo, "Target Selection for a Small Low-Thrust Mission to Near-Earth Asteroids," *Astrodynamics*, Vol. 2, No. 3, 2018, pp. 249–263, 10.1007/s42064-018-0024-y.

[23] C. Hofmann and F. Topputo, "Closed-Loop Guidance for Low-Thrust Interplanetary Trajectories Using Convex Programming," *11th International ESA Conference on Guidance, Navigation & Control Systems*, June 2021. Paper 46.

[24] X. Liu, Z. Shen, and P. Lu, "Exact Convex Relaxation for Optimal Flight of Aerodynamically Controlled Missiles," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 52, No. 4, 2016, 10.1109/TAES.2016.150741.

[25] 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.

[26] M. Sagliano, S. Theil, M. Bergsma, V. D'Onofrio, L. Whittle, and G. Viavattene, "On the Radau Pseudospectral Method: Theoretical and Implementation Advances," *CEAS Space Journal*, Vol. 9, 2017, 10.1007/s12567-017-0165-5.

[27] M. Sagliano, "Pseudospectral Convex Optimization for Powered Descent and Landing," *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 2, 2018, 10.2514/1.G002818.

[28] M. Sagliano, "Generalized hp Pseudospectral-Convex Programming for Powered Descent and Landing," *Journal of Guidance, Control, and Dynamics*, Vol. 42, No. 7, 2019, 10.2514/1.G003731.

[29] D. Garg, *Advances in Global Pseudospectral Methods for Optimal Control*. PhD thesis, University of Florida, 2011.

[30] C. Zhang, F. Topputo, F. Bernelli-Zazzera, and Y. S. Zhao, "Low Thrust Minimum Fuel Optimization in the Circular Restricted Three Body Model," *Advances in the Astronautical Sciences*, Vol. 153, No. 8, 2015, pp. 1597–1615, 10.2514/1.G001080.

[31] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.

[32] 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.

[33] 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.

[34] 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.

[35] 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.

[36] F. Topputo, D. A. Dei Tos, K. Mani, and e. al., "Trajectory design in high-fidelity models," *7th International Conference on Astrodynamics Tools and Techniques (ICATT)*, November 2018, pp. 1–9.

[37] A. Wächter and L. T. Biegler, "Low Thrust Minimum Fuel Optimization in the Circular Restricted Three Body Model," *Mathematical Programming*, Vol. 106, No. 1, 2006, pp. 25—-57, 10.1007/s10107-004-0559-y.