



Peloni, A. , Rao, A. V. and Ceriotti, M. (2018) Automated Trajectory Optimizer for Solar Sailing (ATOSS). *Aerospace Science and Technology*, 72, pp. 465-475. (doi:[10.1016/j.ast.2017.11.025](https://doi.org/10.1016/j.ast.2017.11.025))

This is the author's final accepted version.

There may be differences between this version and the published version. You are advised to consult the publisher's version if you wish to cite from it.

<http://eprints.gla.ac.uk/151589/>

Deposited on: 13 November 2017

Enlighten – Research publications by members of the University of Glasgow
<http://eprints.gla.ac.uk>

Automated Trajectory Optimizer for Solar Sailing (ATOSS)

Alessandro Pelsoni ^{a,*}, Anil V. Rao ^b, Matteo Ceriotti ^a

^a *School of Engineering, University of Glasgow, Glasgow G12 8QQ, Scotland, United Kingdom*

^b *Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, Florida 32611-6250, USA*

Nomenclature

Acronyms

ATOSS	Automated Trajectory Optimizer for Solar Sailing
GA	Genetic Algorithm
NEA	Near-Earth Asteroid
NHATS	Near-Earth Object Human Space Flight Accessible Target Study
NLP	Nonlinear Programming
OCP	Optimal Control Problem
PHA	Potentially Hazardous Asteroid

Symbols

$\mathbf{A}(\mathbf{x})$	Matrix of the dynamics
\mathbf{a}	Acceleration vector, mm/s^2
a_c	Characteristic acceleration, mm/s^2
a	Semimajor axis, AU

* Corresponding author.

E-mail addresses: a.pelsoni.1@research.gla.ac.uk (A. Pelsoni), anilvrao@ufl.edu (A.V. Rao), matteo.ceriotti@glasgow.ac.uk (M. Ceriotti).

$\mathbf{b}(\mathbf{x})$	Vector of the dynamics
e	Eccentricity
f, g	In-plane modified equinoctial elements
i	Inclination, deg
j	Phase number
L	True longitude, rad
L_0	Initial true longitude, rad
L_f	Final true longitude, rad
\hat{L}_f	Lower boundary on L_f , rad
\hat{N}	Unit vector perpendicular to the sail plane, $[N_r \ N_g \ N_h]^T$
p	Semilatus rectum, AU
\mathbf{r}	Cartesian position vector ($r \triangleq \ \mathbf{r}\ $), AU
r_\oplus	Mean Sun-Earth distance, 1 AU
$\{\hat{\mathbf{r}}, \hat{\boldsymbol{\theta}}, \hat{\mathbf{h}}\}$	Orbital reference frame
t_{0f}	Time of flight, days
\mathbf{u}^*	Optimal control vector
\mathbf{X}	Set of free parameters for the shape-based method
\mathbf{x}	State vector
α	Sail cone angle, rad
α^*	Optimal sail cone angle, rad
Δa	Semimajor axis variation, AU
Δv	Velocity increment, km/s
Δt_f	Value used to decrease the lower boundary on the final time, days
δ	Sail clock angle, rad
λ_p	Shaping parameter related to semilatus rectum, AU
λ_{fg}	Shaping parameter related to in-plane modified equinoctial elements

μ	Gravitational parameter of the Sun, $1.3271 \times 10^{11} \text{ km}^3/\text{s}^2$
φ_p	Phasing parameter related to semilatus rectum, rad
φ_{fg}	Phasing parameter related to in-plane modified equinoctial elements

Subscripts and Superscripts

$\hat{\square}$	Unit vector
$\dot{\square}$	Time derivative
\square^T	Transpose
\square_f	Value dependent on boundary conditions at final time
\square_i	Value dependent on boundary conditions at initial time
\square_{\min}	Minimum
\square_{pss}	Pseudo solar sail

Abstract

The problem of finding an optimal solar-sail trajectory must be solved by means of numerical methods, since no analytical, closed-form solutions exist. A new tool named ATOSS (Automated Trajectory Optimizer for Solar Sailing) has been developed for optimizing multi-phase solar-sail trajectories. A shape-based method for solar sailing and a two-stage approach for the optimization are the keys to the success of ATOSS, which operates with minimum inputs required to the user. Once the initial guess is generated by means of the shape-based method, the above mentioned two-stage approach works as follows. First, a solution to the optimal control problem at hand is sought; subsequently, the boundaries on the times are modified so that a better solution, in terms of total mission duration, is searched. Several numerical test cases are presented to demonstrate ATOSS' ability to automatically find optimal solar-sail trajectories for single- and multi-phase optimization problems. Moreover, the shape-based method for solar sailing has been validated as a viable method to produce initial guess solutions for a direct optimization algorithm.

Keywords: Solar sailing; Shape-based method; Automatic trajectory optimization; GPOPS-II

1. Introduction

Propelled only by sunlight, a solar sail is an attractive alternative to conventional low-thrust propulsion, such as electric propulsion, for deep-space missions. A solar sail is not constrained by the amount of available propellant and, therefore, it can achieve goals unattainable for conventional thrusters [1-6]. Nevertheless, no analytical, closed-form solution exists to the solar-sail trajectory problem and an optimal control problem (OCP) must be solved, as in the case of classical low-thrust propulsion. From the control point of view, there are two main differences between a solar sail and a classical low-thrust propulsion: these are related to the magnitude and direction of the acceleration provided. In fact, the magnitude and direction given by a solar sail are strongly related [7] and cannot point towards the Sun.

To date, several studies have been carried out to find preliminary low-thrust trajectories in a quick and reliable way [8-13]. Approximated or sub-optimal trajectories are, in fact, useful in a preliminary mission design, in which a wide range of possible mission scenarios are to be evaluated at the same time, and solving an OCP for each of those is usually inefficient. For this reason, a shape-based method has been recently developed to deal explicitly with solar-sail trajectories [14].

The shape-based method gives approximated trajectories and sub-optimal solutions, hence it can be used to generate initial-guess solutions for higher-fidelity direct-optimization models. A variable-order adaptive Radau collocation method is implemented within GPOPS-II [15], a well-known and widely-used general-purpose optimal control software. It transcribes the continuous problem into a nonlinear programming (NLP) problem, which is solved through a state-of-the-art NLP solver. Recently, the interface for the ESA NLP solver WORHP [16] has been implemented into GPOPS-II and is currently under testing.

Despite the fact that direct-collocation methods are usually more robust to poor initial guesses than indirect methods [17], the choice of the initial guess plays an important role in the quality of the optimized solution. Although this issue is not widely discussed in the literature, few works highlight the importance of a good initial guess even for direct-collocation methods. Porsa et al. [18] showed how the performances of the optimizer, in terms of both fitness function and computational load, are sensitive to the choice of initial guess. The need of a good initial guess was critical for the convergence of the NLP solver in [19]. The influence of different problem formulations on the robustness of the direct-collocation method was investigated in [20], in which it was shown that not all the choices of initial guess guaranteed convergence of the optimizer. In the same paper, it is discussed, for instance, how SNOPT never converged if a good initial guess was not available, whereas IPOPT was more likely to find a solution only if the mesh had few nodes. Moreover, Graham and Rao [21] developed an algorithm

to generate the initial guess for a multi-revolution low-thrust transfer in order to help the convergence of the NLP solver. To do so, a set of optimal control sub-problems needed to be solved to create the initial-guess solution.

Even if a good initial guess is provided, issues can arise in the computation of the solution by a numerical optimizer [17]. A proper scaling of the problem, the differentiation algorithm used to compute the derivatives, and the way the continuous problem is discretized are only three of the possible issues in the numerical optimization. Moreover, different settings of the numerical optimizer can potentially lead to issues in the convergence of the NLP solver. In the adaptive mesh refinements described in [22, 23], for instance, the number of collocation points is explicitly dependent to the maximum/minimum allowed polynomial degree chosen for the discretization. That is, different settings affect the number and location of the collocation points and this can impair the ability of the NLP solver to find a solution for the given formulation of the problem. Therefore, trajectory-optimization problems are usually solved once at a time, allowing the user to manually tweak some settings, often in a trial-and-error fashion, in order to help the convergence of the solver or to get better results. To the best of the authors' knowledge, the work of Peloni et al. [24] is the only one that shows a large number of solar-sail multiple-rendezvous OCPs solved in an automated way.

For the aforementioned reasons, the primary aim of this paper is to present a toolbox able to find solar-sail multiple-rendezvous trajectories in an automated way, thus improving the results shown in [24]. Such tool is the first step towards a more efficient way to generate several solutions for preliminary solar-sail mission design. Moreover, the reliability of the shape-based method for solar sailing as an initial-guess solution for a direct collocation method is validated against a transversal-thrust law.

2. Shaped Solar-Sail Trajectories

In the two-body problem dynamics, it is possible to compute the acceleration \mathbf{a} needed to follow a given trajectory, in an inertial Cartesian frame, as:

$$\mathbf{a} = \ddot{\mathbf{r}} + \mu \frac{\mathbf{r}}{r^3} \quad (1)$$

in which μ is the gravitational parameter of the central body (which is the Sun, in this case) and \mathbf{r} is the position vector of the sailcraft. Therefore, the purpose of the shape-based method for solar sailing is to find a shape of the trajectory such that the acceleration needed to generate it is feasible by the chosen solar sail. The advantage of this approach is that the acceleration history can be obtained analytically, once the shape of the trajectory is fixed. On the contrary, it is not guaranteed that the trajectory is characterized by an acceleration within the boundaries

of the chosen technology. That is, the analytical representation of the trajectory shall have the possibility to be properly modified to fulfill the thrust-profile specifications.

Peloni et al. [14] proposed a set of shaping functions to describe coplanar solar-sail transfers in terms of modified equinoctial elements $\mathbf{x} = [p, f, g, L]^T$ [25, 26]. The shape of the trajectory is described, as a function of the true longitude, by:

$$\begin{cases} p = p_I \exp[p_F (L - L_0)] + \lambda_p \sin(L + \varphi_p) \\ f = f_I + f_F (L - L_0) + \lambda_{f_g} \sin(L + \varphi_{f_g}) \\ g = g_I + g_F (L - L_0) - \lambda_{f_g} \cos(L + \varphi_{f_g}) \end{cases} \quad (2)$$

in which L_0 is the true longitude at the initial time, $\{p_I, f_I, g_I\}$ and $\{p_F, f_F, g_F\}$ depend on the initial and final conditions, respectively. It is possible to verify that the shaping functions shown in Eq. (2) do not describe osculating modified equinoctial elements, as discussed in [27]. That is, it is not true, in general, that $p(L_0) = p_I$, $f(L_0) = f_I$, and $g(L_0) = g_I$. The same holds for the values associated with the final true longitude L_f . In fact, the values of $\{p_I, f_I, g_I\}$ and $\{p_F, f_F, g_F\}$ are found by means of a Newton loop so that the constraints on both initial and final Cartesian state are satisfied, as discussed in [27]. The terms λ_p and λ_{f_g} are the so-called shaping parameters and they directly affect the shape of the trajectory. Lastly, the terms φ_p and φ_{f_g} are the so-called phasing parameters and they affect the phase of the trigonometric functions that characterize the shaping functions. Therefore, as shown in Eq. (2), the four free parameters $\mathbf{X} = [\lambda_p, \lambda_{f_g}, \varphi_p, \varphi_{f_g}]^T$ fully define the trajectory, once initial and final states are given.

The four free parameters shall be properly tuned so that the acceleration retrieved through Eq. (1) is achievable by the sailcraft under consideration. Specifically, a perfectly-reflecting flat solar sail is considered in this work. The acceleration \mathbf{a} of such sailcraft depends on its characteristic acceleration a_c and its distance r from the Sun, as follows:

$$\mathbf{a} = a_c \left(\frac{r_{\oplus}}{r} \right)^2 (\hat{\mathbf{N}} \cdot \hat{\mathbf{r}})^2 \hat{\mathbf{N}} \quad (3)$$

in which $\hat{\mathbf{r}}$ is the Sun-spacecraft unit vector and $\hat{\mathbf{N}}$ is the unit vector perpendicular to the sail plane opposite to the Sun. In the orbital reference frame $\{\hat{\mathbf{r}}, \hat{\boldsymbol{\theta}}, \hat{\mathbf{h}}\}$, $\hat{\mathbf{N}}$ can be expressed by means of the cone angle α and the clock angle δ , which are defined as follows:

$$\hat{\mathbf{N}} = \begin{bmatrix} N_r \\ N_g \\ N_h \end{bmatrix} = \begin{bmatrix} \cos \alpha \\ \sin \alpha \cos \delta \\ \sin \alpha \sin \delta \end{bmatrix} \quad (4)$$

A MATLAB built-in genetic algorithm (GA) is used to find the set of free parameters \mathbf{X} , together with the number of complete revolutions and the time of flight t_{0f} , so that the shaped trajectory is as close as possible to a solar-sail one. Specifically, the time of flight is used to compute the desired Cartesian state of the target object so that the shape of the trajectory can be drawn between the initial and final state. Moreover, a set of nonlinear constraints is implemented so that: a) the acceleration is always directed along the Sun-spacecraft direction, b) the magnitude of the acceleration is compatible with Eq. (3), and c) the time of flight found by the GA and the one found by integrating the variation of the time over true longitude are compatible [9, 27]. In this paper, the objective function of the GA to be minimized is the time of flight. Note that the evaluation of the objective function by the GA is very fast, since the time of flight is one of the optimization variables. Therefore, only the evaluation of the function of the constraints requires sensible computational time. The reader is referred to Ref. [14] for a detailed description of the shape-based method for solar sailing.

Note that the shape-based method described above, as well as the majority of shape-based methods in the literature, deal with rendezvous problems. In fact, if the problem under consideration is an orbit transfer, this method cannot be used as it is. Therefore, a different implementation of the shape-based method is developed for orbit transfers. Specifically, in an orbit transfer problem, the spacecraft is not required to be in any specific point on the desired target orbit. Therefore, the number of complete revolutions and the time of flight are replaced by the final true longitude L_f as the unknown of the GA. Consequently, the objective function for the GA to be minimized becomes the final value of the true longitude. Lastly, since t_{0f} is irrelevant, the third nonlinear constraint discussed above can be ignored. In conclusion, the implementation of the shape-based method for an orbit transfer problem is characterized by one less nonlinear constraint and one less variable to be considered within the GA if compared to the rendezvous problem.

3. ATOSS: Automated Trajectory Optimizer for Solar Sailing

The Automated Trajectory Optimizer for Solar Sailing (ATOSS) has been developed to find optimal solar-sail trajectories automatically and reliably. ATOSS combines an initial-guess generator, which uses the shape-based method described in Section 2, with an optimization strategy based on a two-stage approach. Such approach, described in detail in Section 3.3, first finds a solution to the OCP and then improves it by modifying the

description of the problem. A general-purpose optimal control solver (GPOPS-II [15] in this case) is used to find the optimal solution to the solar-sail trajectory problem.

ATOSS can function in two modes, depending on the availability of a timed sequence. The timed sequence is defined as a sequence of objects with preliminary epochs for the object encounters (timeline). This allows ATOSS to be easily interfaced with the sequence-search algorithm described in [14] and thus to have a standalone toolbox able to both look for preliminary sequences and optimize their trajectories, but also to work independently of it. In those cases for which only a sequence of objects is provided (i.e. a non-timed sequence), ATOSS will self-generate the preliminary timeline. This is generated considering the locally-optimal control laws to maximize the rate of change of the orbital parameters, as described in [7]. In particular, the laws to change individually semimajor axis, eccentricity and inclination are implemented within ATOSS. For each transfer between two consecutive objects, the preliminary timeline considers the time of flight needed to achieve the desired change in one or more orbital elements. In this work, the allowed time of flight for each transfer leg is bounded between 500 and 1,500 days. Moreover, a set stay time is added between two consecutive transfers (100 days, in this work). It is worth noting that Graham and Rao [21] showed that “the solutions to the minimum-time low-thrust optimal control problem [...] have essentially the same number of revolutions as that of the initial guess”. Therefore, to improve ATOSS’ robustness in case of a non-timed sequence, the possibility to add an extra full revolution to the self-generated time of flight is considered.

3.1. Problem formulation

ATOSS solves both single- and multi-phase problems. In this context, a single-phase problem is defined as a transfer between a departure and a target object (celestial bodies); instead, a multi-phase problem is characterized by several consecutive transfer legs, starting and ending at an object. The transfer legs are connected through coasting arcs during which the spacecraft stays in proximity of the target object, and follows the same ballistic trajectory, as schematically shown in Fig. 1. Since the spacecraft is considered in a state of rendezvous with the object between two consecutive transfer legs, the coasting arc is not explicitly modeled inside ATOSS. The multi-phase problem is formulated such that the state of the sailcraft at the times corresponding to the beginning and end of each phase matches with the state of the target object.

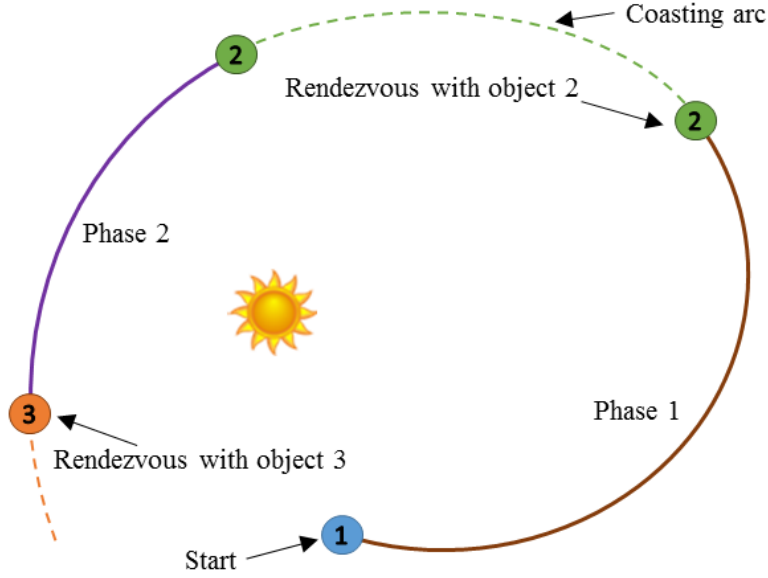


Fig. 1. Schematic of the multi-phase problem under consideration.

The dynamics of each phase are defined as:

$$\dot{\mathbf{x}} = \mathbf{A}(\mathbf{x})\mathbf{a} + \mathbf{b}(\mathbf{x}) \quad (5)$$

in which $\mathbf{A}(\mathbf{x})$ and $\mathbf{b}(\mathbf{x})$ are, respectively, the matrix and vector of the dynamics in accordance to [28]. The acceleration \mathbf{a} is the one given by Eq. (3). A direct collocation method is used to find the optimal control vector $\mathbf{u}^* \equiv \hat{\mathbf{N}}$ such that the total mission duration is minimized while fulfilling the dynamics constraints shown in Eq. (5) at any time.

3.2. Initial-guess generation

Three different methods can be used to generate the initial-guess solution needed for the optimizer:

1. Shape-based method: propagated trajectory.
2. Shape-based method: shaped trajectory.
3. Transversal-thrust law.

Note that the first two are two different variants of the shape-based method described in Section 2; the last one is to be used as a benchmark for the shape-based method.

1. Shape-based method – propagated trajectory

In the first method, the initial-guess trajectory is found propagating the initial state by using the acceleration given by the shape-based method. This choice of initial guess has the advantage that state and control are consistent with each other. The drawback of this choice is that the final state constraint is not (in general) fulfilled.

That is, the main task of the optimizer is to change both state and control histories to find a feasible solution that satisfies the end-point constraints.

2. Shape-based method – shaped trajectory

In the second method, the initial-guess trajectory is given by the shape. This choice of initial guess has the advantage that the initial and final values of the state satisfy the end-point constraints. The drawback is that state and control do not fulfil the dynamics. That is, the main task of the optimizer is to change both trajectory and control to satisfy the equations of motion (path constraint).

3. Transversal-thrust law

In the third method, the initial guess is the trajectory given by a pseudo solar sail with a transversal thrust, which is expressed, in the orbital frame $\{\hat{r}, \hat{\boldsymbol{g}}, \hat{h}\}$, as:

$$\boldsymbol{a}_{pss} = a_c \left(\frac{r_{\oplus}}{r} \right)^2 \begin{bmatrix} 0 \\ \pm 1 \\ 0 \end{bmatrix} \quad (6)$$

The \pm sign in Eq. (6) represents a thrust directed either towards or opposite to the transversal unit vector $\hat{\boldsymbol{g}}$ and, therefore, an outward or inward spiral, respectively. Note that a flat solar sail cannot produce the propulsive acceleration of Eq. (6) because the direction of such acceleration would be defined by $\alpha = \pi/2$, whereas its magnitude by $\alpha = 0$. That is, a pseudo solar sail can be seen as a solar sail for which the magnitude of the thrust does not depend on its direction. The trajectory starts at the initial time and state and it is propagated for a defined time of flight t_{0f} .

3.3. Optimization strategy

The optimization strategy implemented within ATOSS consists of two sequential stages. The general idea of the first stage is to find a solution to the OCP by starting with simpler dynamics and eventually solve the problem with the full dynamics. That is, starting from the chosen initial guess, the OCP is solved in three sequential steps to help the convergence of the numerical optimizer. The solution of each step is used as a first guess for the following step. Here and in the remainder of this paper, the term *first guess* is referred to as the solution used to initialize GPOPS-II. For instance, the optimal solution found for the coplanar solar-sail dynamics is used as the first-guess solution within GPOPS-II for solving the following OCP that considers three-dimensional (3-D) dynamics, as shown in Fig. 2. The description of the first stage is given, for a single-phase problem, as follows (Fig. 2):

1. Solve the single-phase optimization problem within a coplanar (2-D) approximation. The pseudo solar-sail model is considered for the description of the acceleration, as:

$$\mathbf{a}_{pSS} = a_c \left(\frac{r_{\oplus}}{r} \right)^2 \hat{\mathbf{N}} \quad (7)$$

2. Solve the single-phase optimization problem by considering the solar-sail acceleration [Eq. (3)] within a coplanar (2-D) approximation.
3. Solve the single-phase optimization problem by considering the 3-D dynamics with the solar-sail acceleration of Eq. (3).

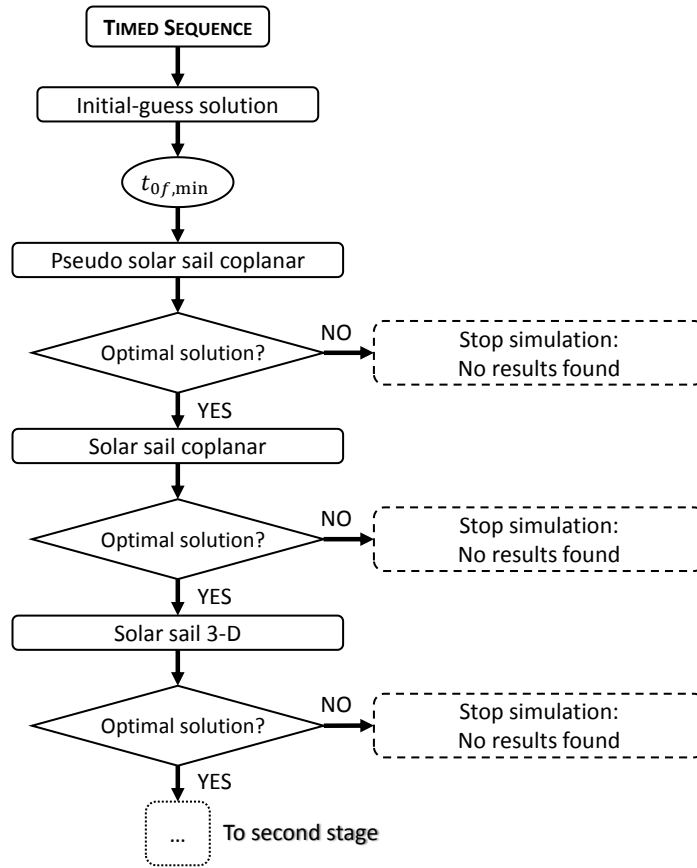


Fig. 2. Flowchart of single-phase ATOSS' optimization strategy for the first stage.

The boundaries on the initial and final time considered in the numerical formulation of each OCP shown in Fig. 2 depend on the first-guess solution. This is implemented so that it is possible to consider an upper limit to the time of flight of a single-phase problem. For what concerns a multi-phase problem, this implementation allows setting the boundaries on initial and final times of two consecutive phases so that the latter phase is not allowed to start before the end of the former. Nevertheless, it has been noted that very often the time of flight of the optimal solution found is the minimum allowed by the problem formulation, which is $t_{of,min}$. A reduction in the lower boundary on the final time by about 50-100 days often resulted in unsuccessful runs of the numerical optimizer.

Therefore, once an optimal solution has been found for the 3-D dynamics, the second stage of the optimization strategy is used.

In the second stage, ATOSS performs a discrete continuation on the lower boundary of the final time, using the previous solution as a first guess for GPOPS-II (Fig. 3). A user-defined value Δt_f is used to decrease the lower boundary on the final time, whereas the boundaries on the initial time are related to the initial time of the first-guess solution. Such continuation is repeated until an unsuccessful run occurs. The solution of the whole optimization is the last optimal solution found. The loop can stop also when a solution has been found if the time of flight is larger than the minimum one allowed.

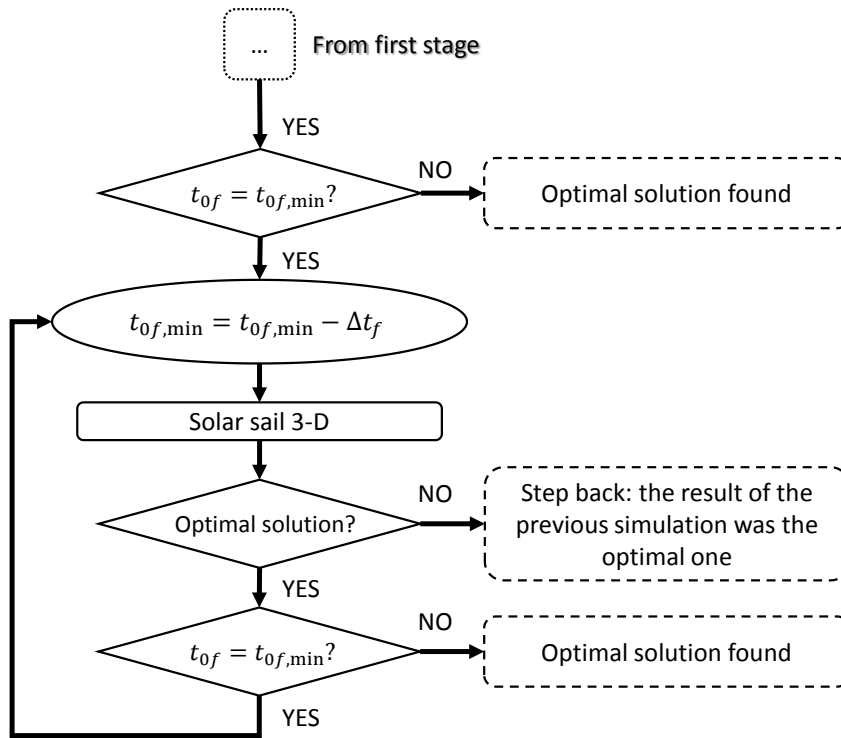


Fig. 3. Flowchart of both single- and multi-phase ATOSS’ optimization strategy for the second stage.

The strategy used for a multi-phase problem is similar. The algorithm begins by optimizing the first phase starting from the selected initial-guess solution and following steps 1) – 3). Once a 3-D solar-sail solution has been found for the first phase, instead of performing the continuation on the lower boundary on the final time, the second phase is added to the problem (Fig. 4). Subsequently, each phase is sequentially added to the previous multi-phase solution (with j phases) so that the updated multi-phase solution (with $(j + 1)$ phases) is computed. The first guess for this new problem is given by patching the solution of the previous optimization with the initial guess of the new phase. The updated multi-phase OCP is therefore solved so that the dynamics considered for the first j phases are the 3-D solar-sail ones, whereas the three-step approach shown in Fig. 2 is considered for the

last phase only. This approach is radically different respect to what was used in [14], in which each phase of the problem was optimized separately and, subsequently, all the single-phase solutions were patched together to build the multi-phase solution. That is, even if the solutions of all phases are optimal, there is no guarantee that the overall multi-phase solution is also optimal. On the contrary, the approach used within ATOSS guarantees the optimality of the entire multi-phase solution. This is in agreement with the Bellman's principle of optimality, which states that "an optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision" [29]. Once a solution for the complete multi-phase problem has been found, the continuation on the lower boundaries on the final time is performed, as shown in Fig. 3. That is, the lower boundary on the final time of each leg is decreased by the selected value Δt_f .

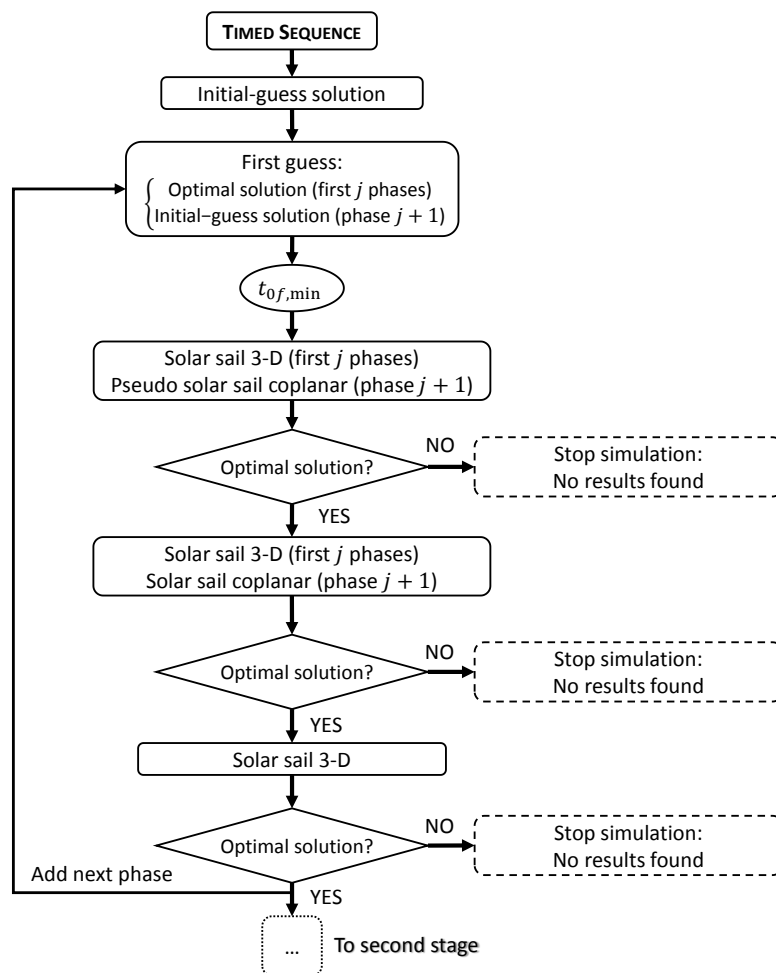


Fig. 4. Flowchart of multi-phase ATOSS' optimization strategy for the first stage.

4. Numerical Test Cases

Several numerical test cases are considered in this work to demonstrate the performances of ATOSS at increasing levels of complexity. The first three test cases aim to validate ATOSS against the literature. In particular, the first two deal with the Earth-Mars and Earth-Venus circular-to-circular orbit transfers shown in [30]. In the third test case, ATOSS is validated on a three-NEA rendezvous against the low-thrust trajectory optimization tool InTrance [31]. The other test cases can be categorized into three optimization campaigns. Campaign 1 deals with single near-Earth asteroid (NEA) rendezvous, whereas multiple NEA rendezvous are considered in Campaign 2. In Campaign 3, ATOSS is used to find solutions to many multiple NEA rendezvous. A solar sail with a characteristic acceleration $a_c = 0.2 \text{ mm/s}^2$ is considered within the three optimization campaigns. Note that this sail characteristic acceleration is within the capability of current and near-term Gossamer sailcraft technology, according to [32, 33]. Therefore, it is a reasonable choice to validate the model on a potential near-term solar-sail mission. Appendix A shows the main properties of the NEAs considered in this work (Table A.1). All the simulations presented in this paper were carried out on a 3.40 GHz Core i7-3770 with 16 GB of RAM and running Windows 7.

4.1. ATOSS validation

1. Circular-to-circular orbit transfers

Quarta and Mengali [30] presented two planar circular-to-circular orbit transfers considering an ideal sail with a characteristic acceleration $a_c = 0.03 \text{ mm/s}^2$. With such a small value of the characteristic acceleration, several complete revolutions are needed for the Earth-Mars and Earth-Venus transfers. This increases the possibility of failure of the numerical optimizer, since a larger number of revolutions might affect the numerical accuracy of the solver. As such, these are interesting test cases to validate ATOSS. The shaped-trajectory variant of the shape-based method (No. 2 in Section 3.2) was used to generate the initial-guess solution and SNOPT was used as NLP solver. The boundaries on L_f for the GA were set as follows. A minimum-time transfer between two circular coplanar orbits is achieved by maximizing the absolute value of the rate of change of the semimajor axis da/dL , which can be expressed, by means of the Lagrange variational equations, as [7]:

$$\frac{da}{dL} = \frac{2a_c r_\oplus^2 a}{\mu} \cos^2 \alpha^* \sin \alpha^* \quad (8)$$

in which a is the semimajor axis and $\alpha^* = \arctan(\sqrt{1/2})$ is the cone angle that maximizes the transversal acceleration [7]. Note that Eq. (8) has been derived in the approximation of circular osculating orbits during the

transfer. This is an acceptable approximation because the transfer is between two circular orbits and the characteristic acceleration is very small. The lower boundary on L_f is set as the value of true longitude \hat{L}_f for which the change in semimajor axis Δa is the desired one, as follows:

$$\Delta a = \int_{L_0}^{\hat{L}_f} \frac{da}{dL} dL \quad (9)$$

The upper boundary on L_f is set so that $L_f \in [\hat{L}_f, \hat{L}_f + 2\pi]$.

ATOSS found a solution for both Earth-Mars and Earth-Venus orbit transfers which are equivalent to those shown in [30]. Specifically, the time of flight for the Earth-Venus transfer found by ATOSS is 3,837 days, which is similar to the 3,844 days of the reference paper. The time of flight for the Earth-Mars transfer found by ATOSS is 8,773 days, which is again equivalent to the 8,800 days of the reference paper. The Earth-Venus and Earth-Mars optimal trajectories are shown in Fig. 5.

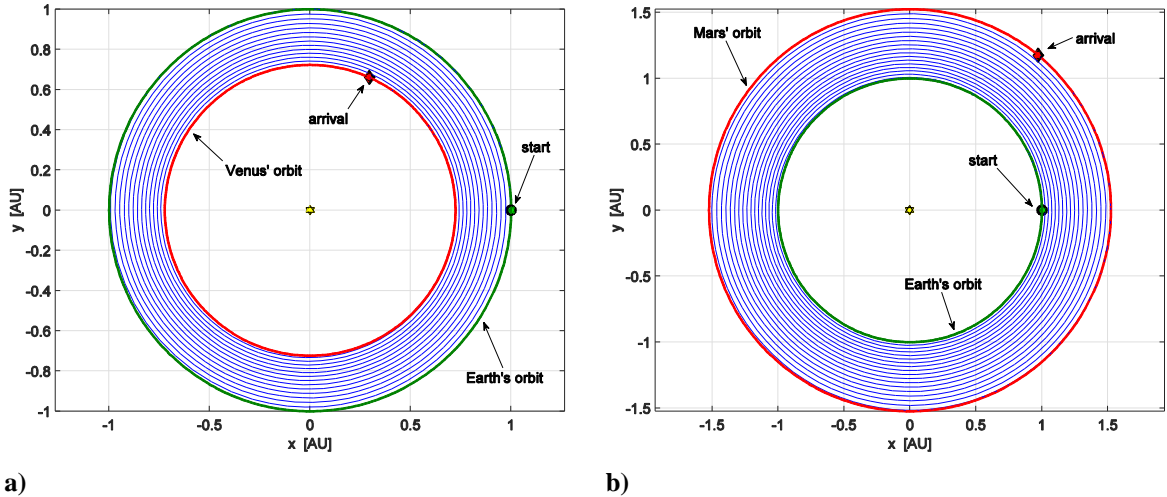


Fig. 5. Optimal orbit transfers in the heliocentric ecliptic reference frame. (a) Earth-Venus. (b) Earth-Mars.

2. Three-NEA rendezvous

Dachwald et al. [32] presented a 3-NEA rendezvous mission through solar sailing, considering a sail with a characteristic acceleration $a_c = 0.3 \text{ mm/s}^2$. ATOSS has been used to find a solar-sail trajectory for the same list of encounter bodies to validate the proposed optimizer. ATOSS has been used considering both a timed and a non-timed sequence as an input. In the first case, the mission parameters of the referenced work were used to produce the timed sequence needed by ATOSS. This has been done to check whether ATOSS can reproduce and refine the trajectory of the referenced work by starting from the same timed sequence. In the second case, instead, only the list of objects and the departing date from the referenced work have been given as an input to ATOSS.

This has been done to check whether ATOSS can find a similar solution to that shown in [32] without any external input. Since the objects in the non-timed sequence are NEAs, it has been chosen to consider only the time of flight given by the law to maximize the rate of change of the inclination to generate the preliminary timeline within ATOSS. In fact, Table A.1 shows that the differences in semimajor axis and eccentricity are small for the objects under consideration, whereas there are significant differences in inclination. For both cases, the shaped-trajectory variant of the shape-based method (No. 2 in Section 3.2) was used to generate the initial-guess solution and SNOPT was used as NLP solver.

The features of the mission are described in Table 1 for the case of the timed sequence used as an input for ATOSS. Figure 6a shows the trajectory of the first leg from Ref. [32], whereas the corresponding trajectory found by ATOSS is shown in Fig. 6b. ATOSS found a solution for which the total mission duration is 8.46 years (i.e. 3,090 days), which is almost a year less than that proposed in Ref. [32]. However, the time spent at the first object is the minimum allowed by the optimizer (i.e. two days). Even though this value of the stay time is in line with the mission requirements, more time might be needed at the object. If this is the case, a slower mission was found in the same optimization run which is one of the intermediate results of the optimization strategy shown in Fig. 3. The characteristics of this mission are shown in Table 2. In this case, the total mission duration is 8.82 years (i.e. 3,220 days), spending 50 days at 2004 GU₉. Such a mission is still approximately 120 days shorter than the reference one. Note that ATOSS gives the possibility to set the minimum allowed stay time at each object. Therefore, if a longer stay time at the object is needed, this can be set and a new optimization run.

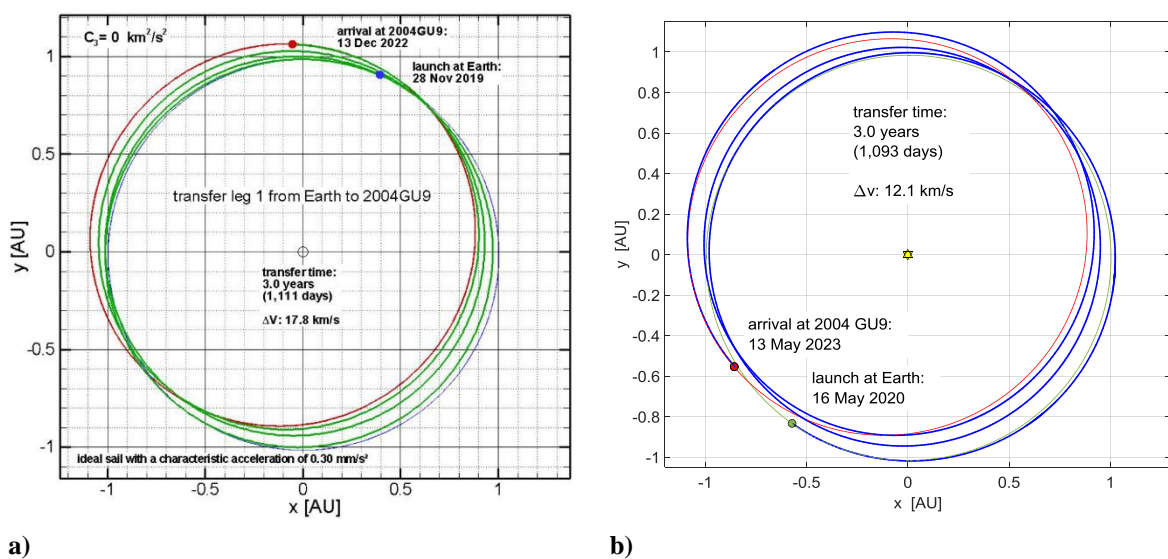


Fig. 6. Trajectory of the first leg from Earth to 2004 GU₉. Heliocentric ecliptic reference frame. (a) Ref. [32]. (b) ATOSS with timed sequence as an input.

Table 1. Mission parameters for the optimal 3-NEA rendezvous (values in brackets are those presented in [32] and used as timed sequence for ATOSS).













Object	Stay time, days		Start	End	Time of flight, days
Earth	---		16 May 2020	13 May 2023	1093
2004 GU ₉	2 (113)		(28 Nov 2019)	(13 Dec 2022)	(1111)
2001 QJ ₁₄₂	133 (90)		15 May 2023 (05 Apr 2023)	28 Jan 2027 (30 Jan 2027)	1354 (1396)
2006 QQ ₅₆	---		10 Jun 2027 (30 Apr 2027)	31 Oct 2028 (24 Jan 2029)	509 (635)

Table 2. Mission parameters for the optimal 3-NEA rendezvous, longer mission (values in brackets are those presented in [32] and used as timed sequence for ATOSS).

Object	Stay time, days		Start	End	Time of flight, days
Earth	---		27 Mar 2020	12 Feb 2023	1053
2004 GU ₉	50 (113)		(28 Nov 2019)	(13 Dec 2022)	(1111)
2001 QJ ₁₄₂	106 (90)		03 Apr 2023 (05 Apr 2023)	30 Jan 2027 (30 Jan 2027)	1398 (1396)
2006 QQ ₅₆	---		16 May 2027 (30 Apr 2027)	19 Jan 2029 (24 Jan 2029)	614 (635)

In the second case, in which a non-timed sequence was used as an input, ATOSS found a solution by self-generating the preliminary timeline for the encounters. The total mission duration found by ATOSS is 8.83 years (i.e. 3,223 days), which is about 120 days less than that shown in [32]. Table 3 shows the characteristics of this mission together with the preliminary timeline self-generated by ATOSS. Note that the sailcraft spends more than 90 days at each object, which is a stay time equivalent to that shown in the referenced paper.

Table 3. Mission parameters for the optimal 3-NEA rendezvous in the case of a non-timed sequence as an input (values in brackets are those self-generated by ATOSS).

Object	Stay time, days		Start	End	Time of flight, days
Earth	---		04 Mar 2020	24 Dec 2022	1026
2004 GU ₉	99 (100)		(28 Nov 2019)	(19 Dec 2022)	(1117)
2001 QJ ₁₄₂	111 (100)		02 Apr 2023 (29 Mar 2023)	23 Oct 2026 (12 Sep 2026)	1300 (1263)
2006 QQ ₅₆	---		11 Feb 2027 (21 Dec 2026)	30 Dec 2028 (04 May 2028)	688 (500)

4.2. Campaign 1: Single NEA rendezvous

Campaign 1 aims to test ATOSS for single-phase OCPs. Moreover, the performances of the shape-based method for solar sailing are assessed against the transversal-thrust law. Eleven single-phase rendezvous are considered as test cases, in which the chosen objects are part of the database used in [14]. This database consists of potentially-hazardous asteroids (PHAs) and the asteroids in the Near-Earth Object Human Space Flight Accessible Target Study (NHATS) [34]. By definition, NHATS asteroids are targets possible to reach from the Earth in a short (less than two years) return mission with low total required Δv . PHAs, on the other hand, are in general characterized by higher Δv if reached from the Earth. For this reason, all test cases considered in this campaign are so that the first object is either Earth or a NHATS asteroid and the second object is a PHA.

Table 4 details the properties of the transfers for each test case as found by means of a sequence-search algorithm, as detailed in [14]. The departing date and the time of flight for each rendezvous are those given by the sequence-search algorithm and are listed in the last two columns of Table 4. The time of flight is considered by ATOSS only if the transversal-thrust law is used as the initial guess, as detailed in Section 3.2. Both inward and outward spirals are considered for the transversal-thrust law, as shown in Eq. (6). Note that, among the chosen PHAs, seven are characterized by an eccentricity $e > 0.2$ and seven are characterized by an inclination $i > 2$ deg (Table A.1).

Table 4. Campaign 1: Properties of the test cases.

Test case	Departing object	Arrival object	Departing date	Time of flight, days
1	Earth	2011 CG ₂	09 Jan 2029	690
2	2005 TG ₅₀	2015 JF ₁₁	26 Jan 2035	718
3	2012 KB ₄	2008 EV ₅	22 Feb 2031	647
4	2009 YF	2002 AW	23 Aug 2035	607
5	2010 WR ₇	1989 UQ	14 Jun 2024	640
6	2005 FG	1999 AQ ₁₀	19 Nov 2027	711
7	2008 TX ₃	2000 EA ₁₄	30 Aug 2028	740
8	2004 JN ₁	2006 KV ₈₉	31 Mar 2035	896
9	2001 QE ₇₁	2001 US ₁₆	06 Dec 2027	697
10	2001 QE ₇₁	2001 US ₁₆	23 Oct 2027	728
11	2015 EF ₇	2000 QK ₁₃₀	17 Apr 2026	531

An optimal solution to all the test cases shown in Table 4 has been sought by using ATOSS with all three available initial-guess approaches discussed in Section 3.2. Moreover, all three available NLP solvers are used. An optimization is considered successful when: a) the NLP solver returns a successful exit flag, b) a further mesh refinement performed by GPOPS-II will not improve the quality of the solution, and c) all the boundaries of time, state and control are verified through a post-processing check. For the sake of conciseness, only the statistical results are shown here. Table 5 shows the number of successful test cases for each initial guess and NLP solver used. The last column of Table 5 shows the number of successful test cases if no distinction in the NLP solver is considered.

Table 5. Campaign 1: Number of successful test cases for each initial guess and NLP solver.

Initial guess	Number of successful test cases			
	IPOPT	SNOPT	WORHP	Any NLP solver
Transversal-thrust law	3/11 (27%)	4/11 (36%)	4/11 (36%)	5/11 (45%)
Shape-based method – propagated trajectory	9/11 (82%)	9/11 (82%)	9/11 (82%)	10/11 (91%)
Shape-based method – shaped trajectory	9/11 (82%)	11/11 (100%)	11/11 (100%)	11/11 (100%)

The overall results show that the use of the shape-based method as an initial-guess solution helps the convergence of GPOPS-II as ten optimal solutions have been found out of the total number of eleven test cases with the shape-based guesses. In fact, an optimal solution has been found for all eleven test cases with the shaped-trajectory variant of the shape-based method. On the contrary, only five solutions have been found using the trajectory obtained through the transversal-thrust law as the initial guess. This validates the shape-based method developed for solar sailing in [14], and endorses its use as an initial guess for a high-fidelity direct-collocation optimization method such as GPOPS-II. The overall statistical results are very similar between the two variants of the shape-based method used if no distinction among the NLP solvers is made. If the number of successful test cases is investigated as a function of both the initial-guess method and the NLP solver used, the shaped trajectory works better than the propagated trajectory: the statistical results show 100% success rate for both SNOPT and WORHP. Nevertheless, Table 5 shows that both variants of the shape-based method, as well as all the available NLP solvers, are robust enough to be used in the early stages of solar-sail trajectory design.

4.3. Campaign 2: Multiple NEA rendezvous

Campaign 2 aims to assess the performances of ATOSS for a multi-phase solar-sail rendezvous. Moreover, the shape-based method for solar sailing is further tested as a method to provide initial-guess solutions in a multi-phase environment as well. In this case, the performances of ATOSS are evaluated based on the total mission duration, which is compared to the results found by means of the optimization strategy discussed in [14].

Two test cases have been chosen to verify the performances of ATOSS. The first one is the five-NEA rendezvous mission shown in [14]. The second test case is the first five-NEA rendezvous mission shown in [24]. The test cases have been optimized using ATOSS with the NLP solver WORHP and the shaped-trajectory variant as the initial-guess solution. In both cases, the sail is injected directly into an interplanetary trajectory at Earth, with zero hyperbolic excess velocity, as done in the referenced papers. It is worth to underline that no results were found by ATOSS with the transversal-thrust law as the initial guess and any of the available NLP solvers.

The parameters of the first optimized test case are summarized in Table 6, whereas Table 7 shows the parameters of the second optimized test case. In these cases, the results of the sequence-search algorithm described in [14] were used as the timed sequences needed by ATOSS. The mission time of the first optimized test case is 3,133 days, whereas the solution shown in [14] for the same sequence described a mission completed in 3,431 days. That is, ATOSS found a multi-phase trajectory that is about 300 days (~10 months) shorter than the reference solution. The mission time of the second optimized test case is 3,183 days, whereas the solution found in [24] for the same sequence showed a mission completed in 3,521 days, which is about 340 days (~11 months) longer than what was found by ATOSS. In both cases, the sailcraft spends more than 90 days in the proximity of each asteroid. Therefore, ATOSS finds better solutions, in terms of total mission duration, to the multiple-NEA-rendezvous problem than the optimization strategy used in the referenced papers [14, 24].

Table 6. Campaign 2, first test case: Results obtained through ATOSS (values in brackets are those of [14]).













Object	Stay time, days		Start	End	Time of flight, days
Earth	---		05 Dec 2025	18 Mar 2027	468
2000 SG ₃₄₄	93 (123)		(10 May 2025)	(26 Feb 2027)	(657)
2015 JD ₃	104 (164)		19 Jun 2027 (29 Jun 2027)	03 Nov 2028 (06 Sep 2028)	503 (436)
2012 KB ₄	94 (160)		16 Feb 2029 (18 Feb 2029)	22 Nov 2030 (24 Sep 2030)	644 (584)
2008 EV ₅	95 (171)		23 Feb 2031 (04 Mar 2031)	22 Nov 2032 (29 Sep 2032)	637 (576)
2014 MP	---		25 Feb 2033 (20 Mar 2033)	04 Jul 2034 (30 Sep 2034)	493 (560)

Table 7. Campaign 2, second test case: Results obtained through ATOSS (values in brackets are those of [24]).

Object	Stay time, days		Start	End	Time of flight, days
Earth	---		10 Apr 2023	20 Aug 2024	497
2012 BB ₁₄	106 (126)		(24 Aug 2022)	(18 Aug 2024)	(725)
2011 CG ₂	91 (123)		04 Dec 2024 (22 Dec 2024)	27 Oct 2026 (03 Oct 2026)	692 (650)
2006 BZ ₁₄₇	107 (166)		26 Jan 2027 (03 Feb 2027)	11 Jan 2029 (21 Nov 2028)	715 (658)
2013 BS ₄₅	124 (188)		28 Apr 2029 (07 May 2029)	05 Sep 2030 (23 Jun 2030)	495 (412)
2014 YN	---		07 Jan 2031 (28 Dec 2030)	27 Dec 2031 (13 Apr 2032)	354 (473)

Lastly, it is interesting to study the evolution of the total mission duration during the second stage of the optimization strategy implemented within ATOSS. As detailed in Section 3.3, once the entire multi-phase 3-D trajectory has been found, a discrete continuation on the lower boundary on the final time is carried out to find a solution with a lower total mission duration. Figure 7 shows the evolution of the total mission duration within ATOSS for the two test cases considered in this section. The total cumulative computational time needed to find a solution is shown along the horizontal axis. Each point in the figure corresponds to a solution found by ATOSS in each intermediate step. The first point of each curve is related to the mission duration of the entire trajectory obtained by patching together the initial-guess trajectories for each transfer leg. Note that the significantly longer time related to the second point is due to ATOSS finding a 3-D multi-phase trajectory starting from the solutions of the shape-based method, which is the first stage of the optimization strategy described in Section 3.3. In

conclusion, Fig. 7 demonstrates the importance of the continuation stage implemented within ATOSS, which decreases the total mission duration by 10% respect to the value of the initial guess. In fact, a gain of 6% of total mission duration is achieved during the second stage of the optimization strategy developed within ATOSS (i.e. between the last and the second point of each curve shown in Fig. 7).

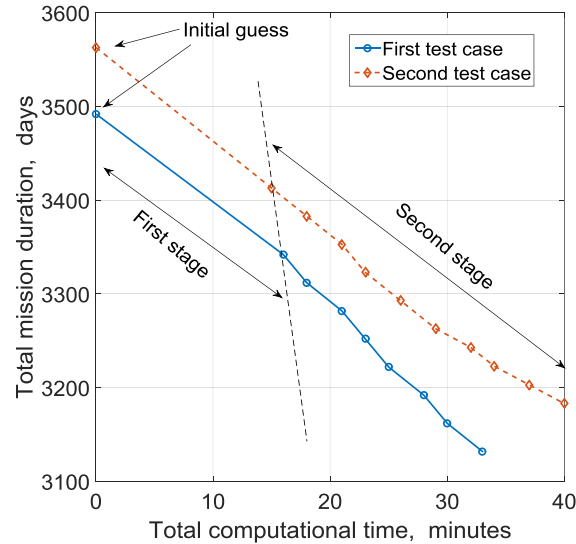


Fig. 7. Campaign 2: Evolution of the total mission duration within ATOSS.

4.4. Campaign 3: Automated optimization

Campaign 3 aims to assess the performances of ATOSS as an automated multi-phase optimizer. To do so, all the 589 preliminary sequences with at least three encounters and at least one PHA found for the second database discussed in [24] are optimized by means of ATOSS. The number of optimal solutions found and the computational time needed for the entire automated optimization campaign are the two performance metrics used for the performance test. To carry out a fair comparison, the NLP solver SNOPT is used, as done in the referenced paper.

Once the initial settings are fixed at the beginning of the optimization campaign, all the simulations are carried out with the same settings and with no user intervention. It is likely that some of the results found can be improved by choosing *ad hoc* settings for each test case so that ATOSS and GPOPS-II are tuned for that specific scenario [35]. Nevertheless, this is beyond the scope of this work, which aims to test the capability of ATOSS to find as many optimal solutions as possible automatically. Once solutions for all the test cases have been found, the user can further optimize the most interesting ones. The results of the automated optimization campaign are schematically shown in Table 8 and compared with the results shown in the referenced paper.

Table 8. Campaign 3: Results and comparison with [24].

Optimizer	Number of solutions found	Success rate	Computational time
ATOSS	390	66%	< 6 days
Ref. [24]	343	58%	~ 30 days

ATOSS found fully-optimized solar-sail trajectories for 390 sequences, i.e. 66% of the total number of test cases. On the other hand, Ref. [24] shows that a solution was found for 343 sequences, i.e. 58% of the total number of test cases. Moreover, the entire automated optimization campaign took less than six days of computational time, whereas the solutions showed in [24] were found in about thirty days on the same machine. An estimate of the quality of the solutions can be obtained by analyzing the total mission durations found by the two optimizers. This is done only on those test cases (292) for which both optimizers found a solution. Two solutions are considered equivalent if the difference in their mission durations is within twenty days. An analysis of the numerical results showed that in 285 test cases (~98%) the solutions found by ATOSS are shorter than those in [24]. On the other hand, the optimizer in Ref. [24] found a shorter solution than ATOSS in only one test case; the two optimizers found equivalent solutions in the remaining six test cases (~2%). These results demonstrate the reliability of ATOSS as an automated optimizer. Moreover, the comparison with the results shown in the referenced paper proves that ATOSS outperforms the previous optimization strategy in success rate, computational speed and quality of the solutions found.

5. Conclusions

In this paper, we introduced and discussed ATOSS, an Automated Trajectory Optimizer for Solar Sailing. ATOSS was developed as a software tool to find solar-sail trajectories in an automated way. Moreover, ATOSS was used to validate the shape-based method for solar sailing proposed in a previous work by the authors. Such validation was performed against a transversal-thrust law on several test cases. The overall results demonstrated the quality of the shape-based method as an initial guess for a direct-collocation optimization method. Planetary orbit transfers, single and multiple near-Earth asteroid (NEA) rendezvous were chosen as test cases to prove the reliability of ATOSS as a single- and multi-phase trajectory optimizer for solar sailing. The solutions of ATOSS were assessed against those found by the low-thrust trajectory optimization tool InTrance. Moreover, the results of an existing optimization strategy shown in previous papers by the authors were used as test beds for ATOSS. This study demonstrated that ATOSS outperforms the existing optimization strategy, being able to find better results in terms of total mission duration, while guaranteeing a time between two consecutive transfers long

enough to carry out studies on the target asteroid. Lastly, an automated optimization campaign was carried out to find the solutions for several solar-sail trajectories. As many as 589 optimizations were performed in a completely automated way by ATOSS. The results showed a 66% success rate with 390 optimal solutions found. This number is larger than what was found in a previous study and it demonstrates the reliability of ATOSS in finding solar-sail trajectories without the need of any input by the user. This is a key feature of ATOSS, especially considering that the numerical optimization is very often sensitive to the optimization parameters. Moreover, the solutions found by ATOSS in the automated optimization campaign are better, in terms of total mission duration, than the ones found in the previous study. Lastly, ATOSS allows setting a minimum desired stay time at each target object. This feature can be fundamental in the mission design of a multiple NEA rendezvous, for which a minimum time for close-up observation might be needed by the onboard instruments.

Conflict of interest statement

The authors declare that there is no conflict of interest regarding this work.

Acknowledgments

Alessandro Piloni gratefully acknowledges support for this research from the Engineering and Physical Sciences Research Council for funding his research under the James Watt sponsorship program, award number 1370838. This work was conducted at the Department of Mechanical and Aerospace Engineering at the University of Florida within a four-month visiting period. Alessandro Piloni gratefully thanks the University of Florida for hosting him. His visit was supported by the Royal Society of Edinburgh under the John Moyes Lessells travel scholarship and by the University of Glasgow under both the Jim Gatheral travel scholarship and the David Fleming-Brown travel scholarship. The authors thank the two anonymous reviewers who helped improving the final version of this manuscript with their feedback and suggestions.

Appendix A. Properties of the NEAs considered.

Table A.1. Properties of the objects considered¹.

Object	Semimajor axis, AU	Eccentricity	Inclination, deg	PHA	NHATS
Earth	1	0	0	no	no
1989 UQ	0.915	0.265	1.299	yes	no
1999 AQ ₁₀	0.934	0.236	6.501	yes	no
2000 EA ₁₄	1.117	0.202	3.555	yes	no
2000 QK ₁₃₀	1.181	0.262	4.722	yes	no
2000 SG ₃₄₄	0.977	0.067	0.111	no	yes
2001 QE ₇₁	1.078	0.159	3.035	no	yes
2001 QJ ₁₄₂	1.062	0.086	3.103	no	yes
2001 US ₁₆	1.356	0.253	1.905	yes	no
2002 AW	1.071	0.256	0.571	yes	no
2004 GU ₉	1.001	0.136	13.650	yes	no
2004 JN ₁	1.085	0.176	1.499	no	yes
2005 FG	1.122	0.213	3.882	no	yes
2005 TG ₅₀	0.923	0.135	2.402	no	yes
2006 BZ ₁₄₇	1.024	0.099	1.410	no	yes
2006 KV ₈₉	1.150	0.273	3.554	yes	no
2006 QQ ₅₆	0.985	0.046	2.796	no	no
2008 EV ₅	0.958	0.084	7.437	yes	yes
2008 TX ₃	1.180	0.187	2.180	no	yes
2009 YF	0.936	0.121	1.523	no	yes
2010 WR ₇	1.046	0.235	1.563	no	yes
2011 CG ₂	1.178	0.158	2.757	yes	yes
2012 BB ₁₄	1.064	0.099	2.645	no	yes
2012 KB ₄	1.093	0.061	6.328	no	yes
2013 BS ₄₅	0.994	0.084	0.773	no	yes
2014 MP	1.050	0.029	9.563	no	yes
2014 YN	0.891	0.136	1.243	no	yes
2015 EF ₇	1.209	0.225	3.570	no	yes
2015 JD ₃	1.058	0.009	2.730	no	yes
2015 JF ₁₁	1.073	0.111	5.400	yes	no

¹<https://cneos.jpl.nasa.gov/orbits/elements.html> (cited 08 August 2015).

References

- [1] Macdonald, M., Hughes, G. W., McInnes, C. R., Lyngvi, A., Falkner, P. and Atzei, A., “Solar Polar Orbiter: A Solar Sail Technology Reference Study”, *Journal of Spacecraft and Rockets*, Vol. 43, No. 5, 2006, pp. 960-972.
doi: 10.2514/1.16408
- [2] Ceriotti, M. and McInnes, C. R., “Generation of Optimal Trajectories for Earth Hybrid Pole Sitters”, *Journal of Guidance, Control, and Dynamics*, Vol. 34, No. 3, 2011, pp. 847-859.
doi: 10.2514/1.50935
- [3] Gong, S., Li, J. and Baoyin, H., “Solar sail transfer trajectory from L1 point to sub-L1 point”, *Aerospace Science and Technology*, Vol. 15, No. 7, 2011, pp. 544-554.
doi: 10.1016/j.ast.2010.10.003
- [4] Bolle, A. and Circi, C., “Modified Sun-Synchronous Orbits by Means of Solar Sails”, *Recent Patents on Space Technology*, Vol. 1, 2011, pp. 72-79.
doi: 10.2174/1877611611101010072
- [5] Circi, C., “Simple Strategy for Geostationary Stationkeeping Maneuvers Using Solar Sail”, *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 2, 2005, pp. 249-253.
doi: 10.2514/1.6797
- [6] Mengali, G., Quarta, A. A., Romagnoli, D. and Circi, C., “H2-reversal trajectory: A new mission application for high-performance solar sails”, *Advances in Space Research*, Vol. 48, No. 11, 2011, pp. 1763-1777.
doi: 10.1016/j.asr.2010.11.037
- [7] McInnes, C. R., *Solar Sailing: Technology, Dynamics and Mission Applications*, Springer Praxis Publishing, Chichester, UK, 1999, pp. 32-55, 112-120, 136-148.
- [8] Petropoulos, A. E. and Longuski, J. M., “Shape-Based Algorithm for the Automated Design of Low-Thrust, Gravity Assist Trajectories”, *Journal of Spacecraft and Rockets*, Vol. 41, No. 5, 2004, pp. 787-796.
doi: 10.2514/1.13095
- [9] De Pascale, P. and Vasile, M., “Preliminary Design of Low-Thrust Multiple Gravity-Assist Trajectories”, *Journal of Spacecraft and Rockets*, Vol. 43, No. 5, 2006, pp. 1065-1076.
doi: 10.2514/1.19646
- [10] Wall, B. J. and Conway, B. A., “Shape-Based Approach to Low-Thrust Rendezvous Trajectory Design”, *Journal of Guidance, Control, and Dynamics*, Vol. 32, No. 1, 2009, pp. 95-102.
doi: 10.2514/1.36848
- [11] Taheri, E. and Abdelkhalik, O., “Initial three-dimensional low-thrust trajectory design”, *Advances in Space Research*, Vol. 57, No. 3, 2016, pp. 889-903.
doi: 10.1016/j.asr.2015.11.034
- [12] Zeng, K., Geng, Y. and Wu, B., “Shape-based analytic safe trajectory design for spacecraft equipped with low-thrust engines”, *Aerospace Science and Technology*, Vol. 62, 2017, pp. 87-97.
doi: 10.1016/j.ast.2016.12.006
- [13] Niccolai, L., Quarta, A. A. and Mengali, G., “Solar sail trajectory analysis with asymptotic expansion method”, *Aerospace Science and Technology*, Vol. 68, 2017, pp. 431-440.

doi: 10.1016/j.ast.2017.05.038

- [14] Peloni, A., Ceriotti, M. and Dachwald, B., “Solar-Sail Trajectory Design for a Multiple Near-Earth-Asteroid Rendezvous Mission”, *Journal of Guidance, Control, and Dynamics*, Vol. 39, No. 12, 2016, pp. 2712-2724.
doi: 10.2514/1.G000470
- [15] Patterson, M. A. and Rao, A. V., “GPOPS – II: A MATLAB Software for Solving Multiple-Phase Optimal Control Problems Using hp-Adaptive Gaussian Quadrature Collocation Methods and Sparse Nonlinear Programming”, *ACM Transactions on Mathematical Software*, Vol. 41, No. 1, 2014, Article 1.
doi: 10.1145/2558904
- [16] Büskens, C. and Wassel, D., “The ESA NLP Solver WORHP”, *Modeling and Optimization in Space Engineering*, edited by G. Fasano and J.D. Pinter, Springer Optimization and Its Applications, Springer-Verlag, New York, NY, USA, 2012, pp. 85-110.
- [17] Rao, A. V., “Survey of Numerical Methods for Optimal Control”, *2009 AAS/AIAA Astrodynamics Specialist Conference*, AAS Paper 09-334, Pittsburgh, PA, USA, 2009.
- [18] Porsa, S., Lin, Y.-C. and Pandya, M. G., “Direct Methods for Predicting Movement Biomechanics Based Upon Optimal Control Theory with Implementation in OpenSim”, *Annals of Biomedical Engineering*, Vol. 44, No. 8, 2016, pp. 2542-2557.
doi: 10.1007/s10439-015-1538-6
- [19] Van den Bogert, A. J., Blana, D. and Heinrich, D., “Implicit methods for efficient musculoskeletal simulation and optimal control”, *Procedia IUTAM*, Vol. 2, 2011, pp. 297-316.
doi: 10.1016/j.piutam.2011.04.027
- [20] De Groote, F., Kinney, A. L., Rao, A. V. and Fregly, B. J., “Evaluation of Direct Collocation Optimal Control Problem Formulations for Solving the Muscle Redundancy Problem”, *Annals of Biomedical Engineering*, Vol. 44, No. 10, 2016, pp. 2922-2936.
doi: 10.1007/s10439-016-1591-9
- [21] Graham, K. F. and Rao, A. V., “Minimum-Time Trajectory Optimization of Multiple Revolution Low-Thrust Earth-Orbit Transfers”, *Journal of Spacecraft and Rockets*, Vol. 52, No. 3, 2015, pp. 711-727.
doi: 10.2514/1.a33187
- [22] Liu, F., Hager, W. W. and Rao, A. V., “Adaptive mesh refinement method for optimal control using nonsmoothness detection and mesh size reduction”, *Journal of the Franklin Institute*, Vol. 352, No. 10, 2015, pp. 4081-4106.
doi: 10.1016/j.jfranklin.2015.05.028
- [23] Patterson, M. A., Hager, W. W. and Rao, A. V., “A hp Mesh Refinement Method for Optimal Control”, *Optimal Control Applications and Methods*, Vol. 36, No. 4, 2014, pp. 398-421.
doi: 10.1002/oca.2114
- [24] Peloni, A., Dachwald, B. and Ceriotti, M., “Multiple Near-Earth Asteroid Rendezvous Mission: Solar-Sailing Options”, *Advances in Space Research*, (in press).
doi: 10.1016/j.asr.2017.10.017
- [25] Walker, M. J. H., Ireland, B. and Owens, J., “A set of modified equinoctial orbit elements”, *Celestial Mechanics*, Vol. 36, No. 4, 1985, pp. 409-419.
doi: 10.1007/bf01227493

- [26] Walker, M. J. H., “A set of modified equinoctial orbit elements” (Errata), *Celestial Mechanics*, Vol. 38, No. 4, 1986, pp. 391-392.
doi: 10.1007/bf01238929
- [27] Novak, D. M. and Vasile, M., “Improved Shaping Approach to the Preliminary Design of Low-Thrust Trajectories”, *Journal of Guidance, Control, and Dynamics*, Vol. 34, No. 1, 2011, pp. 128-147.
doi: 10.2514/1.50434
- [28] Betts, J. T., *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming (Second Edition)*, Soc. for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2010, pp. 265-267.
- [29] Bellman, R., *Dynamic Programming*, Princeton University Press, Princeton, NJ, USA, 1957, pp. 83.
- [30] Quarta, A. A. and Mengali, G., “Semi-Analytical Method for the Analysis of Solar Sail Heliocentric Orbit Raising”, *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 1, 2012, pp. 330-335.
doi: 10.2514/1.55101
- [31] Dachwald, B., “Low-Thrust Trajectory Optimization and Interplanetary Mission Analysis Using Evolutionary Neurocontrol”, PhD Thesis, Institut für Raumfahrttechnik, Universität der Bundeswehr München, Munich, Germany, 2004.
- [32] Dachwald, B., Boehnhardt, H., Broj, U., Geppert, U., Grundmann, J. T., Seboldt, W., et al., “Gossamer Roadmap Technology Reference Study for a Multiple NEO Rendezvous Mission”, *Advances in Solar Sailing*, edited by M. Macdonald, Springer Praxis Books, Springer-Verlag, Berlin, Germany, 2014, pp. 211-226.
- [33] Grundmann, J. T., Boden, R. C., Ceriotti, M., Dachwald, B., Dumont, E., Grimm, C. D., et al., “Soil to Sail - Asteroid Landers on Near-Term Sailcraft as an Evolution of the GOSSAMER Small Spacecraft Solar Sail Concept for In-Situ Characterization”, *5th IAA Planetary Defense Conference - PDC 2017*, International Academy of Astronautics Paper IAA-PDC-17-05-19, Tokyo, Japan, 2017.
- [34] Barbee, B. W., Esposito, T., Pinon, E., Hur-Diaz, S., Mink, R. G. and Adamo, D. R., “A Comprehensive Ongoing Survey of the Near-Earth Asteroid Population for Human Mission Accessibility”, *AIAA Guidance, Navigation and Control Conference*, AIAA Paper 2010-8368, Toronto, Canada, 2010.
- [35] Wolpert, D. H. and Macready, W. G., “No Free Lunch Theorems for Optimization”, *IEEE Transactions on Evolutionary Computation*, Vol. 1, No. 1, 1997, pp. 67-82.
doi: 10.1109/4235.585893