DOI: 10.13009/EUCASS2022-4923

An investigation on shape-based methods with different polynomial basis for low-thrust mission design

Pablo Solano-López, Sergio Cuevas del Valle*, Alondra Solá Molina* and Hodei Urrutxua Cereijo**

**Aerospace Systems and Transport Research Group (GISAT-ASTRG) | Universidad Rey Juan Carlos, Camino del Molino 5, 28943, Fuenlabrada, Madrid, Spain.*

Abstract

Shape-based methods have gained increasing attention within the astronautical community in recent times. Despite their successful application to low-thrust trajectory design, their mathematical formulation has not been completely explored, nor their possible numerical and computational. This works presents several aspects in the formulation of these shape-based methods as general optimal control solvers. First, new polynomial families are introduced and compared as key elements in the presented methodology. Additionally, polynomial nodes are used to construct the optimization grid to enhance the numerical performance of the algorithm. Finally, the proposed optimal control solver is evaluated on several testbench low-thrust transfers missions.

1. Introduction

Shape-based methods have gained increasing attention within the astronautical community in recent times, with extensive applications within optimization problems. The rationale behind such methodologies lies in exploiting particular functions to ease the representation of the orbital motion of the system, typically a spacecraft. Such analytical expressions, usually obtained by imposing boundary or joint conditions of the discretized motion problem, enable a quick and fast generation of preliminary trajectories. These preliminary results are then used for mission design trajectories or initial guesses, which may undergo further refinement within more complex optimization solvers. Clearly, the functions used to represent the trajectory at hands is a key element of the methodology: their mathematical properties are then inherited by the numerical algorithm and, consequently, better convergences and more complex dynamics will be enabled by a proper choice.

Shape-based methods for trajectory design were first introduced by Petropoulos and Longuski [1] by selecting an exponential sinusoid function to describe the trajectory of a low thrust accelerated spacecraft. Thereafter, sinusoids as function family have been a traditional choice to represent spacecraft dynamics for other applications [2]. Wall and Conway [3] presented inverse polynomials to match the spacecraft boundary conditions and its intrinsic dynamics. More recently, Xie et al. [4] suggested a rapid shaping method based on the radial coordinate form of the initial and target orbits, and Roa et al. [5] introduced the concept of generalized logarithmic spirals in a series of works.

Orthogonal polynomial bases, as well as general spirals, are a common design tool for direct transcription optimal control solvers and numerical approximation across a wide range of fields, from Fluid Mechanics to Astrodynamics. However, they are not often selected to construct shape-based methods for orbital mechanics applications. More recently, Taheri [6] introduced a shape-based formulation to describe spacecraft trajectories based on a finite Fourier series. Based on Taheri's work, Hou et al. [7,8] presented a shape-based method to design the 3D trajectories of electric solar wind sails, relying on a Bézier curves approximation, which builds upon the family of Bernstein polynomials. Despite their successful application to this low-thrust trajectory design optimization, their mathematical formulation was not completely explored to its full potential, nor were their numerical and computational advantages fully exploited. In addition, despite some recent work on optimal control [9] general shape-based methods have not been employed as a representative tool for optimization engines, but as a low-cost, fast technique to generate dynamically compliant trajectories to be refined afterwards in more detailed design phases.

This work presents a novel approach to tackle optimization problems in astrodynamics using enhanced shape-based methods, together with an assessment of the viability of these algorithms as general optimization solvers. New functional representations of the system's time evolution are introduced, by means of an orthogonal version of the Bernstein polynomials family, to enhance the classical algorithm's numerical behaviour and improve on its

DOI: 10.13009/EUCASS2022-4923

convergence properties. Additionally, a direct performance comparison is performed and presented against classical orthogonal bases, which have still not been employed in this methodology. The optimization-associated collocation problem is then reformulated on the natural nodes of the selected functional bases. The proposed scheme is directly applied in the formulation of generic low-thrust orbital transfers. Finally, several benchmark missions of interest are solved by the proposed techniques for demonstration purposes, introducing metaheuristic algorithms for determining the overall formulation of the algorithm.

The rest of this document is structured as follows: Section II presents the shape-based methodology with its general mathematical formalism. This is later applied to specific low-thrust, time-free and time-fixed orbital transfer problems in Section III, testing its capabilities. Section IV is devoted to present and solve testbench missions, solved under various constructions of our algorithm, while introducing genetic algorithms in combination with our shape-based approach to find supreme, Pareto front multi-objective optimal transfer solutions. Finally, conclusions and further research are discussed in Section V.

2. Shape-based methods as an optimization methodology

This section develops the shape-based approach as a general solver for optimal control problems, expanding the details and introducing the novelties of the formulation when compared to previous work [7,8].

Shape-based methods are semi analytical techniques to solve general optimization Bolza Problems of the form

$$\begin{aligned}
&\min \; J = \mathcal{M}\left[\boldsymbol{s}(0),\boldsymbol{s}(t_f),t_0,t_f\right] + \int_{t_0}^{t_f} \mathcal{L}\left[\boldsymbol{s}(t),\boldsymbol{u}(t_f),t\right]dt \\
&s.\,t \qquad \dot{\boldsymbol{s}} = \boldsymbol{f}\left[\boldsymbol{s}(t),\boldsymbol{u}(t),t,\boldsymbol{\beta}\right], \qquad \forall t\in\left[t_0,t_f\right] \\
&\qquad\quad \boldsymbol{h}\left[\boldsymbol{s}(t),\boldsymbol{u}(t_f),t\right] = 0 \\
&\qquad\quad \boldsymbol{g}\left[\boldsymbol{s}(t),\boldsymbol{u}(t_f),t\right] \le 0, \qquad \forall t\in\left[t_0,t_f\right]
\end{aligned} \tag{1}$$

where the state of the dynamical system is described by the vector $\boldsymbol{s}(t)$ and whose first-order evolution with respect to the independent variable t is governed by the vector field $\boldsymbol{f}\left[\boldsymbol{s}(t),\boldsymbol{u}(t),t,\boldsymbol{\beta}\right]$ with $\boldsymbol{\beta}$ a set of parameters and $\boldsymbol{u}(t)$ is the control vector field.

The optimal solution is given by the determination of the phase space flow $\boldsymbol{s}^*(t)$ and control application $\boldsymbol{u}^*(t)$ minimizing the cost function J while satisfying the boundary conditions equalities $\boldsymbol{h}\left[\boldsymbol{s}(t),\boldsymbol{u}(t_f),t\right]$ and path constraints $\boldsymbol{g}\left[\boldsymbol{s}(t),\boldsymbol{u}(t_f),t\right]$.

To reach that solution, the proposed algorithm is analogous to direct transcription methods, *e.g.* [13], in which the infinite dimensional Bolza problem is discretized into a Non-Linear Programming Problem (NLP), where the evaluation of the discrete cost function is directly optimized [10].

Consequently, a discrete independent variable grid is defined, $T = \{t_i\}$, at which the cost function, the path constraints and the boundary inequalities are evaluated. As for the shape-based approach, the state vector evolution is then prescribed by setting a pre-determined functional shape and then computing it at the very same grid. Finally, regarding the dynamic constraints, they are intrinsically imposed by computing the control law $\boldsymbol{u}(t_i)$ as their residual to the state evolution derivatives.

The optimization or decision variables are usually defined as the set of parameters that will determine the prescribed shape of the state's phase space trajectory. The process is presented hereafter:

DOI: 10.13009/EUCASS2022-4923

Firstly, the state vector is projected onto some polynomial basis P_i , either orthogonal or non-orthogonal and computed at the temporal mesh,

$$s(t_i) = \sum_{k=0}^{N_i} c_k P_k(t_j) \tag{2}$$

The polynomial order N_i may be different for each of the state vector components in a general formulation, nevertheless, for the rest of this work the contrary will be assumed to ease the notation. In case a wider scope is required, without difficulty this condition can be relaxed to allow for different practical order expansions.

From Eq. (2), once the basis is defined, the derivatives of the state vector with respect to the independent variable are clearly available. As a result, in the studied NLP problem, the set of decision variables will be primarily composed of the coordinates c_k , while other options could also be included.

From this expression, the control law evaluated at the discrete grid $u(t_i)$ can be computed as

$$\boldsymbol{u}(t_j):\ \dot{\boldsymbol{s}}(t_j) - \boldsymbol{f}\left[\boldsymbol{s}(t_j), \boldsymbol{u}(t_j), t_j, \boldsymbol{\beta}\right] = 0 \tag{3}$$

The specific difference of our approach with respect to other methodologies is the inclusion of the boundary conditions in the prescribed functional/polynomial shape of the trajectory. Indeed, using such a semi analytic approach, the optimization solver needs only to handle path constraints. This is achieved by defining the symmetric set $\mathcal{B} = \{c_1, c_2, c_{N-1}, c_N\}$ of polynomial coefficients and then fixing it for boundary conditions through the following linear system:

$$\begin{aligned} \sum_{c_k \in \mathcal{B}} \boldsymbol{c_k} P_k(0) &= \boldsymbol{s}(0) - \sum_{c_k \notin \mathcal{B}} \boldsymbol{c_k} P_k(0) \\ \sum_{c_k \in \mathcal{B}} \boldsymbol{c_k} P_k(t_f) &= \boldsymbol{s}(t_f) - \sum_{c_k \notin \mathcal{B}} \boldsymbol{c_k} P_k(t_f) \end{aligned} \tag{4}$$

The decision variable set to be optimized is therefore reduced to $\boldsymbol{x} = \{c_3, c_4, c_5, ..., c_{N-4}, c_{N-3}, c_{N-2}\}$ together with additional degrees of freedom, such as the time of flight. For Cauchy Two-Boundary Value Problems (TBVP), a minimum third-order polynomial expansion is needed to allow free degrees of freedom to be determined.

For a given TBV Bolza problem of the form of Equation (1), the complete shape-based optimization procedure is summarized in the following scheme:

1. Define the discrete sampling grid $T = \{t_i\}$ in the interval $[t_0, t_f]$. For time-free problems, a domain rescaling can be used to include the final time as an optimization variable.
2. Generate an initial guess $\varsigma(t_i)$ for the optimal phase space flow using a densified sampling grid and Equation (2) for a third-order expansion of the state vector, to analytically include boundary conditions in it.
3. Compute the initial guess for the decision variable $\boldsymbol{x}$ using a least-squares method on $\varsigma(t_i)$.
4. Optimize the decision variable vector $\boldsymbol{x}$ using an NLP solver. At each iteration,
   a. Solve the set $\mathcal{B}$ through Equation (4) to impose boundary conditions.
   b. Compute the control law using Equation (3) at the sampling grid.
   c. Minimize the cost function J in Equation (1) over the sampling grid while respecting path constraints.

DOI: 10.13009/EUCASS2022-4923

5. Once solved, obtain the optimal state evolution and the final control law through Equation (2) and (3), respectively.

Standard NLP solver algorithms suffice to compute the optimal solution of the proposed problem via the shape-based approach. In this work, all examples have been computed using the Sequential-Quadratic Programming (SQP) algorithm built in Matlab's fmincon function [14,15].

2.1 Orthogonal polynomial families for phase space flow projection

As it was introduced in Section 1, up to now, little work has been dedicated to the use of orthogonal polynomial families to express the evolution of the system's state vector in shape-based methods [6], despite their vast experience in direct transcription methods and numerical optimization. The latest work on shape-based methods, by Huo et al. [7,8], focused on Bernstein polynomials to model the motion of spacecraft in low-thrust transfers, which are precisely non-orthogonal.

They present a $n+1$ Bernstein basis polynomials of degree n are defined as

$$P_{i,n}(\tau) = \binom{n}{i} \tau^i (1-\tau)^{n-i} \tag{5}$$

Bernstein polynomials are naturally interesting for TVBP problems, as they show roots at both [0,1], allowing for analytical expressions to impose boundary conditions on the optimal trajectory to be solved for. They also show natural smoothness and approximation properties that proved to be interesting for low-thrust trajectory design.

However, the classical orthogonal families usually show enhanced numerical properties within optimization techniques, and consequently, they will be studied through this work. Specifically, Chebyshev, Legendre, Hermite and Laguerre polynomials are presented. All of them require an appropriate mapping to their domain of definition $\varphi : [t_0, t_f] \to \mathcal{D}$.

Moreover, this work also introduces an additional orthogonal set of polynomials, the orthogonal Bernstein family [11], which are defined as

$$P_{i,n}(\tau) = \left[2(n-i)+1\right]^{1/2} \sum_{k=0}^{i} (-1)^k \frac{\binom{2n+1-k}{i-k}\binom{i}{k}}{\binom{n-k}{i-k}} P_{i-k,n-k}(\tau) \tag{6}$$

They are introduced not only for completion and comparison with previous works, but also to show their numerical enhanced properties when compared to their non-orthogonal version. Although there is a clear increase in mathematical complexity, they still retain the characteristic smoothness of classical Bernstein polynomials.

2.2 The nodal sampling distribution grid

As already discussed, the continuous problem is evaluated and solved in the discrete set of sampling points $T = \{t_i\}$, as in direct transcription methods. The selection of these points may follow any distribution of interest as they will modify consequently the error distribution of the trajectory discretization. Being the functional approach only a part of the optimization process, as it will be shown this error tends to be of second order compared to optimization algorithm errors, and so, they are a secondary problem to be studied. In this work, linear, normal, and random distributions have been standardly used.

One of the main novelties of this work is the introduction of the nodal sampling distribution as a design parameter, based on the natural roots of the state vector polynomial expansion basis. The nodal distribution is defined as the disjunct union of node sets for each polynomial in the expansion

$$T = \bigcup \left\{ \tau_{k,j} \mid P_k\left(\tau_{k,j}\right) = 0 \right\} \quad \textbf{(7)}$$

Indeed, the use of the nodal distributions allows to analytically quantify the approximation error of the polynomial family, given classical Approximation Theory results, and enhance the numerical behaviour of the expansion when compared to other distributions. Moreover, they intrinsically exploit the complete mathematical domain at which the polynomial family is defined. For both the Laguerre and Hermite families, that are supported in an open domain, an approximation to their nodes is achieved by means of a bijective mapping of the Legendre nodes as an initial guess for a Newton-Rhapson method.

3. Shape-based methods for low-thrust orbital transfers

This section develops the application of the presented methodology to optimal boundary value control problems in Orbital Mechanics, specifically continuous-thrusted orbital transfers or rendezvous problems. In fact, as already discussed, the shape-based approach was developed as a fast method to generate accurate initial guesses for these problems, to be later used in more sophisticated mission design and optimization phases.

In particular, the shape-based approach applies to both fixed initial and final boundary conditions problems, including both time-free and time-fixed transfers.

Particularising Equation (1) for this case requires the definition of a dynamical system and state vector that will completely define the evolution of the system with respect to an independent variable. For the sake of simplicity, the orbital transfer problem under discussion will be studied under Keplerian dynamics with time as the independent variable, although other formulations may be used, yielding better results from the authors' experience, including the use of regularized coordinates. In the same fashion, the state vector is defined to be the position vector of the system expressed in some inertial frame in cylindrical coordinates, $\boldsymbol{s} = \left[\rho \quad \theta \quad z\right]^T$, while other parametrizations are obviously available.

The TBVP is given by

$$\begin{cases} \ddot{\rho} - \rho\dot{\theta} + \dfrac{\mu}{r^3}\rho = a_\rho \\ \rho\ddot{\theta} + 2\dot{\rho}\dot{\theta} = a_\theta \\ \ddot{z} + \dfrac{\mu}{r^3} z = a_z \end{cases}$$
$$\begin{cases} \boldsymbol{s}(0) = \boldsymbol{s}_0 & \dot{\boldsymbol{s}}(0) = \dot{\boldsymbol{s}}_0 \\ \boldsymbol{s}\left(t_f\right) = \boldsymbol{s}_f & \dot{\boldsymbol{s}}\left(t_f\right) = \dot{\boldsymbol{s}}_f \end{cases} \quad \textbf{(8)}$$

where μ is the gravitational parameter of the central body of interest, t_f is the final time of flight, $\boldsymbol{a}$ denotes the acceleration control vector resolved in cylindrical coordinates and $r = \sqrt{\rho^2 + z^2}$ is the radial distance to the origin of the inertial frame in which the motion is described.

In the following, the TBVP just presented will be treated in normalized coordinates, given by characteristic distances r_* and characteristic time t_*, such that $\mu_* = 1$. Moreover, the evolution of the system is parametrized by the normalized

DOI: 10.13009/EUCASS2022-4923

time variable $\tau = t/t_f$, therefore providing an homogenous treatment of both time-free and time-fixed problems. The TBVP is transformed into

$$\begin{cases} \ddot{\rho} - \rho\dot{\theta} + t_f^2 \dfrac{\mu}{r^3}\rho = u_\rho \\ \rho\ddot{\theta} + 2\dot{\rho}\dot{\theta} = u_\theta \\ \ddot{z} + t_f^2 \dfrac{\mu}{r^3} z = u_z \end{cases} \qquad \textbf{(9)}$$
$$\begin{cases} \boldsymbol{s}(0) = \boldsymbol{s}_0 & \dot{\boldsymbol{s}}(0) = t_f \dot{\boldsymbol{s}}_0 \\ \boldsymbol{s}(1) = \boldsymbol{s}_f & \dot{\boldsymbol{s}}(1) = t_f \dot{\boldsymbol{s}}_f \end{cases}$$

Derivatives denoted by $(\dot{\blacksquare})$ refer to the normalized time variable from now on, and we choose $\boldsymbol{u}$ as the normalized acceleration.

Finally, the state vector components are projected onto some orthogonal or non-orthogonal polynomial family, as given in Equation (2).

For constrained low-thrust orbital transfers, the continuous Bolza optimization problem to be finally solved is therefore

$$\begin{aligned} &\arg\min_x \; J = g\left[t_f, \boldsymbol{u}(\tau)\right] + \int_0^1 f\left[\boldsymbol{s}(\tau), \boldsymbol{u}(\tau), t\right] d\tau \\ &s.\,t \qquad \ddot{\rho} - \rho\dot{\theta} + t_f^2 \frac{\mu}{r^3}\rho = u_\rho \\ &\qquad\quad\;\; \rho\ddot{\theta} + 2\dot{\rho}\dot{\theta} = u_\theta \\ &\qquad\quad\;\; \ddot{z} + t_f^2 \frac{\mu}{r^3} z = u_z \qquad \textbf{(10)} \\ &s_i = \sum_{k=0}^{N_i} c_k P_k(\tau) \\ &\begin{cases} \boldsymbol{s}(0) = \boldsymbol{s}_0 & \dot{\boldsymbol{s}}(0) = t_f \dot{\boldsymbol{s}}_0 \\ \boldsymbol{s}(1) = \boldsymbol{s}_f & \dot{\boldsymbol{s}}(1) = t_f \dot{\boldsymbol{s}}_f \end{cases} \\ &t_f^2 \boldsymbol{a}_{\min} < \boldsymbol{u} < t_f^2 \boldsymbol{a}_{\min} \end{aligned}$$

where $\boldsymbol{a}_{min}$ and $\boldsymbol{a}_{max}$ are appropriate control bounds. These bounds are logically case-dependent and while a_{min} is often equal to 0, a_{max} will depend on mission details. The optimization variable vector $\boldsymbol{x} = \begin{bmatrix} \boldsymbol{c} & t_f \end{bmatrix}^T$ is composed of the polynomial coordinates for each state vector component and possibly the final time of flight in time-free problems. For minimum-time problems, the cost function is $J = t_f$ while for minimum-propellant missions it can be defined as the discrete integral $J = \int_0^1 |\boldsymbol{u}| \, d\tau$.

Once solved, the optimal solution is then mapped back to the time and dimensional domain, including the final cost function

DOI: 10.13009/EUCASS2022-4923

$$
\begin{aligned}
&\boldsymbol{a} = a_* \boldsymbol{u} / t_f^2 \\
&\rho(\tau) = r_* \rho(\tau) \\
&z(\tau) = r_* z(\tau) \\
&\Delta V = v_* t_f^{-1} \int_0^1 |\boldsymbol{u}| \, d\tau \\
&t_f = t_* t_f
\end{aligned}
\tag{11}
$$

Under this formulation, the low-thrust transfer problem can be then solved by means of the shape-based algorithm described in Section 2.

In the case of low-thrust transfers, one of the critical points on its resolution is the generation of an adequate initial guess. While direct transfers are a natural initial guess choice, after some benchmarking on the problem they have proven to be non-very robust compared to the choice of a multi-revolution trajectory. This last option allows to set the number of revolutions introduced as an extra optimization variable (it may also be fixed for mission design purposes) and, as a result, they increase the parametric optimization space. Consequently, we will use multi-revolution for this study.

The initialization process starts by first estimating the direct transfer time of flight $t_{f,0}$ for time-free problems through the Vis-Viva Theorem [12]

$$
t_{f,0} = \frac{v_f^2 - v_0^2 - 2\mu\left(r_f^{-1} - r_0^{-1}\right)}{a_{\max}\left(\dfrac{2\mu}{r_0 + r_f}\right)^{1/2}}
\tag{12}
$$

where r_0 and r_f refers to the boundary initial and final orbital radius and v_i is the associated orbital velocity. This initial estimation is corrected by a preliminary number of revolutions N_0

$$
\begin{aligned}
&N_0 = \left\lceil \frac{\theta_f - \theta_0 + \frac{1}{2} t_{f,0}\left(\dot{\theta}_0 + \dot{\theta}_f\right)}{2\pi} \right\rceil \\
&t_{f,0} = N_0 t_{f,0}
\end{aligned}
\tag{13}
$$

To impose the boundary conditions, the original in-plane angle θ_f and first-time derivative of the initial and final state vector shall be corrected at each iteration of the optimization method, including the initial guess, by

$$
\begin{aligned}
&\theta_{f,j} = \theta_f + 2\pi N_j \\
&\dot{\boldsymbol{s}}_j(0) = t_{f,j} \dot{\boldsymbol{s}}_0
\end{aligned}
\tag{14}
$$

where j indicates the iteration index.

Consequently, the initial guess of the trajectory is computed by solving the boundary condition linear system using an a-priori selected third-order polynomial expansion. At this point, different families behave differently: while Bernstein polynomials provide an analytical, closed expression for such initial guess [7,8], other families will require to numerically solve the system. Nevertheless, it is possible to perform a transformation between bases, using a simpler base for the initial trajectory then transferring it to the family for the final optimization process. This option presents

DOI: 10.13009/EUCASS2022-4923

another advantage: it is usually the case that the initial trajectory is over-sampled in a densified time mesh, while the final evaluation is planned for a sparser set of sampling points; in this situation, a least-squares solution can be employed to retrieve the initial guesses for the polynomial expansion from the over-sampled trajectory.

4. Testbench missions, applications, and studies

The following Section introduces several low-thrust orbital transfers examples to exemplify the capabilities of the methodology presented. Moreover, a metaheuristic technique to construct the algorithm is also introduced, revealing its effectiveness in finding global maximum solutions.

4.1 Low-thrust transfer examples

The shape-based methodology will be now applied to compute the optimal, minimum-propellant, time-free transfer between two LEO orbits in an inertial, geocentric reference system. The initial and final orbital elements are assumed to be known, including the initial and final anomaly. If this was not the case, the latter can always be selected as an optimization variable of the problem. Table 1 summarizes the initial and final classical orbital elements (where the true anomaly has been used) of the transfer in normalized coordinates.

Table 1: Initial and final orbital elements of the transfer

	$\boldsymbol{a}$	$\boldsymbol{e}$	$\boldsymbol{i}$ **[deg]**	**RAAN [deg]**	$\boldsymbol{\omega}$ **[deg]**	$\boldsymbol{\theta}$ **[deg]**
Departure orbit	1	0.001	0	0	0	95
Arrival orbit	1.05	0.005	1	15	10	270

Several algorithms were constructed based on different polynomial families and sampling grids for comparative purposes. In all cases, the sampling grid contained 60 nodes. Additionally, the polynomial expansion was restricted to the 9-th order.

The first reported case, considered as the baseline solution for the comparison, was computed using a 60-th order expansion and 500 nodes. Maximum thrust available was restricted to 0.05 mm/s^2, which corresponds to the characteristic thrust level of current electric propulsion systems.

Table 2 summarizes the different results of the methodology applied for the described low-thrust problem, including all the algorithms considered and the properties of the solution. Figures 1, 2 and 3 refer to the optimal trajectories found and the associated acceleration profiles. The time taken to solve the problem in each case was averaged on 25 iterations. Simulations have been performed using an 11th Gen Intel(R) Core(TM) i7-1165G7 @ 2.80GHz with 15.7 GB of RAM.

All the studied families and their variations converge to an optimal solution, as they all satisfy first-order optimality conditions. Nevertheless, different results are achieved depending on the exact construction of the shape-based method, depending on the sampling grid and the selected family. This is the expected conclusion, as the different combinations of sampling grid and polynomial basis provide the algorithm with different numerical properties and capabilities of reproducing the natural dynamics of the problem.

Moreover, the evaluation of some polynomial families is obviously more computationally expensive than others: including the resolution time result in a trade-off between relative error and solving time. This can be appreciated when comparing the baseline solution, clearly computationally expensive, with the rest of the studied families: For 2 orders of magnitude in solving time, 3σ relative errors are of 30% for the estimation of the ΔV and 3% for the time of flight.

The observed difference in error between ΔV and time of flight is also a natural result: while the computation of the first directly depends on the evaluation of a numerical integration scheme over the finite grid and it is therefore directly affected by it, the computation of the final time of flight is achieved by imposing the dynamics of the problem, which are grid independent.

DOI: 10.13009/EUCASS2022-4923

Table 2: Results of the different algorithms on the proposed mission scenario

Polynomial basis	**Sampling Grid**	**Optimal total ΔV [m/s]**	**Time of Flight [days]**	**Solving time [s]**	**Function evaluations**	**Iterations**
Bernstein	Sundman	897.40	603.64	106.0246	39033	201
Bernstein	Linear	1011.44	587.90	0.6613	2816	80
Bernstein	Random	935.65	594.57	0.5879	2067	58
Orthogonal Bernstein	Linear	1010.64	588.78	1.1245	3104	79
Orthogonal Bernstein	Random	1009.13	613.45	1.2692	3389	88
Orthogonal Bernstein	Normal	1010.96	861.06	1.1522	3321	88
Chebyshev	Chebyshev	1010.22	587.88	1.0511	3681	100
Chebyshev	Sundman	999.70	599.72	0.5342	3002	83
Legendre	Legendre	1010.26	587.86	1.0188	3463	95
Laguerre	Legendre	1136.49	589.27	1.0981	4476	124
Hermite	Legendre	1065.70	588.96	0.9498	3154	82

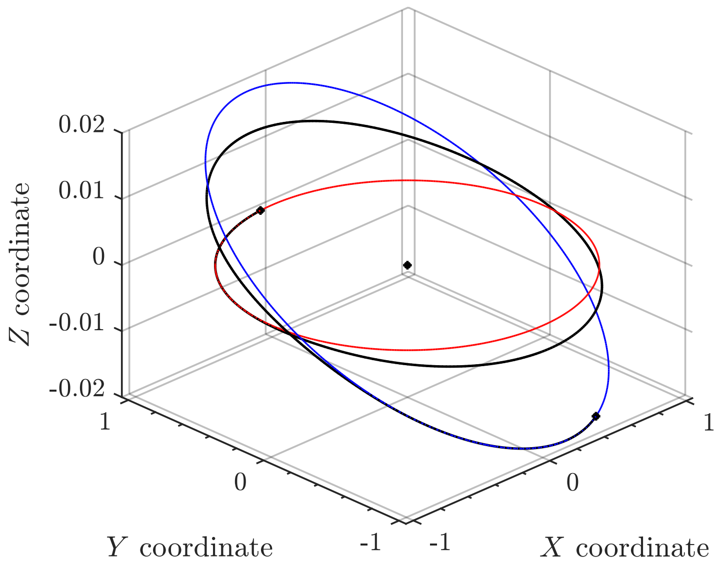

Figure 1: Baseline low-thrust trajectory for the Bernstein-Sundman case

DOI: 10.13009/EUCASS2022-4923

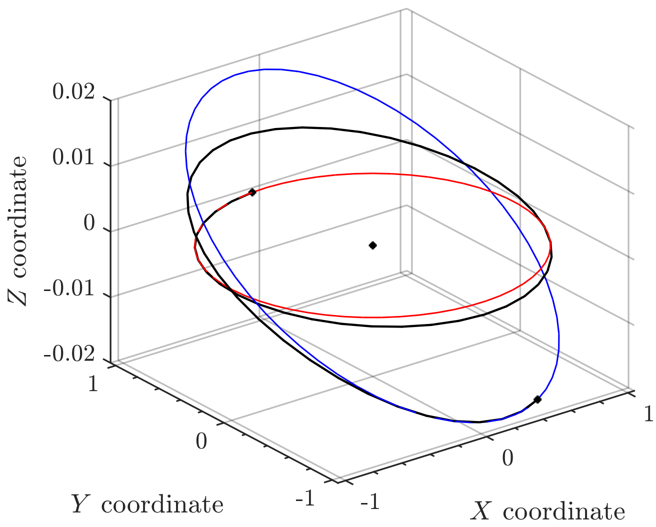

Figure 2: Optimal low-thrust trajectory for the Chebyshev-Sundman case

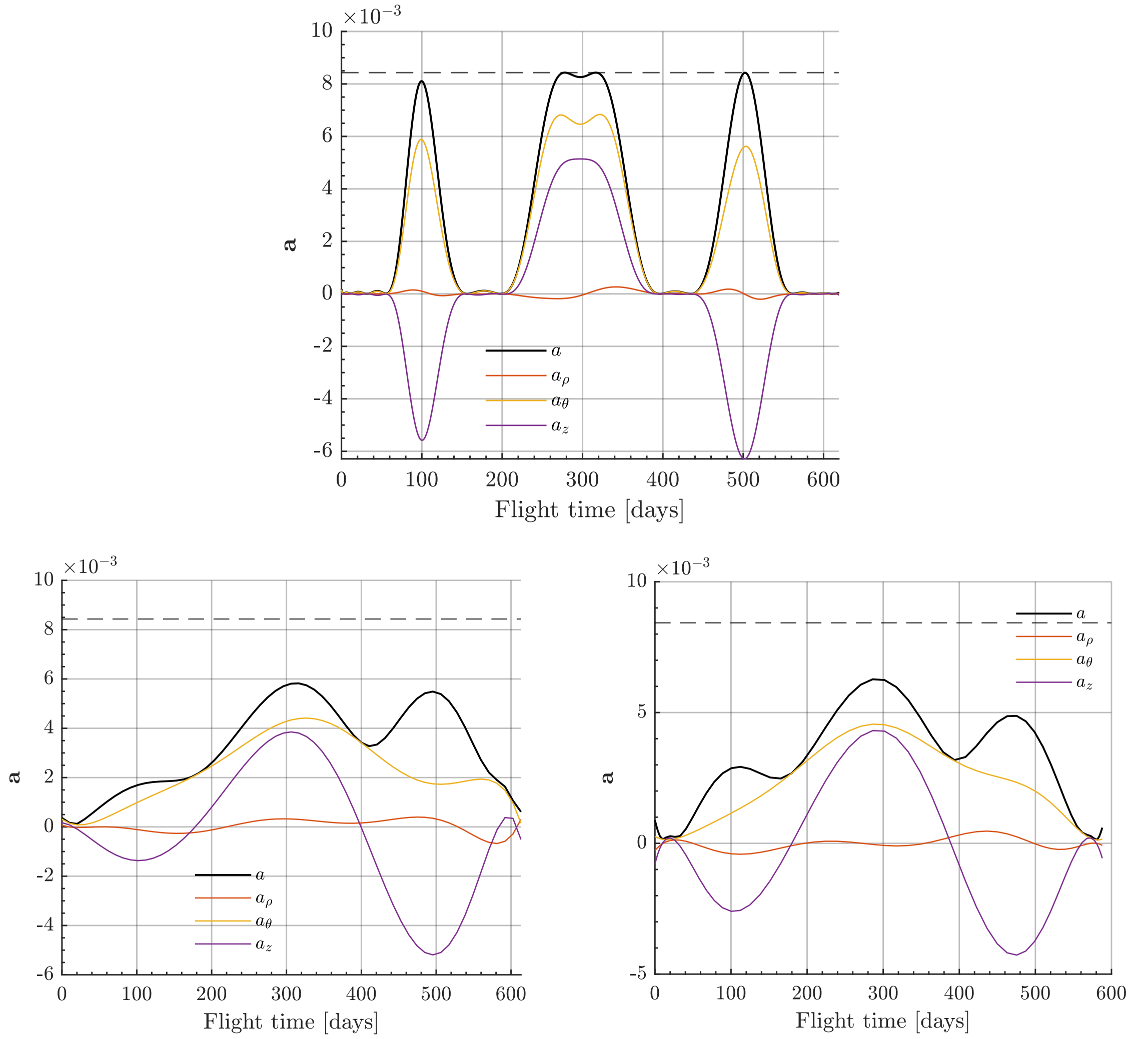

Figure 3: Constrained acceleration profile for the baseline, Chebyshev-Sundman and Chebyshev-Chebyshev cases

DOI: 10.13009/EUCASS2022-4923

4.2 Metaheuristic construction of the algorithm

As it can be extracted from the presented results (see Table 2), although with small variations, both the selected polynomial basis and topology of the final optimal sampling grid affect the numerical solution. At the same time, it can also be concluded that all the families reach an optimal solution and that the fundamental solution features remain invariant for all cases.

These differences may be traced back to the intrinsic numerical capabilities of the algorithms as function of its construction: the number of nodes, their distribution, the function polynomial basis and possibly the degree of expansion of the state vector evolution.

A genetic algorithm (in particular, Matlab's implementation of NSGA-II [16]) is now proposed to metaheuristically construct the shape-based method to yield a global maximum solution. The performance index to be minimized is multi-objective and depends on the specific application in which the shape-based approach will be used: if the optimal solution found may be refined by more complex optimization solver, the solution time may be the figure to be minimized, instead, for example, of the final ΔV.

In addition, some sampling grid topologies-polynomial basis combinations are not possible, given the domain of definition of the latter. All in all, this construction can be posed as mixed-integer NLP problem, for which genetic algorithms are a standard solver.

The previous mission scenario is now solved using this technique, in which the search space $\mathfrak{I}$ is expanded by: i) the polynomial basis; ii) the number of sampling nodes; iii) the distribution of such nodes; iv) the polynomial expansion. For this case, the cost function to be minimized by the solution $\varepsilon \in \mathfrak{I}$ is the combination of both the final and the time of flight in a multi-dimensional index

$$\arg\min_{\varepsilon \in \mathfrak{I}} J = \left[\Delta V/\sigma \quad t_f/\sigma\right]^T \tag{15}$$

where σ is a Boolean parameter coding the convergence results of the algorithm. A more complex function balancing the number of function evaluations, or the solving time is also suggested. The sparse Pareto front results, obtained after seconds, are found in Table 3, whereas Figures 4, 5, 6 and 7 show the minimum ΔV and t_f optimal low-thrust trajectories and their associated acceleration profiles.

Table 3: Metaheuristic Pareto front for the proposed mission scenario

Polynomial basis	**Sampling Grid**	**Number of nodes**	**Polynomial order**	**Optimal total ΔV [m/s]**	**Time of Flight [days]**	**Solving time [s]**	**Function evaluations**	**Iterations**
Legendre	Chebyshev	46	10	995.09	586.07	1.1122	4672	118
Legendre	Linear	47	10	1026.09	585.48	0.7752	3307	85
Bernstein	Regularized	60	10	971.35	599.11	1.1349	3558	91
Chebyshev	Legendre	33	10	994.18	586.29	1.1042	4870	121
Chebyshev	Linear	33	10	1024.95	585.53	0.5425	2541	62
Bernstein	Regularized	35	10	972.13	599.05	1.0882	4166	107
Legendre	Chebyshev	42	10	994.77	586.16	1.0938	4740	120

DOI: 10.13009/EUCASS2022-4923

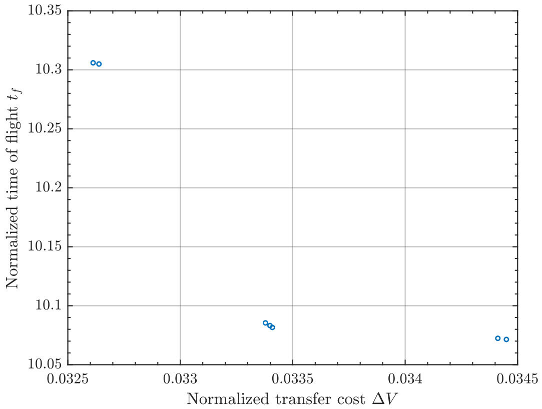

Figure 4: Projected sparse Pareto front solution

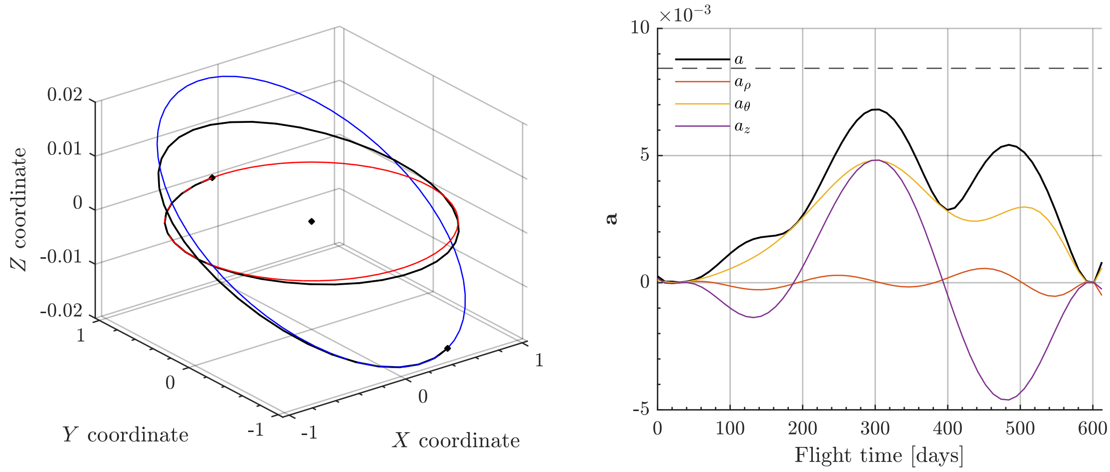

Figure 5: Global Maximum ΔV low-thrust trajectory (left) and associated acceleration profile (right)

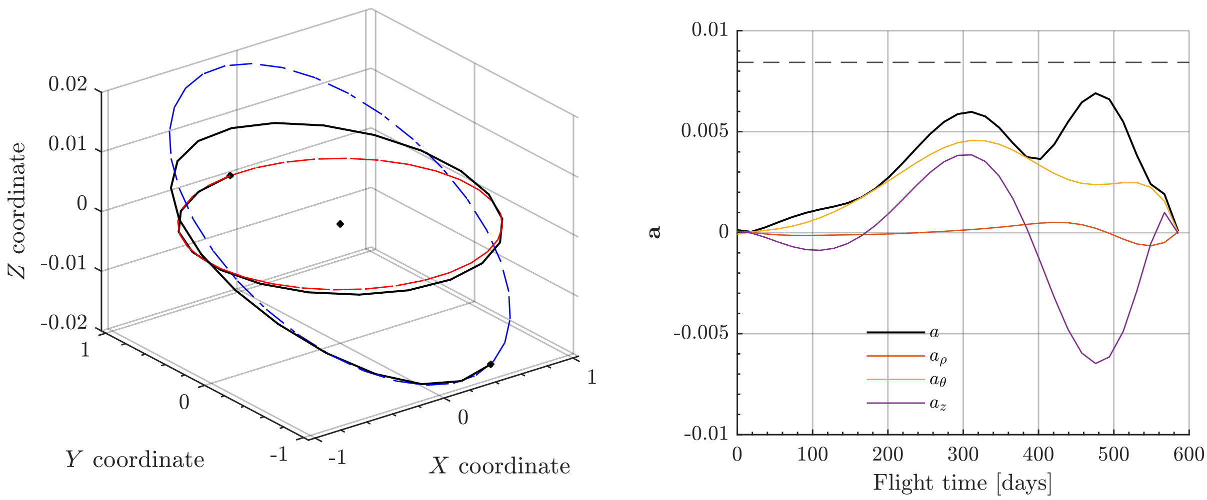

Figure 6: Global Maximum t_f low-thrust trajectory (left) and associated acceleration profile (right)

The first conclusion that can be extracted from the obtained Pareto front is that orthogonal bases are highly promoted under the elitist scheme, showing up in 75% of the Pareto front. Moreover, a regularized independent variable grid provides the cheapest trajectories, also noted from the initial comparison given in Table 2.

Regarding the order of the polynomial expansion, all Pareto front solutions show the maximum allowed, which also could be induced from the previous study but not verified. This tendency was preserved when going further than the order presented in Table 3 up to orders of the baseline solution.

The solutions' number of nodes present an almost normally distribution. All combinations show similar numerical performance in terms of function evaluations and iterations.

Another interesting consequence is that sparse sampling grids appear frequently as a solution of the Pareto front. This can be related to greater errors in the computation of the cost function, leading to a coarser, spurious solutions. On the other hand, when compared to the original results in Table 2, which were computed using a doubly dense optimization grid, no great difference is observed. Such result seems to point at a saturation tendency of the final optimal solution with the number of nodes: after exceeding a certain threshold, their influence on the final results is minimized.

This suggest the relative lack of importance of the number of nodes in the grid when solving the problem, whose solution is mainly driven by the exact geometrical distribution of the grid or equivalently, its density.

Finally, the use of Bernstein polynomials provides the fastest resolution time, which is a consequence on how boundary conditions are handled: in this case, the Bernstein family allows for analytical expressions instead of a numerical computation for Equations 4.

5. Conclusions

This work presents an extended formulation of classical shape-based methods for general Two-Boundary Value Bolza problems, with applications within low-thrust trajectory optimization and design. Such methodology enables a low-cost NLP optimization to replace more computation-hard classical direct transcription methods. New functional bases on which to express the system's motion are presented together with their performance comparison against classical orthogonal polynomials. Moreover, natural nodes of the selected natural bases are employed as the independent variable sampling grid on which to solve the problem, to profit from the numerical enhancements this approach brings when combined with the functional bases state vector projection. The presented methodology is finally particularised and discussed within orbital low-thrust transfers.

Several benchmark missions are presented and solved using the aforementioned techniques to demonstrate their capabilities in real mission scenarios, demonstrating the capabilities of the method under different combinations of parameters. Moreover, genetic algorithms are employed to construct optimal metaheuristic formulations of the proposed methodology to find supreme, Pareto front solutions to the presented missions and assess the effect of the formulation on the optimal result. The nature of such dependency and the exact agents affecting the method are still to be unveiled. A preliminary study on the effect of boundary conditions on the convergence rate of the algorithm is at the very moment being conducted.

Despite the general accuracy of the obtained results and the general demonstrated numerical properties of the method, in terms of efficiency and simplicity, several lines of research are still open and remain as interesting follow-up works for the presented research. First, the formal study of the exact evolution of the numerical error both in the computation of the cost function and the optimal trajectory approximation is still to be conducted. Moreover, the algorithm has been devised to be applicable to additional insightful test cases, including optimal landing trajectories on asteroids, attitude path planning and Initial Orbit Determination, which will further reveal the benefits of the proposed methodology.

Acknowledgments

This research was supported by Project Grant PID2020-112967GB-C33 by the Spanish State Research Agency.

All Authors wish to acknowledge funding from grant PID2020-112576GB-C22 of the Spanish State Research Agency and the European Regional Development Fund.

DOI: 10.13009/EUCASS2022-4923

References

[1] A. E. Petropulos and J.M. Longuski, Shape-Based algorithm for automated design of low-thrust, gravity-assist trajectories, Journal of Spacecraft and Rockets, 41 (2004), 787-796.
[2] D. Izzo, Lambert's problem for exponential sinusoids, Journal of Guidance, Control, and Dynamics, 29 (2006), 1242-1245.
[3] B. Wall and B. Conway, Shape-based approach to low-thrust rendezvous trajectory design, Journal og Guidance, Control, and Dynamics, 32 (2009), 95-102.
[4] C. Xie, G. Zhang and Y. Zhang, Shaping approximation for low-thrust trajectories with large out-of-plane motion, Journal of Guidance, Control, and Dynamics, 39 (2016), 2276-2786.
[5] J. Roa, J. Peláez and J. Senent, New analytic solutions with continous thrust : generalizez logarithmic spirals, Journal of Guidance, Control, and Dynamics, 36 (2016), 2336-2351.
[6] E. Taheri, Rapid space trajectory generation usign Fourier series shape-based approach, 2014.
[7] Z. Fan, M. Huo, N. Qi, C. Zhao, Z. Yu and T. Lin, Initial design of low-thrust trajectories based on Bézier curve-based shaping approach, Journal of Aerospace Engineering, 234 (2020).
[8] M. Huo, G. Mengali, A. Quarta et al., Electric sail trajectory design with Bézier curve-based shaping, Aerospace Science and Technology, 88 (2019), 126-135.
[9] D. Morante, M. Sanjurjo and M. Soler, Multi-objective low-thrust interplanetary trajectory optimization based on generalized logarithmic spirals, Journal of Guidance, Control, and Dynamics, 42 (2018), 1-15.
[10] Morante, David. (2020). Hybrid Multi-Objective Trajectory Optimization of Low-Thrust Space Mission Design. 10.13140/RG.2.2.24942.97603.
[11] Bellucci, Michael. On the explicit representation of orthonormal Bernstein polynomials. 2014.
[12] Taheri, E., Junkins, J.L. How Many Impulses Redux. J Astronaut Sci 67, 257–334 (2020). https://doi.org/10.1007/s40295-019-00203-1.
[13] B. Conway, A survey of method available for the numerical optimization of continuous dynamic systems, Journal of Optimization Theory and Applications, 152 (2012), 271-306.
[14] *fmincon*. MathWorks. https://es.mathworks.com/help/optim/ug/fmincon.html
[15] Numerical Optimization: Theoretical and Practical Aspects. J. Frédéric Bonnans, J. Charles Gilbert, Claude Lemaréchal, Claudia A. Sagastizábal. SpringerLink. 2006.
[16] *gamultiobj*. MathWorks. https://es.mathworks.com/help/gads/gamultiobj.html