Politecnico di Torino

ScuDo
Scuola di Dottorato ~ Doctoral School
WHAT YOU ARE, TAKES YOU FAR

Doctoral Dissertation
Doctoral Program in Aerospace Engineering (35^{th}cycle)

Performance Analysis of Direct and Indirect Formulations of the Legendre Pseudospectral Method for the Optimization of Spacecraft Trajectories

By

Alessia De Iuliis

******

Supervisor(s):
Prof. L. Casalino

Doctoral Examination Committee:
Prof. Michèle Lavagna, Referee, Politecnico di Milano.
Prof. Alessandro Zavoli, Referee, Università degli studi di Roma 'La Sapienza'
Prof. Dario Giuseppe Pastrone, Politecnico di Torino
Prof. Sandra Pieraccini, Politecnico di Torino
Dr. Marco Sagliano, German Aerospace Center (DLR)

Politecnico di Torino
2023

Declaration

I hereby declare that, the contents and organization of this dissertation constitute my own original work and does not compromise in any way the rights of third parties, including those relating to the security of personal data.

Alessia De Iuliis
2023

* This dissertation is presented in partial fulfillment of the requirements for **Ph.D. degree** in the Graduate School of Politecnico di Torino (ScuDo).

I would like to dedicate this thesis to Gloria, a newly bloomed flower

Acknowledgements

And I would like to acknowledge my supervisor Professor Lorenzo Casalino, whose unwavering support, insightful guidance, and encouragement have been invaluable throughout this journey. His expertise, dedication, and passion for research have been a constant source of inspiration.

Abstract

Pseudospectral methods have gained significant popularity as a state and control parameterization technique, finding extensive use in various optimal problems due to improved CPU performance and their versatile nature. Depending on the global polynomials used for the approximation of the controls and state variables, different pseudospectral methods are presented in literature. This work focuses on Legendre Pseudospectral methods, where Lagrange polynomials, obtained from orthogonal Legendre polynomials, are employed to globally interpolate state and control variables; the collocation can then be applied using Legendre-Gauss-Lobatto, Legendre-Gauss-Radau or Legendre-Gauss points.

This thesis aims at assessing the accuracy, efficiency, and applicability of pseudospectral methods to orbital transfers and formation reconfiguration problems while using a limited number of discretization points. This feature is essential to reduce the computational effort and enhance the practical implementation of these methods.

In this thesis, both direct and indirect formulations of the pseudospectral method are considered. The direct formulation employs Legendre-Gauss-Lobatto or Legendre-Gauss-Radau points for collocation, while the indirect Legendre-Gauss-Lobatto pseudospectral method is developed to cope with the difficulty of the direct method to approximate a non-smooth function with a finite series of smooth functions and it implements a smoothing technique to improve its convergence rate.

The pseudospectral methods yield discrete-time values for the state and control variables that fulfill the discretized constraints, producing a discrete-time feasible solution. The Bellman method is employed here to validate the practical application of the solutions to continuous-time dynamics and verify optimality and accuracy. The Bellman method allows for the optimization problem to be solved recursively, moving the initial conditions toward the final ones. The trajectory is partitioned into discrete segments, and the states within each segment are integrated using discrete control interpolations. It is analyzed how the quantity of segments affects the

ultimate states reached after the final integration and whether they meet the final boundary conditions specified by the problems.
Contributions of this thesis include analysis and numerical comparison of different direct and indirect formulations of Legendre pseudospectral method. For all the approaches it has been studied the effect of the number of nodes on the control law reconstruction, highlighting their positive and negative aspects. Furthermore, the implementation of Bellman algorithm enables the conversion of discrete solutions into continuous-time solutions, thereby verifying the applicability and practicality of the discrete control law in real scenarios. These methods prove to be a viable approach for efficiently and accurately solving orbital transfer and formation reconfiguration problems with low-thrust.

Contents

List of Figures **x**

List of Tables **xv**

Nomenclature **xvii**

1 Introduction **1**

1.1 Motivation for research 4

1.1.1 Analysis of the influence of the number of nodes on Pseudospectral method results 5

1.1.2 Use of Pseudospectral Method in an indirect optimization method 6

1.1.3 Analysis of the utilization of Pseudospectral methods within the framework of the Bellman method 7

1.1.4 Optimization of Spacecraft Trajectories 8

1.2 Thesis Summary & Contribution 9

2 Mathematical Background **11**

2.1 Optimal Control Theory 11

2.1.1 Continuous Bolza Problem 11

2.1.2 Indirect Transcription 13

2.1.3 Direct Transcription 15

2.2 Numerical Approximation Methods 18
2.2.1 Global Polynomial Approximations 18
2.2.2 Gaussian Quadrature approximation 21
2.2.3 Orthogonal Collocation 23

3 Methodology **24**
3.1 The Legendre Pseudospectral Method 24
3.1.1 Covector Mapping Theorem 29
3.2 The Indirect Legendre Pseudospectral Method 33
3.2.1 Continuation Smoothing Approach 34
3.3 Pseudospectral Methods Comparison 39
3.3.1 Radau Pseudospectral Method 40
3.3.2 Gauss Pseudospectral Method 41
3.3.3 Single State Problem 43
3.3.4 Orbit Transfer Problem 46
3.4 Bellman Pseudospectral Method 49

4 Results **53**
4.1 Orbital Transfer 54
4.1.1 Problem Formulation 54
4.1.2 Influence of the number of nodes 58
4.1.3 Comparative analysis between Pseudospectral Methods and Trapezoidal Collocation Method 73
4.1.4 Bellman Pseudospectral Method 76
4.1.5 3D Case 84
4.2 Formation Reconfiguration 92
4.2.1 PS Method 95
4.2.2 Ind.PS Method 99

4.2.3 Trapezoidal collocational method 101
4.2.4 Bellman Algorithm 102

5 Conclusion **113**

References **119**

List of Figures

2.1 Lagrange polynomial approximation of the function $f(x) = 1/(1 + 25x^2)$ using 25 equidistant points 19

2.2 Comparison of approximation accuracy for function $f(x) = e^t$ using uniform and non-uniform spacing points in function of the number of support points considered 20

2.3 Comparison of position of 10 points using equidistant, Legendre-Gauss, Legendre-Gauss-Radau and Legendre-Gauss-Lobatto collocation points 22

3.1 Visual representation of the covector mapping principle, showcasing the unification of direct and indirect methods in optimal control, as initially proposed by Ross and Fahroo 33

3.2 Optimal controls (u_ε) and added costs (εF) 37

3.3 Hyperbolic tangent control 38

3.4 Optimal solution for single-state problem 44

3.5 Errors on state, control and costate for the single-state problem using different Pseudospectral methods 45

3.6 States, control and costates for the Moyer and Pinkham problem using LG Pseudospectral method 48

3.7 States, control and costates for the Moyer and Pinkham problem using LGR Pseudospectral method 48

3.8 States, control and costates for the Moyer and Pinkham problem using LGL Pseudospectral method 48

3.9 States and costates for the Moyer and Pinkham problem using Ind.PS method 49

3.10 Bellman Pseudospectral Method flowchart 52

4.1 Trajectory, thrust, switching function, and in-plane thrust angle for Case 1 solved with the indirect method 59

4.2 Error on the final mass with different number of nodes for Case 1 considering LGL and LGR methods 59

4.3 Thrust and switching function reconstruction for Case 1 using 10 nodes, considering LGL and LGR method 60

4.4 Thrust and switching function reconstruction for Case 1 using 30 nodes, considering LGL and LGR method 60

4.5 Thrust and switching function reconstruction for Case 1 using 50 nodes, considering LGL and LGR method 61

4.6 Error on the final mass with different number of nodes for Case 1 considering Ind.PS method 62

4.7 Thrust and switching function reconstruction for Case 1 using 40 and 50 nodes, considering Ind.PS method 63

4.8 Trajectory, thrust, switching function, and in-plane thrust angle for Case 2 solved with the indirect method 64

4.9 Error on the final mass with different number of nodes for Case 2 considering LGL and LGR method 65

4.10 Thrust and switching function reconstruction for Case 2 using 20 nodes, considering LGL and LGR method 66

4.11 Thrust and switching function reconstruction for Case 2 using 60 nodes, considering LGL and LGR method 67

4.12 Thrust and switching function reconstruction for Case 2 using 70 nodes, considering LGL and LGR method 68

4.13 Error on the final mass with different number of nodes for Case 2 and considering Ind.PS method 69

4.14 Thrust and switching function reconstruction for Case 1 using 25 and 60 nodes, considering Ind.PS method 69

4.15 Trajectory, thrust, switching function, and in-plane thrust angle for Case 3 solved with the indirect method 70

4.16 Error on the final mass with different number of nodes for Case 3 and considering LGL and LGR method 71

4.17 Thrust and switching function reconstruction for Case 3 using 25 nodes, considering LGL and LGR method 71

4.18 Thrust and switching function reconstruction for Case 3 using 50 nodes, considering LGL and LGR method 72

4.19 Thrust and switching function reconstruction for Case 3 using 70 nodes, considering LGL and LGR method 73

4.20 Thrust reconstruction for Case 2 using 60 nodes, considering TC method 75

4.21 Thrust reconstruction for Case 1 using 30 nodes and $N_B = 4$, considering Bellman algorithm with LGR, LGL and Ind.PS. The thrust was reconstructed interpolating the discrete control 77

4.22 Thrust reconstruction for Case 1 using 30 nodes and $N_B = 4$, considering Bellman algorithm with LGR and Ind.PS method. The thrust was reconstructed using sf 78

4.23 Periapsis and apoapsis evolution for Case 1 using 30 nodes and $N_B = 4$, considering Bellman algorithm with LGR and Ind.PS method. The thrust was reconstructed using sf 79

4.24 Thrust reconstruction for Case 2 using 40 nodes and $N_B = 8$, considering Bellman algorithm with LGL and LGR method. The thrust was reconstructed interpolating the discrete control. 82

4.25 Thrust reconstruction for Case 2 using 40 nodes and $N_B = 8$, considering Bellman algorithm with Ind.PS and LGR method. The thrust was reconstructed using sf. 82

4.26 Periapsis and apoapsis evolution for Case 2 using 40 nodes and $N_B = 4$, considering Bellman algorithm with LGR and Ind.PS method. The thrust was reconstructed using sf 84

4.27 Trajectory, thrust, switching function, and control angles for 3D Case solved with the indirect method 85

4.28 Thrust reconstruction for 3D Case using LGR, LGL, and Ind.PS method with 40 nodes 86

4.29 Semi-major axis, eccentricity and inclination for 3D Case using LGR, LGL, and Ind.PS method with 40 nodes 86

4.30 Thrust reconstruction for 3D Case using Bellman algorithm with LGR, LGL, and Ind.PS with 40 nodes and $N_B = 4$. The thrust was reconstructed interpolating the discrete control. 87

4.31 Periapsis, apoapsis, inclination and eccentricity evolution for the 3D Case using 40 nodes and $N_B = 4$, considering Bellman algorithm with LGL, LGR, and Ind.PS. The thrust was reconstructed interpolating the discrete control. 88

4.32 Thrust reconstruction for 3D Case using Bellman algorithm with LGR and Ind.PS with 40 nodes and $N_B = 4$. The thrust was reconstructed using sf. 90

4.33 Periapsis, apoapsis, inclination and eccentricity evolution for the 3D Case using 40 nodes and $N_B = 4$, considering Bellman algorithm with LGR and Ind.PS. The thrust was reconstructed using sf. 90

4.34 Acceleration and trajectory for Case$_{\mathrm{FR}}$ 1 solved with LGL PS method 96

4.35 Acceleration and trajectory for Case$_{\mathrm{FR}}$ 2 solved with LGL PS method 96

4.36 Acceleration and trajectory for Case$_{\mathrm{FR}}$ 1 solved with *IPOPT* LGL PS method 97

4.37 Acceleration and trajectory for Case$_{\mathrm{FR}}$ 2 solved with *IPOPT* LGL PS method 98

4.38 Acceleration and trajectory for Case$_{\text{FR}}$ 1 solved with LGR PS method 98

4.39 Acceleration and trajectory for Case$_{\text{FR}}$ 2 solved with LGR PS method 99

4.40 Acceleration, primer vector and trajectory for Case$_{\text{FR}}$ 1 solved with Ind.PS method 100

4.41 Acceleration, primer vector and trajectory for Case$_{\text{FR}}$ 2 solved with Ind.PS method 101

4.42 Acceleration and trajectory for Case$_{\text{FR}}$ 1 solved with TC method . 102

4.43 Acceleration and trajectory for Case$_{\text{FR}}$ 2 solved with TC method . 102

4.44 Acceleration and trajectory for Case$_{\text{FR}}$ 1 solved using Bellman+LGL PS method and $N_B = 2$. 103

4.45 Acceleration and trajectory for Case$_{\text{FR}}$ 2 solved using Bellman+LGL PS method and $N_B = 2$. 103

4.46 Acceleration and trajectory for Case$_{\text{FR}}$ 1 solved with Bellman+LGR PS method, using discrete control interpolation 104

4.47 Acceleration, primer vector and trajectory for Case$_{\text{FR}}$ 1 solved with Bellman+Ind.PS method, using discrete control interpolation 105

4.48 Acceleration and trajectory for Case$_{\text{FR}}$ 2 solved with Bellman+LGR PS method, using discrete control interpolation 107

4.49 Acceleration, primer vector and trajectory for Case$_{\text{FR}}$ 2 solved with Bellman+Ind.PS method, using discrete control interpolation 107

4.50 Acceleration and trajectory for Case$_{\text{FR}}$ 1 solved with Bellman+LGR PS method, using sf for control reconstruction 108

4.51 Acceleration, primer vector and trajectory for Case$_{\text{FR}}$ 1 solved with Bellman+Ind.PS method, using sf for control reconstruction 109

4.52 Acceleration and trajectory for Case$_{\text{FR}}$ 2 solved with Bellman+LGR PS method, using sf for control reconstruction 110

4.53 Acceleration, primer vector and trajectory for Case$_{\text{FR}}$ 2 solved with Bellman+Ind.PS method, using sf for control reconstruction 111

List of Tables

4.1 Computational time for Case 1 using LGL, LGR, Ind.PS, and trapezoidal collocational method. 74

4.2 Computational time for Case 2 using LGL, LGR, Ind.PS, and trapezoidal collocational method. 75

4.3 Errors for the final states for Case 1 using Bellman + PS with $N_B = 4$. The thrust was reconstructed interpolating the discrete control. . . . 78

4.4 Errors for the final states and final mass for Case 1 using Bellman + LGR and Ind.PS with $N_B = 4$. The thrust was reconstructed using sf. 80

4.5 Errors for the final states for Case 2 using Bellman + PS with $N_B = 4, 6, 8$. The thrust was reconstructed interpolating the discrete control. 81

4.6 Errors for the final states for Case 2 using Bellman + Ind.PS with $N_B = 4, 6, 8$. The thrust was reconstructed interpolating the discrete control. 81

4.7 Errors for the final states and final mass for Case 2 using Bellman + Ind.PS with $N_B = 4, 6, 8$. The thrust was reconstructed interpolating the discrete control. 83

4.8 Errors for the final states and final mass for the 3D Case using Bellman + PS/ Ind.PS with $N_B = 4$. Results are shown for the two techniques: thrust reconstructed interpolating the discrete control and thrust reconstructed using sf. 89

4.9 Initial and final ROEs (meters) for Case$_{\text{FR}}$ 1 and 2. 95

4.10 Errors on the final ROEs after the application of Bellman algorithm for Case$_{\text{FR}}$ 1, using discrete control interpolation. 106

4.11 Errors on the final ROEs after the application of Bellman algorithm for Case $_{\mathrm{FR}}$ 2, using discrete control interpolation. 106

4.12 Errors on the final ROEs after the application of Bellman algorithm for Case $_{\mathrm{FR}}$ 1, using sf for control reconstruction. 108

4.13 Errors on the final ROEs after the application of Bellman algorithm for Case $_{\mathrm{FR}}$ 2, using sf for control reconstruction. 110

4.14 Resulting ΔV after the application of Bellman algorithm, using discrete control interpolation. 112

4.15 Resulting ΔV after the application of Bellman algorithm, using sf. . 112

Nomenclature

Roman Symbols

$\mathbf{p}$	primer vector
a	semi-major axis
e	eccentricity
i	inclination
M	mean anomaly
m	mass
m_p	propellant mass
n	chief's mean motion
P	chief's orbital period
r	radial distance
R_E	Earth radius
t_p	time of periapsis passage
u	radial velocity
v	transversal velocity
w	normal velocity
T	Thrust

Greek Symbols

β	thrust magnitude
$\delta\Lambda$	relative mean longitude
δa	relative semi-major axis
δe	relative eccentricity vector
δi	relative inclination vector
ΔV	change in velocity
ε	continuation parameter
λ	costate
Ω	Right Ascension of the Ascending Node
ω	argument of periapsis
ϕ	latitude angle
ψ	out-of-plane thrust angle
σ	in-plane thrust angle
θ	longitude angle
ϑ	mean argument of latitude

Superscripts

x^*	optimal state

Subscripts

0	initial condition
c	chief
d	deputy
f	final condition

r_a apoapsis

r_p periapsis

x_B state obtained with Bellman Pseudospectral method

FR Formation Reconfiguration

T Target

Acronyms / Abbreviations

sf switching function

BVP Boundary Value Problem

CGL Chebyshev-Gauss-Lobatto

CMP Chebishev Pseudospectral Method

CMT Covector Mapping Theorem

ECI Earth Centered Inertial reference frame

GVE Gauss Variational Equations

HBVP Hamiltonian Boundary Value Problem

Ind.PS Legendre-Gauss-Lobatto Indirect Pseudospectral Method

JPM Jacobi Pseudospectral Method

KKT Karush-Kuhn-Tucker

LG Legendre-Gauss

LGL Legendre-Gauss-Lobatto

LGR Legendre-Gauss-Radau

LPM Legendre Pseudospectral Method

NLP Non-linear Programming

PMP Pontryagin's Maximum Principle

PS	Pseudospectral
ROE	Relative Orbital Elements
RTN	Radial-Tangential-Normal reference frame
TC	Trapezoidal Collocation

Chapter 1

Introduction

The birth of optimal control can be traced back to 1697 in Groningen, a university town in The Netherlands, when Johann Bernoulli, a professor of mathematics at the local university, published his solution to the brachystochrone problem. The year before he had challenged his contemporaries to solve this problem: *If in a vertical plane two points A and B are given, then it is required to specify the orbit AMB of the movable point M, along which it, starting from A, and under the influence of its own weight, arrives at B in the shortest possible time*[1].
Six of the best mathematicians of the time submitted solutions to the problem: Newton, Leibniz, de l'Hôpital, Tschirnhaus, and the Bernoulli brothers; paving the way to the optimal control theory. One of the earliest advancements of the optimal control was the creation of calculus of variations, developed by Euler in 1733 [2], further improved by Lagrange (1792) and, in the 19^{th} century, by Legendre, Jacobi, Hamilton, and Weierstrass. In the 1950s, Richard Bellman's pioneering work in dynamic programming [3] established sufficient conditions for optimality through the use of the Hamilton-Jacobi-Bellman equation. However, the conventional wisdom holds that optimal control theory was born in 1962 in the former Soviet Union, with the work on the "Pontryagin minimum principle" by L. S. Pontryagin and his group [4]. They enriched and broaden the calculus of variation, stating that a minimization path must satisfy the Euler-Lagrange equations where the optimal controls minimize the Hamiltonian. The widespread adoption of optimal control theory was then made possible by the commercial availability of digital computers in the 1950s [5]. In the late 1950s and early 1960s, researchers such as Lawden [6], Leitmann [7], Miele [8], and Breakwell [9] showcased the use of calculus of variations in optimizing

aerospace flight paths through shooting algorithms. Meanwhile, Kelley [10] and Bryson [11] improved the methodology by introducing gradient algorithms that addressed the inherent instability of shooting methods. Numerical methods for solving optimal control problems have seen substantial advancements in recent years. Today, there is an extensive selection of methods available with various levels of complexity, each utilizing different approaches to tackle the problem. The basic idea behind these methods is to transform the continuous-time problem into a discrete, finite-dimensional problem that can be solved using computational techniques. The solution is then obtained by refining the approximate solution until it meets a specified tolerance level. Optimal control problems can be solved using different computational methods. Most of these methods can be grouped into two major categories: indirect and direct methods [12]. Indirect methods rely on the necessary optimality condition, derived from the minimum principle [4], to obtain the optimal trajectory, that is the extremal trajectory with the lowest cost. They are rapid and precise and they lead to the solution of a Boundary Value Problem (BVP) . Moreover, their solution assures first-order optimality condition. However, these methods present small radius of convergence and they require the derivation of the optimality conditions and a good tentative solution of the costates, which is often non-intuitive and difficult to obtain. For problem with path constrains, it is also necessary to have *a priori* knowledge of the switching structure.

On the other hand, direct methods discretize the optimal control problem to a parameter optimization problem and then solve the resulting Non-linear Programming problem (NLP) [13] by well-established algorithms based on equations known as the Karush-Kuhn-Tucker (KKT). Direct methods have the advantage of eliminating the need to derive the first-order necessary conditions. They also have a much larger convergence radius compared to indirect methods, thus reducing the need for an accurate initial guess, and do not require an initial guess for the costate. Additionally, the switching structure does not need to be known beforehand. However, a disadvantage of many direct methods is that they may provide inaccurate costate or no costate information at all, making it uncertain whether the NLP solution is indeed the optimal solution to the original optimal control problem. A feature that differentiates the different direct methods is the choice of which quantities to discretize and how to approximate the continuous-time dynamics. The parametrization can be done on the control variable only or on both control and state variables [14]. Examples of control parametrization include shooting [15] and multi-shooting methods [16]. The former

parameterizes the control variables and uses them in the integration throughout the entire interval, resulting in the determination of the states from the initial values; while the latter, divides the path into shorter segments, it integrates starting from their initial values (which become additional parameters), and then applies conditions to ensure continuity between the intervals. On the other hand, when both the state and the control are discretized within the NLP and the continuous-time differential equations are converted into algebraic equations, the sensitivity issues of direct shooting methods are avoided at the cost of a larger NLP [13, 17, 18].

Pseudospectral (PS) methods [19, 20] have gained significant popularity as a type of state and control parameterization method and they are based on Spectral collocation methods, initially developed to solve fluid dynamics problems [21, 22, 20]. Pseudospectral uses a finite basis of global interpolating polynomials to approximate the state and control at a set of discretization points. The interpolating polynomial's derivative is used to estimate the time derivative of the state in the dynamic equations, which is then constrained to equal the vector field of the equations at a set of collocation points. Though any unique set of collocation points can be chosen, an orthogonal collocation—in which the collocation points are linear combinations of such polynomials and their derivatives—is generally opted. As a result, the terms Pseudospectral [20] and orthogonal [23] collocation are often used interchangeably. Nowadays, thanks to the improved performance of CPUs and PS versatility, they are extensively used. In fact, literature presents a vast number of diverse optimal control problems that have been solved using Pseudospectral methods, such as space station attitude control [24, 25], ascent guidance [26, 27], interplanetary solar sail mission design [28, 29], low-thrust Earth-to-Jupiter rendezvous [30], multiple revolutions low-thrust Earth-orbit transfers [31], lunar guidance [32, 33], launch vehicle trajectory optimization [34, 35] and libration-point stationkeeping [36, 37]; just to mention some of the countless applications.

Depending on the global polynomials used for the approximation of the controls and state variables, different PS methods are presented in literature, such as Chebishev Pseudospectral (CPM) [38], Legendre Pseudospectral (LPM) [19, 39–41], and Jacobi Pseudospectral (JPM) [42]. To approximate state and control the CMP uses Chebyshev polynomials and it performs orthogonal collocation at the Chebyshev-Gauss-Lobatto (CGL) points. To improve its performance Fahroo and Ross introduced a modification to the method, that employs a Clenshaw-Curtis quadrature [43]. In Legendre Pseudospectral method, Lagrange polynomials obtained from

orthogonal Legendre polynomials are employed to globally interpolate state and control variables; Legendre-Gauss-Lobatto (LGL) [19, 44], Legendre-Gauss-Radau (LGR) [45–47], and Legendre-Gauss (LG) [48, 49] points can be used for orthogonal collocation. The Legendre polynomials are a subset of Jacobi polynomials, which are used to locate the collocation points in the more general Pseudospectral method known as the Jacobi Pseudospectral method [42]. On the other hand, the Hermite-LGL method [50] uses piecewise cubic polynomials rather than Lagrange polynomials, and collocates at a subset of the LGL points.
This thesis primarily investigates the effectiveness and efficiency of Legendre Pseudospectral methods, both in direct and indirect formulations, for numerical solutions of optimal control problems. Additionally, it aims to introduce improvements to enhance their performance.

1.1 Motivation for research

The following sections outline the motivations behind this research, which can be summarized as follows: our goal is to examine the performance of Pseudospectral methods and assess how the number of nodes affects the results. Specifically, using a lower number of nodes can significantly reduce computational time, making it a potentially suitable method for autonomous guidance concepts. Therefore, it is intriguing to analyze how few nodes can still yield satisfactory results. We also aim to explore the application of the indirect formulation of the Pseudospectral method in addressing the challenges encountered by the direct Pseudospectral method when solving bang-bang control problems. Additionally, we integrate both the Pseudospectral methods with the Bellman method to transition from a discrete-time to a continuous-time solution. By doing so, we will assess the practical feasibility of the solutions obtained. Ultimately, we will apply these methods to orbital transfer and formation reconfiguration problems to further our understanding of their effectiveness.

1.1.1 Analysis of the influence of the number of nodes on Pseudospectral method results

The motivation behind this analysis is driven by the aspiration to achieve quick solutions by utilizing a reduced number of nodes to address the optimal control problem, while not losing too much accuracy of the solution.
The utilization of traditional direct collocation methods for solving finite-thrust problems can lead to a significantly large NLP problem, which requires extensive computational resources to meet accuracy standards [51]. As a result, the development of large-scale NLP solvers, such as SNOPT [52] for example, was imperative for the effective implementation of this approach. It is noteworthy that the quest to solve optimal control problems has played a pivotal role in driving the growth and evolution of large-scale NLP methods. This has led to remarkable achievements such as those presented by Gill et al. [52], who were able to successfully solve more than a thousand example problems, each comprising up to 40,000 variables and constraints. Similarly, Betts and Erb [51] tackled a highly intricate trajectory optimization problem, which involved over 200,000 variables and around 150,000 constraints. Additionally, Ferris and Munson [53] demonstrated innovative techniques that enable the efficient resolution of quadratic programming problems with up to 10×10^6 variables. These exceptional achievements show the immense potential of large-scale NLP methods to address complex real-world problems. However, in light of the study done by Ross et al. [54, 55], an alternative approach was pursued in this thesis, with a focus on exploring the feasibility and computational efficiency of solving optimal control problems using a limited number of nodes, trading some optimality of the solution. Therefore, initially, the impact on the final cost function and the reconstruction of control variables were evaluated for bang-bang problems, while varying the number of nodes used in the analysis. This was done in order to comprehensively assess the effects of using a reduced number of nodes on the accuracy and quality of the solutions obtained, while still maintaining a reasonable level of computational efficiency.

1.1.2 Use of Pseudospectral Method in an indirect optimization method

In order to enhance the solutions provided by the direct Pseudospectral method for solving bang-bang control problems, an indirect Pseudospectral method has been developed. The direct method, while effective in certain cases, often produces control solutions that exhibit intermediate thrust values during the transition from a firing arc to a coast arc. The development of the indirect method aims to address this limitation and improve the accuracy of the obtained solutions.

The Pseudospectral method has been extensively used in the past years to solve numerous optimal control problems [56–58, 20, 59]. However, when the problem at hand is a switching optimal control problem, the method suffers the difficulty to approximate a non-smooth function by a finite series of smooth functions, the so-called Gibbs phenomenon [20]. To cope with the difficulty of reconstructing the discontinuities in the control functions, in the last decades, several modified Pseudospectral schemes were presented in literature. The PS Knotting Method [60] is one of the first of these methods found in literature. Ross and Fahroo introduce the concept of PS knot used to exchange information across time intervals in the form of event conditions. The method has a direct multi-phase approach and in each phase, it applies Legendre PS method. Based on this work, Gong et al. proposed an autonomous mesh refinement that automatically determines the size and the layout of the grid to achieve a desired estimation accuracy [61]. Darby et al. develop a hp-adaptive PS method in which, to obtain a desired accuracy, the number and width of intervals and the polynomial degree must be specified iteratively [62]. In his work Shamsi [63] presents a modified Pseudospectral procedure for obtaining bang-bang optimal solution using piecewise continuous approximation polynomials for states and piecewise constant functions for controls and a discretized integral form of the optimal control problem. The homotopic and PS method are merged together in Guo et al. [64]. Here, the smooth optimal control problem is first solved through the PS method. Then, using the costates estimated based on the results of the direct optimization, the homotopic procedure leads to the solution of the nonsmooth optimal control problem. Mehrpouya [65, 66] implements an indirect Legendre-Gauss-Radau PS method to approximate the solution of a class of switching optimal control problems. He converts the problem to the solution of a system of algebraic equations using as initial guess the solution of a simplified optimization problem.

This study aims to develop an indirect Legendre-Gauss-Lobatto (LGL) Pseudospectral method for approximating the solution of a specific class of switching optimal control problems. While Pseudospectral methods using a limited number of nodes are known to have difficulty in accurately reconstructing bang-bang thrust profiles due to the presence of intermediate thrust values, it was decided to address this issue with an indirect method. The problem is readily transformed into a system of algebraic equations, where the unknown parameters were the values of the functions at the LGL points. However, finding a suitable initial guess to solve the system of equations was challenging. To overcome this issue, a smoothing technique was implemented.

1.1.3 Analysis of the utilization of Pseudospectral methods within the framework of the Bellman method

After obtaining solutions using a limited number of discretization points with both the direct and indirect Pseudospectral methods, it becomes intriguing to assess their practical applicability from an engineering perspective. Specifically, we aim to evaluate the suitability of these results for integration purposes, as numerical propagator outcomes are often regarded as the benchmark or "truth" in engineering analysis.
The Bellman Pseudospectral method [55] is a numerical algorithm that is used for solving optimal control problems. The method combines the Bellman principle of optimality with the Pseudospectral approach to produce an efficient and accurate solution. The Bellman principle of optimality states that an optimal control problem can be broken down into a sequence of subproblems that are also optimal. This allows for the optimization problem to be solved recursively, moving the initial conditions toward the final ones. The PS methods generate discrete-time values for the state and control variables that fulfill the discretized constraints, producing a discrete-time feasible solution. While increasing the number of nodes can be used to address the discrepancy between discrete-time and continuous-time feasibility, it may not be the most efficient approach. Hence, to obtain a valid solution for the continuous-time optimal control problem, the Bellman method is employed here. This requires the discrete solution to be mapped back to the continuous-time domain. The trajectory is partitioned into discrete segments, and the states within each segment are integrated using discrete control interpolation. It is intriguing to

explore how the quantity of segments affects the ultimate states reached after the final integration and whether they meet the final boundary conditions specified by the problem.

1.1.4 Optimization of Spacecraft Trajectories

Orbital Transfer

Optimizing orbital transfers between two circular orbits is a problem that has received significant attention in the field of astrodynamics. The problem involves determining the most efficient and economical way to transfer a spacecraft from one circular orbit to another while minimizing fuel consumption and time. The oldest example of this problem is the two-impulse Hohmann transfer (1925) [67]. However, its proof of optimality came much later [68, 69]. To find the optimal solution to this class of problems, various numerical optimization techniques have been used. These techniques include gradient-based and gradient-free optimization methods, such as nonlinear programming, genetic algorithms [70], and particle swarm optimization [71]. These methods have been used to minimize the total transfer time or fuel consumption while satisfying the physical constraints of the problem. In this research Legendre PS method, using both LGL and LGR collocation points, and Indirect LGL PS methods have been used to solve orbital transfer problems with finite thrust and fixed final time. The problem wants to minimize the propellant consumption or alternatively, maximize the final mass. Discrete results are then converted to continuous-time solutions using the Bellman method and they are compared with optimal solutions obtained through an indirect method.

Formation Reconfiguration

In recent years, research showed an increasing interest in space flight mechanics and dynamics and control for spacecraft formation flying. These kinds of problems include formation reconfiguration, distributed space systems missions, and on-orbit serving and they require good establishment and reconfiguration of the relative motion of co-orbiting vehicles in a safe, accurate, and fuel-efficient manner. TanDEM-X [72], PRISMA [73, 74] and AVANTI [75] are just few of the several missions that have already proven the efficiency of formation flying missions. These examples

considered impulsive maneuvers for autonomous optimum formation reconfiguration over a given time interval. However, the development of all-electric spacecraft, rises also great interest in the optimization of formation flying with low-thrust electric propulsion systems, since they ensure a lower fuel consumption thanks to their high specific impulse. [76–78]. This study focuses on space trajectory optimization for finite-thrust formation reconfiguration of the relative motion between two spacecraft. One spacecraft, which defines the reference orbit, is designated the "chief" and it is uncontrolled. The other spacecraft is designated the "deputy" and it is controlled by a three-component thrust input. In this work, the goal is to evaluate the performance of two mathematical methods to quickly obtain optimal or near-optimal solutions. This is achieved using a low number of collocation points and applying the Bellman method to the problems.

1.2 Thesis Summary & Contribution

This section aims to provide an overview of the thesis chapters and highlights their significant contributions. The thesis deals with various mathematical concepts that are required to understand theoretical advancements with respect to the literature. Chapter 2 presents these concepts in a clear and simple manner and explains the reasoning behind using these mathematical constructs and their benefits.
The methodology section (chapter 3) first introduces the Legendre Pseudospectral method and its relative covector mapping theorem, which provides an order-preserving transformation of the Lagrange multipliers associated with the discretized problem to the discrete covectors associated with the optimal control problem. The indirect formulation of the Legendre Pseudospectral method is presented along with a continuation smoothing technique that improves its convergence rate. The Radau Pseudospectral method and Gauss Pseudospectral method are also briefly introduced to enable comparison among the three methods for a single-state problem and a more realistic problem such as an orbit-raising problem. The Bellman Pseudospectral method is also presented as a way to obtain a valid solution for the continuous-time optimal control problem.
In chapter 4, two case studies are presented, namely, orbital transfer and formation reconfiguration. For the orbital transfer problems, the influence of the number of nodes is analyzed and the limitation of using a low number of nodes becomes apparent

for some study cases. The Bellman Pseudospectral method is used to overcome this limitation. Furthermore, a 3D case with an inclination variation is also considered for completeness. As for the formation reconfiguration problem, the PS method, Indirect PS method, and Bellman method are compared for two study cases, one considering the variation of the relative semi-major axis and mean longitude and the other requiring a variation of the eccentricity vector.

Finally, chapter 5 summarizes the significant contributions of this thesis and suggests potential future research directions. The thesis investigates the effectiveness of Pseudospectral methods in solving optimal control problems using different collocation points and direct and indirect formulations. Moreover, it provides insights into the advantages and limitations of different approaches.

Chapter 2

Mathematical Background

In this chapter, the theoretical and mathematical concepts, extensively used in this thesis, are introduced. In particular, the fundamentals of optimal control theory and numerical approximation are presented in detail since they are the basis of the undertaken research.

2.1 Optimal Control Theory

Optimal Control Theory is a mathematical framework that deals with the problem of finding the best control strategy to govern the behavior of dynamic systems. The goal is to find the optimal control inputs that drive a system from an initial state to a desired final state while minimizing some measure of performance. Optimal control theory has found applications in diverse fields, including engineering, economics, and management sciences. In particular, in 1919, the American rocket pioneer, Robert H. Goddard posed one of the first aerospace optimal control problems [5].
In this section, the optimal control problem is presented in a mathematical context and various methods for solving such problems are discussed.

2.1.1 Continuous Bolza Problem

The Continuous Bolza problem is a classical formulation of the optimal control problem. It seeks to find the optimal control inputs, $\mathbf{u}(t) \in \mathbb{R}^m$, that minimize the

cost functional

$$J = E(\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f) + \int_{t_0}^{t_f} F(\mathbf{x}(t), \mathbf{u}(t), t)\, dt \tag{2.1}$$

where $\mathbf{x}(t) \in \mathbb{R}^{N_x}$ and $\mathbf{u}(t) \in \mathbb{R}^{N_u}$ are the state and the control of the problem, t_0 is the initial time and t_f the final time; subject to the dynamic constraint

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t), \qquad t \in [t_0, t_f] \tag{2.2}$$

the boundary condition

$$\mathbf{e}(\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f) = \mathbf{0} \tag{2.3}$$

and inequality path constraint

$$\mathbf{h}(\mathbf{x}(t), \mathbf{u}(t), t) \leq \mathbf{0}, \qquad t \in [t_0, t_f] \tag{2.4}$$

The functions E, F, $\mathbf{f}$, $\mathbf{e}$, and $\mathbf{h}$, presented in Equation 2.1-2.4, are defines as:

$$\begin{aligned}
E &: \mathbb{R}^{N_x} \times \mathbb{R} \times \mathbb{R}^{N_x} \times \mathbb{R} \to \mathbb{R} \\
F &: \mathbb{R}^{N_x} \times \mathbb{R}^{N_u} \times \mathbb{R} \to \mathbb{R} \\
\mathbf{f} &: \mathbb{R}^{N_x} \times \mathbb{R}^{N_u} \times \mathbb{R} \to \mathbb{R}^{N_x} \\
\mathbf{e} &: \mathbb{R}^{N_x} \times \mathbb{R} \times \mathbb{R}^{N_x} \times \mathbb{R} \to \mathbb{R}^{N_e} \\
\mathbf{h} &: \mathbb{R}^{N_x} \times \mathbb{R}^{N_u} \times \mathbb{R} \to \mathbb{R}^{N_h}
\end{aligned} \tag{2.5}$$

The problem independent variable is time t, which runs from t_0 to t_f. Since certain numerical techniques, like Pseudospectral methods, have fixed time interval that lies in $[-1, 1]$, a variable transformation to τ so that $[\tau_0, \tau_f] = [-1, 1]$ is needed:

$$t = \frac{1}{2}[(t_f - t_0)\tau + (t_f + t_0)] \tag{2.6}$$

With free initial and/or final times the mapping expressed by Equation 2.6 is still valid and the continuous Bolza problem can be restated as: Minimize the cost function

$$J = E(\mathbf{x}(\tau_0), t_0, \mathbf{x}(\tau_f), t_f) + \frac{t_f - t_0}{2} \int_{\tau_0}^{\tau_f} F(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau; t_0, t_f)\, d\tau \tag{2.7}$$

with the following constraints

$$\dot{\mathbf{x}} = \frac{t_f - t_0}{2}\mathbf{f}(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau; t_0, t_f) \tag{2.8}$$

$$\mathbf{e}(\mathbf{x}(\tau_0), t_0, \mathbf{x}(\tau_f), t_f) = \mathbf{0} \tag{2.9}$$

$$\mathbf{h}(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau; t_0, t_f) \leq \mathbf{0} \tag{2.10}$$

Equation 2.7-2.10 represents the *transformed continuous Bolza problem*.

2.1.2 Indirect Transcription

Calculus of variations can be used to derive a set of first-order conditions for optimality [3, 79, 80] and it solves the transformed Bolza problem, obtained in the previous section. This approach is known as "indirect" because it is focused on the optimality conditions and not on the performance index. One must search for an extremal solution to find the result that satisfies the optimality conditions. This solution must be a minimum and this is verified by using the second-order conditions. The fundamental theorem of calculus of variations states that if $\mathbf{x}^*$ is an extremal, the cost function J must have zero variation at $\mathbf{x}^*$. This can be expressed as:

$$\delta J(\mathbf{x}^*, \delta\mathbf{x}) = \mathbf{0}, \qquad \forall \text{ admissible } \delta\mathbf{x} \tag{2.11}$$

$$\begin{aligned} J_a = E(\mathbf{x}(\tau_0), t_0, \mathbf{x}(\tau_f), t_f) - \nu^T \mathbf{e}(\mathbf{x}(\tau_0), t_0, \mathbf{x}(\tau_f), t_f) \\ + \frac{t_f - t_0}{2} \int_{-1}^{1} \left[F(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau; t_0, t_f) - \lambda^T(t) \left(\frac{d\mathbf{x}}{d\tau} - \mathbf{f}(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau; t_0, t_f) \right) \right. \\ \left. - \mu^T(\tau)\mathbf{h}(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau; t_0, t_f) \right] d\tau \end{aligned} \tag{2.12}$$

As in Equation 2.11, the variation of the augmented cost function is then set to zero for any variation on the problem variables and this leads to the definition of a set of first-order necessary conditions for optimality. To simplify these first-order necessary conditions, it is possible to define an augmented Hamiltonian functional, H:

$$H = (\mathbf{x}, \lambda, \mu, \mathbf{u}, \tau; t_0, t_f) = F(\mathbf{x}, \mathbf{u}, \tau; t_0, t_f) + \lambda^T(t)\mathbf{f}(\mathbf{x}, \mathbf{u}, \tau; t_0, t_f) \\ -\mu^T \mathbf{h}(\mathbf{x}, \mathbf{u}, \tau; t_0, t_f) \tag{2.13}$$

Hence it is possible to define the first-order necessary conditions as the Hamiltonian Boundary Value Problem (HBVP):

$$\begin{gathered}
\frac{d\mathbf{x}^T}{d\tau} = \frac{t_f - t_0}{2}\mathbf{f}(\mathbf{x}, \mathbf{u}, \tau; t_0, t_f) = \frac{t_f - t_0}{2}\,\frac{\partial H}{\partial \lambda} \\
\frac{d\lambda^T}{d\tau} = \frac{t_f - t_0}{2}\left(-\frac{\partial F}{\partial \mathbf{x}} - \lambda^{\mathbf{T}}\frac{\partial \mathbf{f}}{\partial \mathbf{x}} + \mu^{\mathbf{T}}\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right) = -\frac{t_f - t_0}{2}\,\frac{\partial H}{\partial \mathbf{x}} \\
\mathbf{0^T} = \frac{\partial F}{\partial \mathbf{u}} + \lambda^{\mathbf{T}}\frac{\partial \mathbf{f}}{\partial \mathbf{u}} - \mu^{\mathbf{T}}\frac{\partial \mathbf{h}}{\partial \mathbf{u}} = \frac{\partial H}{\partial \mathbf{u}} \\
\mathbf{e}(\mathbf{x}(\tau_\mathbf{0}), t_0, \mathbf{x}(\tau_\mathbf{f}), t_f) = \mathbf{0} \\
\lambda(\tau_0)^T = -\frac{\partial E}{\partial \mathbf{x}(\tau_0)} + \nu^T \frac{\partial \mathbf{e}}{\partial \mathbf{x}(\tau_0)} \\
\lambda(\tau_f)^T = \frac{\partial E}{\partial \mathbf{x}(\tau_f)} - \nu^T \frac{\partial \mathbf{e}}{\partial \mathbf{x}(\tau_f)} \\
H(t_0) = \frac{\partial E}{\partial t_0} - \nu^T \frac{\partial \mathbf{e}}{\partial t_0} \\
H(t_f) = -\frac{\partial E}{\partial t_f} + \nu^T \frac{\partial \mathbf{e}}{\partial t_f} r \\
\mu_j(\tau) = 0 \text{ , when } \quad \mathbf{h}_j(\mathbf{x}, \mathbf{u}, \tau; t_0, t_f) < 0, \quad j = 1, \ldots, c \\
\mu_j(\tau) \leq 0 \text{ , when } \quad \mathbf{h}_j(\mathbf{x}, \mathbf{u}, \tau; t_0, t_f) = 0, \quad j = 1, \ldots, c
\end{gathered} \tag{2.14}$$

When the control appears in the Hamiltonian in a linear form, it cannot be uniquely determined from these optimality conditions. Hence, the weak form of Pontryagin's minimum principle is taken into account, finding the permissible control that globally minimizes H in Equation 2.13. Pontryagin's minimum principle states

that the optimal control, $\mathbf{u}^*$, satisfies:

$$H(\mathbf{x}^*, \mathbf{u}^*, \lambda^*, \mu^*, \tau; t_0, t_f) \leq H(\mathbf{x}^*, \mathbf{u}, \lambda^*, \mu^*, \tau; t_0, t_f), \qquad \forall \mathbf{u} \in U, \quad \tau \in [-1, 1] \tag{2.15}$$

where U denotes the domain of the control variables.
To determine the nature of the stationary point found with the equations presented in this section, a second-order sufficiency check must be performed, to confirm if the extremal solution is a minimum or a maximum.
The use of indirect method in optimal control has several advantages that make it a popular choice for solving optimization problems. It is known for providing high accuracy in the results. It has low computational cost and time, making it use-efficient. Moreover, it provides theoretical insight into the problem, which can be useful in understanding the underlying principles. However, the indirect method also has some disadvantages that should be considered. In fact, treating complex equations or constraints with this method can be difficult, and it may require making preliminary assumptions about the constraint structure. Furthermore, the method may be less robust compared to other methods and may not provide robust results in all cases. Lastly, it can find stationary solutions which may be suboptimal and that are dependent on tentative solutions.

2.1.3 Direct Transcription

By discretizing the problem and constraining the discrete parameters to satisfy algebraic equations, the infinite-dimension Bolza problem can be converted, or transcribed, into a NLP problem. That is, the continuous dynamic system can be optimized without the use of the necessary conditions or costate variables in a "direct" optimization. The steps required for a direct transcription method are [13]:

1. Discretization of the continuous-time dynamics into a finite number of time steps.

2. Formulation of an optimal control problem as a nonlinear programming (NLP) problem, including the objective function and constraints.

3. Solution of the NLP problem using numerical optimization algorithms to obtain the control inputs.

4. Validation of the resulting solution to ensure it meets desired specifications, assessing the accuracy of the finite-dimensional approximation.

5. Iteration of steps 2-4 until an acceptable solution is obtained.

Discretization can be done on the control only, or on both state and control. In the first case, the control inputs are discretized while the dynamics are propagated using the approximate control inputs. It is necessary to ensure that any additional constraints are satisfied and, along with the cost, they help guide the search direction in the NLP optimization problem. On the other hand, a further classification into *local* and *global* methods can be applied to state and control parametrization. When a local method is considered, the dynamics are split into subintervals at the point $t_0 \leq t_1, \ldots, t_i, \ldots, t_N \leq t_f$, and state and control are found as:

$$\mathbf{x}_{i+1} = \mathbf{x} + \int_{t_i}^{t_{i+1}} \mathbf{f}(\mathbf{x}, \mathbf{u}, t)\, dt \tag{2.16}$$

where quadrature approximation are used to replace the integral part of Equation 2.16:

$$\int_{t_i}^{t_{i+1}} \mathbf{f}(\mathbf{x}, \mathbf{u}, t)\, dt \approx h_i \sum_{j=1}^{K} \beta_j \mathbf{f}(\mathbf{x}_j, \mathbf{u}_j, t_j), \qquad t_i \leq \tau_j \leq t_{i+1} \tag{2.17}$$

Some example of local methods are: Euler, Runge-Kutta and Hermite-Simpson. Galerkin, Tau and Collocational methods are instead global methods, also known as Pseudospectral methods [21]. In his work Fornberg [20] extensively presents Pseudospectral methods and their peculiarity to approximate the solution $\mathbf{x}(t)$ by a finite sum:

$$\mathbf{X}(t) = \sum_{k=1}^{M} a_k \phi_k(t) \tag{2.18}$$

where $\phi_k(t)$, $k = 1, \ldots, M$ are called trials, expansion, or approximating functions, indistinctly. They are the basis for the truncated series expansion of the solution and they usually are trigonometric functions or orthogonal polynomials. To determine the different element a_k, presented in Equation 2.18, different test functions, which try to ensure satisfaction of the differential equations, are used in Tau, Galerkin, and Collocational methods. For example, Tau method picks expansion coefficients that satisfy the boundary conditions and that warrant that the residual, defined as:

$$\mathbf{R}_M(t)\phi_i(t) = \dot{\mathbf{X}}(t) - \mathbf{f}(\mathbf{X}(t), \mathbf{U}(t), t) \tag{2.19}$$

is orthogonal to the basis functions. That means that the inner product between the residual and the basis function is zero:

$$< \mathbf{R}_M(t)\phi_i(t) > \equiv \int_{t_0}^{t_f} \mathbf{R}_M(t)\phi_i(t)\, dt = 0, \quad \forall i = 1,\ldots,M \tag{2.20}$$

When considering a Galerkin method, $\widehat{\phi}_i(t),\ i = 1,\ldots,M$ is a new set created by combining the original basis functions and in which all the functions satisfy the boundary conditions. Here the expansion coefficients are those in which the residual is orthogonal to the new basis functions. As regard a collocational method, in this case, the test functions are the Dirac delta functions and the residual must be null at suitably chosen sets of collocation points, as:

$$\mathbf{R}_M(t_k) = \mathbf{0}, \quad \forall k = 1,\ldots,M \tag{2.21}$$

Satisfying Equation 2.21 and the boundary conditions, the expansion coefficients for the collocational methods are selected.
To summarize, direct methods have several advantages and disadvantages, such as:

- Pros
  - They have the capability of treating complex problems
  - They are well-suited to problems with constraints, as they can be easily incorporated into the system of equations
  - They have high robustness
- Cons
  - They may not always produce highly accurate solutions. In some cases, refinement of the solution may be required to achieve the desired level of accuracy
  - They can be computationally expensive
  - The solution may depend on tentative values and it may be suboptimal

2.2 Numerical Approximation Methods

In this section, we delve into the mathematical aspects of discretization and approximation methods required for direct Pseudospectral transcription. These methods are not only applicable to direct Pseudospectral transcription but can also be employed to discretize the Hamiltonian boundary value problem in the context of indirect Pseudospectral transcription.

2.2.1 Global Polynomial Approximations

This section explores the concept of global approximation in the context of Pseudospectral methods. Unlike the local collocation methods, such as the Hermite-Simpson scheme, which utilize piecewise-continuous functions on arbitrary subintervals, Pseudospectral methods employ globally interpolating Lagrange polynomials as basis functions to approximate the state and control across the entire interval, $\tau \in [-1,1]$ resulting in a global collocational scheme. These polynomials are defined using a set of $(N+1)$ point, $\tau_0,\ldots,\tau_N$, on the time interval. The approximation of the state, control, and costate of the optimal control problem is performed using [81]:

$$y(\tau) \approx y^N(\tau) = \sum_{l=0}^{N} \phi_l y^N(\tau_l) \tag{2.22}$$

where $y^N(\tau)$is a polynomial approximation of degree N on the interval $[-1,1]$ and $\phi_l(\tau)$ $(l=0,\ldots,N)$ are the Lagrange polynomials, defined as:

$$\phi_l(\tau) = \prod_{j=0, j\neq l}^{N} \frac{\tau-\tau_j}{\tau_l-\tau_j} = \frac{g(\tau)}{(\tau-\tau_l)\dot{g}(\tau)} \tag{2.23}$$

$g(\tau)$ is used to define the positions of the support points since it generates the trial function (often related to Legendre and Chebyshev polynomials), while $\dot{g}(\tau)$ is its time derivative. For Lagrange polynomial, it can be shown that:

$$\phi_l(\tau_j) = \begin{cases} 1, & l=j \\ 0, & l\neq j \end{cases} \tag{2.24}$$

From this property, it follows that the function approximation is equal to the true function at the $(N+1)$ points:

$$y(\tau_j) = y^N(\tau_j) \tag{2.25}$$

When it comes to discretizing a continuous-time interval, a straightforward approach is to partition the interval into equally spaced support points. However, this discretization scheme has limitations when it comes to polynomial approximation. The error between the polynomial approximation and the true function is not necessarily reduced as the number of support points increases, particularly when the support points are uniformly spaced. This phenomenon, known as the Runge phenomenon [82], results in an increase in approximation error near the boundaries as the degree of the polynomial approximation increases. This phenomenon is clearly shown in Figure 2.1. Here, a Lagrange approximation of the function $f(x) = 1/(1+25x^2)$ using 25 equally spaced points is presented and the poor accuracy of the approximation near the boundaries is evident.

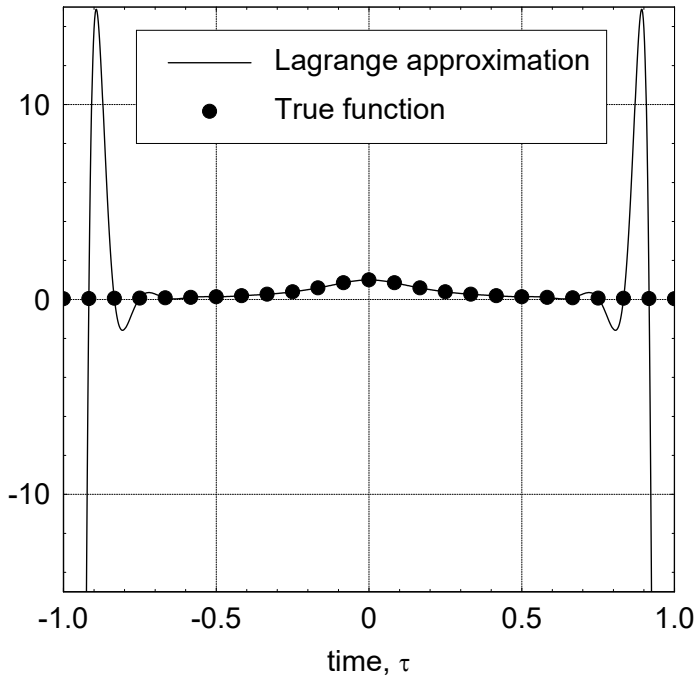


Fig. 2.1 Lagrange polynomial approximation of the function $f(x) = 1/(1+25x^2)$ using 25 equidistant points.

This issue has been widely studied in literature [83, 84] and various methods have been proposed to address it. One common approach is to use non-uniform support points, such as the ones found using the roots of Chebyshev or Legendre polynomials, which are specially chosen to reduce the Runge phenomenon. These points have the characteristic to be denser towards the boundaries of the problem. The comparison between equidistant points and non-uniformly spaced points, specifically Legendre-Gauss-Lobatto points, is shown in Figure 2.2. The error is calculated as the maximum absolute difference between the polynomial approximation and the true function, e^t, and it is displayed in a logarithmic scale to better visualize the convergence. The results show that the Lagrange approximation using LGL points has a faster convergence rate and a smaller error than the uniform points, especially for high degrees.

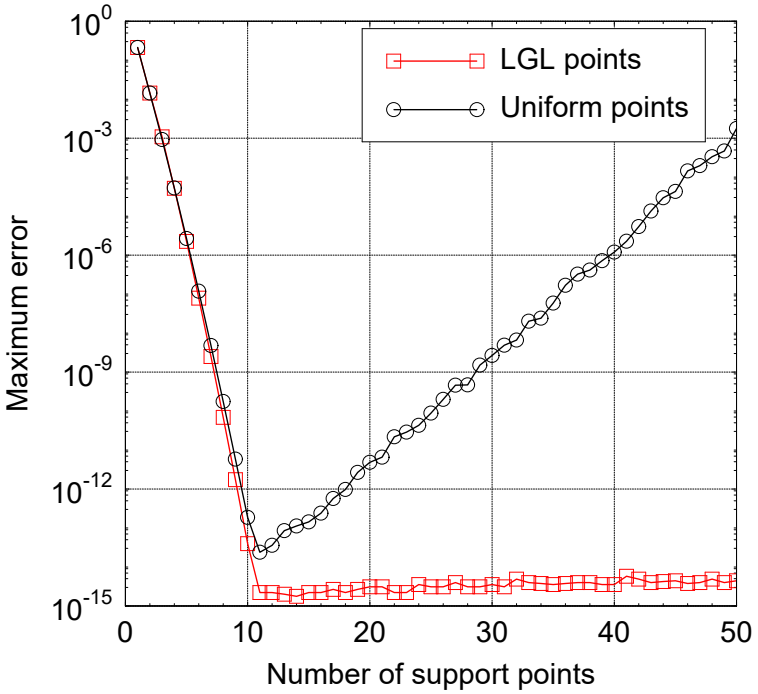


Fig. 2.2 Comparison of approximation accuracy for function $f(x) = e^t$ using uniform and non-uniform spacing points in function of the number of support points considered.

These aspects are emphasized in subsection 4.1.3 where a comparative analysis between Pseudospectral methods and the Trapezoidal Collocation method (TC)

[13] is presented. In this context, Pseudospectral methods employing non-uniform support points exhibit superior performance compared to TC, which relies on equally spaced support points.

2.2.2 Gaussian Quadrature approximation

In the optimal control problem presented in section 2.1 the integral part of the cost functional in Equation 2.7 and the dynamics constraints in Equation 2.8 must be discretized and approximated with a high degree of accuracy. Hence, when considering a Pseudospectral method, the concept is to approximate:

$$\int_a^b f(\tau)\,d\tau \approx \sum_{i=1}^{K} w_i f(\tau_i) \tag{2.26}$$

by choosing the quadrature points $(\tau_1,\ldots,\tau_k)$ on the interval $\tau \in [-1,1]$ and the quadrature weights $(w_i,\, i=1,\ldots,K)$ in some optimal fashion, aiming to minimize the error (or eliminate it entirely) for polynomials f of the highest possible degree. Because there are $2K$ free parameters (the points and the weights), this integral approximation should be able to achieve a $2K-1$ degree of precision. Different approaches can be used in Gaussian quadrature. For example, it is possible to consider as the quadrature points, or nodes K Legendre-Gauss points, and they can be defined as the roots of the Legendre polynomial of degree K^{th}:

$$L_K(\tau) = \frac{1}{2^K K!}\frac{d^K}{d\tau^K}\left[\left(\tau^2-1\right)^K\right] \tag{2.27}$$

While the weights are calculated by:

$$w_i = \int_{-1}^{1} \tilde{\phi}_i(\tau)\,d\tau = \frac{2}{\left(1-\tau_i^2\right)\left[\dot{L_K}(\tau_i)\right]^2}, \quad i=1,\ldots,K \tag{2.28}$$

where $\dot{L_K}$ is the derivative of the K^{th}-degree Legendre polynomial. LG nodes lie in $\tau \in (-1,1)$.

Another possible approach uses the Legendre-Gauss-Radau (LGR) as nodes. They are defined as the roots of $L_K(\tau)+L_{K-1}(\tau)$. These support points lie on $\tau=[-1,1)$. This leads to the reduction of the degree of freedom, making this choice of points accurate to $2K-2$ degree of precision. As regard the weights for the LGR points,

they can be calculated as:

$$w_1 = \frac{2}{K^2}$$
$$w_i = \frac{1}{(1-\tau_i)\left[\dot{L}_{K-1}(\tau_i)\right]^2}, \quad i = 2,\ldots,K \tag{2.29}$$

Legendre-Gauss-Lobatto points are often used in Pseudospectral methods and lie within the interval $[-1,1]$. This approach is unique in that it requires the quadrature points to be located at both endpoints, reducing the degree of freedom by two and making the scheme accurate up to $2K-3$. By finding the roots of $(1-\tau^2)\dot{L}_{K-1}(\tau)$, it is possible to determine the K LGL points, where $\dot{L}_{K-1}(\tau)$ is the derivative of the $(K-1)^{th}$ order Legendre polynomial. For these points, the corresponding weights can be expressed as:

$$w_i = \frac{2}{K(K-1)}\frac{1}{[L_{k-1}(\tau_i)]^2}, \quad i = 1,\ldots,K \tag{2.30}$$

Figure 2.3 presents the position of 10 points per each type of quadrature method.

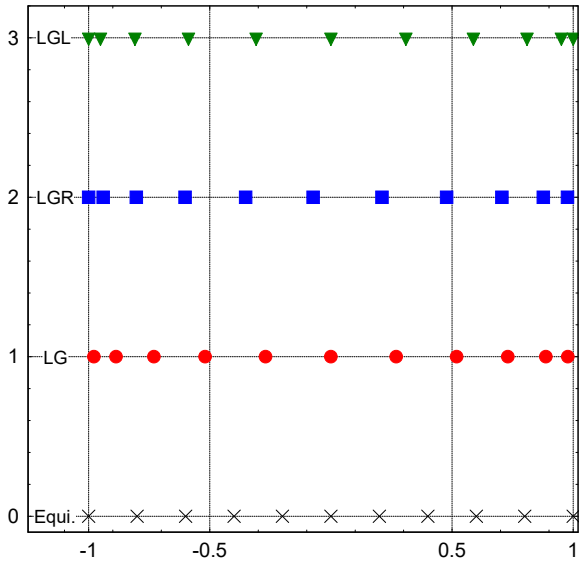


Fig. 2.3 Comparison of position of 10 points using equidistant, Legendre-Gauss, Legendre-Gauss-Radau and Legendre-Gauss-Lobatto collocation points

2.2.3 Orthogonal Collocation

To transform the dynamic equations into algebraic conditions, the Pseudospectral methods use orthogonal collocation. Assuming that K is the number of the considered collocation points and $N+1$ are the points used to approximate the state. It is possible to calculate the derivative of the approximated state at τ_k, the k-th collocation points, as:

$$\dot{\mathbf{x}}(\tau_k) \approx \dot{\mathbf{x}}^N(\tau_k) = \sum_{l=0}^{N} \dot{\phi}_l(\tau_k)\mathbf{x}^N(\tau_l) = \sum_{l=0}^{N} D_{kl}\mathbf{x}^N(\tau_k), \quad k = 1,\dots,K \tag{2.31}$$

$D \in \mathbb{R}^{K\times N+1}$ is the differentiation matrix and it is expressed as:

$$D_{kl} = \begin{cases} \frac{\dot{g}(\tau_k)}{(\tau_k-\tau_l)\dot{g}(\tau_l)}, & \text{if } k \neq l \\ \frac{\ddot{g}(\tau_l)}{2\dot{g}(\tau_l)}, & \text{if } k = l \end{cases} \tag{2.32}$$

In this way, the differential equation is replaced by a set of algebraic conditions evaluated at the collocation points. In the nonlinear program the continuous differential equations are transformed in a set of residuals:

$$\mathbf{R_k} = \sum_{l=0}^{N} D_{kl}\mathbf{x}^N(\tau_l) - \frac{t_f - t_0}{2}\mathbf{f}\left(\mathbf{x}^N(\tau_k), \mathbf{u}^N(\tau_k), \tau_k; t_0, t_f\right) = \mathbf{0}, \quad k = 1,\dots,K \tag{2.33}$$

Chapter 3

Methodology

This chapter introduces the Legendre Pseudospectral method, which forms the foundation of the methodology utilized in this study. An indirect formulation of the method is also presented, which utilizes a smoothing technique to enhance its convergence. Additionally, to gain a better understanding of the various approaches and how their results differ, we apply three different Pseudospectral methods to solve a single-state problem and an orbit-raising problem. Lastly, the Bellman Pseudospectral method is presented. It incorporates Bellman's principle of optimality and maps the discrete solution to the continuous time domain.

3.1 The Legendre Pseudospectral Method

In the subsequent section, we will outline the approach to resolving the Bolza Problem by utilizing its approximation. To ensure clarity, it is essential to provide clear definitions for the following problems:

- *Problem B*: original continuous Bolza problem
- *Problem* B^N: discretization of *Problem B* through LGL points
- *Problem* B^λ: Bolza problem in the primal-dual space
- *Problem* $B^{\lambda N}$: discretization of *Problem* B^λ through LGL points
- *Problem* $B^{N\lambda}$: dualization to *Problem* B^N

These problems will be further elaborated on in their application within the Legendre Pseudospectral method.
The Direct Pseudospectral Method is a cutting-edge technique for solving optimal control problems, which was first introduced by Elnagar et al. [19]. In their work, they applied this method to address a linear quadratic control problem. Since then, Fahroo and Ross have made significant contributions to the development and refinement of the Pseudospectral Method [44, 85–87].
The Direct Pseudospectral Method consists of two key steps [44, 88]:

1. A global polynomial approximation for the state and control functions in terms of their values at the Legendre-Gauss-Lobatto (LGL) points is adopted

2. Equations are imposed that approximate the solution to the original problem by requiring that the state equations are exactly satisfied by their approximations at the LGL points.

The polynomials used in this method are globally interpolating Lagrange polynomials obtained from orthogonal Legendre polynomials. Hence, by definition, the coefficients of the polynomial expansion are exactly the values of the functions at the LGL points. By utilizing the same LGL points to discretize both the integral and differential portions of the problem, already presented in subsection 2.1.1, the optimal control problem is transformed into a nonlinear programming (NLP) problem for the values of the states and controls at the LGL nodes.
The problem independent variable is time t, which runs from t_0 to t_f. Since the LGL points lie in the interval $[-1,1]$, a variable transformation to τ so that $[\tau_0, \tau_f] = [-1,1]$ is needed:

$$t = 0.5[(t_f - t_0)\tau + (t_f + t_0)] \tag{3.1}$$

Hence *Problem B* can now be stated as:

$$Problem\,B = \begin{cases} Minimize: J(\mathbf{u}(\cdot),\mathbf{x}(\cdot),t_f) = & E[\mathbf{x}(-1),\mathbf{x}(1),t_0,t_f]+ \\ & \frac{t_f-t_0}{2}\int_{-1}^{1} F[\mathbf{x}(\tau),\mathbf{u}(\tau)]d\tau \\ \dot{\mathbf{x}}(\tau) = \frac{t_f-t_0}{2}\mathbf{f}[\mathbf{x}(\tau),\mathbf{u}(\tau)] \\ \mathbf{e}[\mathbf{x}(-1),\mathbf{x}(1),t_0,t_f] = \mathbf{0} \\ \mathbf{h}[\mathbf{x}(\tau),\mathbf{u}(\tau)] \leq \mathbf{0} \end{cases} \tag{3.2}$$

Here $L_N(\tau)$is the Legendre polynomial of degree N on the interval $[-1,1]$. The LGL points, $\tau_l, l = 0,...,N$, are the initial and final points ($\tau_0 = -1$ and $\tau_N = +1$) and the zeros of the derivative of the Legendre polynomial $\dot{L}_N$ when $1 \leq l \leq N-1$. We approximate the continuous variables by N^{th} degree polynomials of the form

$$\mathbf{x} \approx \mathbf{x}^N(\tau) = \sum_{l=0}^{N} \mathbf{x}(\tau_l)\phi_l(\tau) \tag{3.3}$$

$$\mathbf{u} \approx \mathbf{u}^N(\tau) = \sum_{l=0}^{N} \mathbf{u}(\tau_l)\phi_l(\tau) \tag{3.4}$$

where $\phi(\tau)$ are the Lagrange polynomials of order N. They can be expressed as:

$$\phi_l(\tau) = \prod_{\substack{k=0 \\ k\neq l}}^{N} \frac{\tau-\tau_k}{(\tau_l-\tau_k)} = \frac{z(\tau)}{(\tau-\tau_l)\dot{z}(\tau_l)} \tag{3.5}$$

where $z(\tau)$ is an intermediate function, defined as:

$$z(\tau) = \prod_{k=0}^{N}(\tau-\tau_k) \tag{3.6}$$

$z(\tau)$ contains the roots of $\dot{L}_N(\tau)$ plus the two endpoints (1 and -1).

$$z(\tau) = (\tau-\tau_0)\dot{L}_N(\tau)(\tau-\tau_N) = ((\tau+1)\dot{L}_N(\tau)(\tau-1) = (\tau^2-1)\dot{L}_N(\tau) \tag{3.7}$$

When considering the Legendre equation:

$$\frac{d}{dt}\left[\left(\tau^2-1\right)\dot{L}_N(\tau)\right] = N(N-1)L_N(\tau) \tag{3.8}$$

Hence:

$$\dot{z}(\tau) = N(N-1)L_N(\tau) \tag{3.9}$$

Considering Equation 3.7 and Equation 3.9, it is possible to write:

$$\phi_l(\tau) = \frac{\left(\tau^2-1\right)\dot{L}_N(\tau)}{(\tau-\tau_l)N(N-1)L_N(\tau)} \tag{3.10}$$

The Lagrange polynomials of order N satisfy the Kronecker relationship:

$$\phi_l(\tau_k) = \delta_{lk} = \begin{cases} 1 & if \quad l = k \\ 0 & if \quad l \neq k \end{cases} \tag{3.11}$$

This implies that:

$$\mathbf{x}^N(\tau_l) = \mathbf{x}(\tau_l),\ \mathbf{u}^N(\tau_l) = \mathbf{u}(\tau_l) \tag{3.12}$$

Differentiating and evaluating Equation 3.10 at node k yields:

$$\dot{\phi}_l(\tau_k) = \frac{1}{N(N-1)L_N(\tau_l)}\left[\frac{2\tau_k\dot{L}_N(\tau_k)}{\tau_k-\tau_l} + \frac{\left(\tau_k^2-1\right)\ddot{L}_N(\tau_k)}{\tau_k-\tau_l} - \frac{\left(\tau_k^2-1\right)\dot{L}_N(\tau_k)}{(\tau_k-\tau_l)^2}\right] \tag{3.13}$$

Considering Equation 3.8, it is possible to write:

$$N(N-1)L_N(\tau) = 2\tau\dot{L}_N(\tau) + \left(\tau^2-1\right)\ddot{L}_N(\tau) \tag{3.14}$$

This enables to simplify Equation 3.13:

$$\dot{\phi}_l(\tau_k) = \frac{1}{N(N-1)L_N(\tau_l)}\left[\frac{N(N-1)L_N(\tau_k)}{\tau_k-\tau_l} - \frac{\left(\tau_k^2-1\right)\dot{L}_N(\tau_k)}{(\tau_k-\tau_l)^2}\right] \tag{3.15}$$

As required by the second step of the method, the condition that the approximate solutions satisfy the differential equations exactly at the specific LGL points must be imposed. The derivative $\dot{\mathbf{x}}^N(\tau)$ in terms of $\mathbf{x}^N(\tau)$ at the collocation points τ_k can be found differentiating Equation 3.12 and evaluating the result at τ_k to obtain a matrix multiplication:

$$\dot{\mathbf{x}}^N(\tau_k) = \sum_{l=0}^{N} D_{kl}\mathbf{x}(\tau_l) \tag{3.16}$$

where D_{kl} are entries of the $(N+1)\times(N+1)$ differentiation matrix $\mathbf{D}$
Since the product $\left(\tau_k^2-1\right)\dot{L}_N(\tau_k)=0$ in Equation 3.15:

$$D_{kl}=\dot{\phi}_l(\tau_k)=\frac{L_N(\tau_k)}{(\tau_k-\tau_l)L_N(\tau_l)} \tag{3.17}$$

By further manipulation, it is possible to obtain the differentiation matrix values at the boundaries and when $k=l$. To eliminate the terms $(\tau_k-\tau_l)$ in the denominator, we apply L'Hôpital's Rule to Equation 3.15. The resulting terms give us the differentiation matrix:

$$\mathbf{D}:=[D_{kl}]:=\begin{cases}\frac{L_N(\tau_k)}{L_N(\tau_l)}\cdot\frac{1}{\tau_k-\tau_l}, & \text{if } k\neq l\\ -\frac{N(N+1)}{4}, & \text{if } k=l=0\\ \frac{N(N+1)}{4}, & \text{if } k=l=N\\ 0, & \text{otherwise}\end{cases} \tag{3.18}$$

The NLP problem can now be formulated as *Problem* B^N. The integral part in Equation 3.90 can be discretized. Hence, using the Gauss-Lobatto integration rule yields:

$$\begin{aligned}J^N(\mathbf{x},\mathbf{u},t_f)&=E\left[\mathbf{x}_0,\mathbf{x}_N,t_0,t_f\right]+\frac{t_f-t_0}{2}\int_{-1}^{1}F\left[\mathbf{x}(\tau),\mathbf{u}(\tau)\right]\,d\tau\\&=E\left[\mathbf{x}_0,\mathbf{x}_N,t_0,t_f\right]+\frac{t_f-t_0}{2}\sum_{k=0}^{N}F(\mathbf{x}_k,\mathbf{u}_k)w_k\end{aligned} \tag{3.19}$$

where w_k are the weights given by:

$$w_k=\frac{2}{N(N-1)}\frac{1}{\left[L_N(\tau_k)\right]^2},\qquad k=0,1,\ldots,N \tag{3.20}$$

Subject to:

$$\left[\frac{t_f-t_0}{2}\right]\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)-\sum_{l=0}^{N}D_{kl}\mathbf{x}_l=\mathbf{0},\qquad k=0,\ldots,N \tag{3.21}$$

$$\mathbf{e}\left[\mathbf{x}_0,\mathbf{x}_N,t_0,t_N\right]=\mathbf{0} \tag{3.22}$$

$$\mathbf{h}(\mathbf{x}_k,\mathbf{u}_k)\leq\mathbf{0},\qquad k=0,\ldots,N \tag{3.23}$$

The previous equations demonstrate the method's simplicity, which preserves most of the structure of the continuous problem. This simplicity is achieved by collocating the equations at the LGL points, and the functions, except for the differentiation matrix **D** that relates the different $\mathbf{x}_k$, are evaluated only at the LGL points, without depending on neighboring points. It's worth noting that the state differential constraint in Equation 3.21, which approximates the derivative in the state equations using the differentiation matrix, differs significantly from the approximation of the state equations achieved by implicit integration rules.

3.1.1 Covector Mapping Theorem

The Covector Mapping Theorem (CMT) uses a set of simple "closure conditions", imposed on the discrete primal-dual variables, so that a linear diagonal transformation of the Lagrange multipliers of the discrete problem provides a consistent approximation to the discrete covectors of the Bolza problem [88–90]. The theorem demonstrates how to achieve the high performance of the indirect method through the significantly simpler implementation of the direct method. The solution of the direct formulation can in fact be used to reconstruct the costates of the indirect formulation without the need to introduce them as additional variables.
In section 3.1, *Problem B* and *Problem* B^N have been presented. Possible optimal solution to *Problem B* can be found, through the Minimum Principle, searching for vector-covector pairs in the primal-dual space. This can be denoted as *Problem* B^λ that satisfies Equation 3.90 and:

$$\dot{\lambda}(t) = -\frac{\partial L(t)}{\partial \mathbf{x}} \tag{3.24}$$

$$\frac{\partial L(t)}{\partial \mathbf{u}} = \mathbf{0} \tag{3.25}$$

$$\lambda(t_0) = -\frac{\partial E_e}{\partial \mathbf{x}(t_0)} \tag{3.26}$$

$$\lambda(t_f) = \frac{\partial E_e}{\partial \mathbf{x}(t_f)} \tag{3.27}$$

$$H(t_0) = \frac{\partial E_e}{\partial t_0} \tag{3.28}$$

$$H(t_f) = -\frac{\partial E_e}{\partial t_f} \tag{3.29}$$

Here L defines the D-form of the Lagrangian of the Hamiltonian [91]:

$$L(\mathbf{x}, \mathbf{u}, \lambda, \mu) = H(\mathbf{x}, \mathbf{u}, \lambda) + \mu^T \mathbf{h}(\mathbf{x}, \mathbf{u}) \tag{3.30}$$

μ satisfies the complementary condition:

$$\mu^T(t)\mathbf{h}(t) = 0 \qquad \forall t \in [t_0, t_f] \tag{3.31}$$

and E_e is expressed as:

$$E_e\left(\mathbf{x}(t_0), \mathbf{x}(t_f), t_0, t_f, \nu\right) = E\left(\mathbf{x}(t_0), \mathbf{x}(t_f), t_0, t_f\right) + \nu^T \mathbf{e}\left(\mathbf{x}(t_0), \mathbf{x}(t_f), t_0, t_f\right) \tag{3.32}$$

It is also possible to discretize *Problem* B^{λ}, approximating the costate by the N^{th} degree polynomial:

$$\lambda(t(\tau)) \approx \lambda^N(t(\tau)) = \sum_{l=0}^{N} \lambda_l \phi_l(\tau) \tag{3.33}$$

This leads to *Problem* $B^{\lambda N}$ that wants to find $\mathbf{X_{NP}} = \left(\mathbf{X}^N; \mathbf{U}^N\right)$ and $\Lambda_{NP} = (\lambda_0; \ldots; \lambda_N; \mu_0; \ldots; \mu_N; \nu_0; \nu_f)$ satisfying Equation 3.21-Equation 3.23 and for $k = 0, \ldots, N$:

$$\sum_{l=0}^{N} D_{kl} \lambda_l = -\frac{\partial L_k}{\partial \mathbf{x}_k} \tag{3.34}$$

$$\frac{\partial L_k}{\partial \mathbf{u}_k} = \mathbf{0} \tag{3.35}$$

$$\lambda_0 = -\frac{\partial E_e}{\partial \mathbf{x}_0} \tag{3.36}$$

$$\lambda_N = \frac{\partial E_e}{\partial \mathbf{x}_f} \tag{3.37}$$

$$H_0 = \frac{\partial E_e}{\partial t_0} \tag{3.38}$$

$$H_f = -\frac{\partial E_e}{\partial t_f} \tag{3.39}$$

$$\mu_k^T \mathbf{h}_k = \mathbf{0}, \qquad \mu_k \geq \mathbf{0} \tag{3.40}$$

It is also possible to apply a dualization to *Problem* B^N. On that account, the Karush-Kuhn-Tucker (KKT) conditions for *Problem* B^N are derived. Defining the Lagrangian:

$$\bar{J}^N\left(\mathbf{X}_{NP},\tilde{\nu},\tilde{\lambda},\tilde{\mu}\right)=J^N(\mathbf{X}_{NP})+\tilde{\nu}^T\mathbf{e}(\mathbf{x}_0,\mathbf{x}_N,t_0,t_f)+ \tag{3.41}$$
$$\sum_{i=0}^{N}\left(\tilde{\lambda}_i^T\left[\left(\frac{t_f-t_0}{2}\right)\mathbf{f}_i(\mathbf{X}_{NP})-\mathbf{d}_i(\mathbf{X}^N)\right]+\tilde{\mu}_i^T\mathbf{h}_i(\mathbf{X}_{NP})\right)$$

where $\tilde{\nu},\tilde{\lambda},\tilde{\mu}$ are the KKT multipliers associated with the NLP. The KKT first-order necessary conditions can be written as [44]:

$$\frac{\partial\bar{J}^N}{\partial\mathbf{x}_k}=\mathbf{0},\quad \frac{\partial\bar{J}^N}{\partial\mathbf{u}_k}=\mathbf{0},\quad \frac{\partial\bar{J}^N}{\partial t_N}=\mathbf{0}\quad \frac{\partial\bar{J}^N}{\partial t_0}=\mathbf{0} \tag{3.42}$$

This results in:

$$\frac{\partial L}{\partial\mathbf{x}_k}\left(\mathbf{x}_k,\mathbf{u}_k,\frac{\tilde{\lambda}_k}{w_k},\frac{\tilde{\mu}_k}{w_k}\right)+\sum_{i=0}^{N}D_{ki}\left(\frac{\tilde{\lambda}_i}{w_i}\right)=\mathbf{0}\qquad k=1,\ldots,N-1 \tag{3.43}$$

Hence at the interior points the LGL-weight-normalized multipliers $\frac{\tilde{\lambda}_k}{w_k},\frac{\tilde{\mu}_k}{w_k}$ satisfy the same equations as the discrete costates (Equation 3.34). Moreover, at the initial and final nodes, the KKT multipliers satisfy the following condition:

$$-w_0\left(\frac{\partial L}{\partial\mathbf{x}_0}\left(\mathbf{x}_0,\mathbf{u}_0,\frac{\tilde{\lambda}_0}{w_0},\frac{\tilde{\mu}_0}{w_0}\right)+\sum_{i=0}^{N}D_{0i}\frac{\tilde{\lambda}_i}{w_i}\right)=\frac{\tilde{\lambda}_0}{w_0}+\frac{\partial\tilde{E}_e}{\partial\mathbf{x}_0}\equiv\mathbf{c}_0 \tag{3.44}$$

$$w_N\left(\frac{\partial L}{\partial\mathbf{x}_N}\left(\mathbf{x}_N,\mathbf{u}_N,\frac{\tilde{\lambda}_N}{w_N},\frac{\tilde{\mu}_N}{w_N}\right)+\sum_{i=0}^{N}D_{Ni}\frac{\tilde{\lambda}_i}{w_i}\right)=\frac{\tilde{\lambda}_N}{w_N}+\frac{\partial\tilde{E}_e}{\partial\mathbf{x}_N}\equiv\mathbf{c}_N \tag{3.45}$$

where $\tilde{E}_e=E_e(\mathbf{x}_0,\mathbf{x}_N,t_0,t_N,\tilde{\nu})$.
Applying the KKT condition for the variables t_N,t_0 results in Lagrange multipliers that satisfy also:

$$\frac{1}{2}\sum_{i=0}^{N}w_iH\left(\mathbf{x}_i,\mathbf{u}_i,\frac{\tilde{\lambda}_i}{w_i}\right)=-\frac{\partial\tilde{E}_e}{\partial t_N} \tag{3.46}$$

$$\frac{1}{2}\sum_{i=0}^{N}w_iH\left(\mathbf{x}_i,\mathbf{u}_i,\frac{\tilde{\lambda}_i}{w_i}\right)=\frac{\partial\tilde{E}_e}{\partial t_0} \tag{3.47}$$

Hence it is possible to formulate *Problem* $B^{N\lambda}$ as: Find $\mathbf{X}_{NP}$ and $\tilde{\Lambda}_{NP} = (\tilde{\lambda}_0; \ldots; \tilde{\lambda}_N; \tilde{\mu}_0; \ldots; \tilde{\mu}_N; \tilde{\nu}_0; \tilde{\nu_f})$ that satisfy Equation 3.21-Equation 3.23 in addition to the following nonlinear algebraic relations:

$$\frac{\partial L_k}{\partial \mathbf{u}_k}\left(\mathbf{x}_k, \mathbf{u}_k, \frac{\tilde{\lambda}_k}{w_k}, \frac{\tilde{\mu}_k}{w_k}\right) = \mathbf{0} \qquad k = 0, \ldots, N \tag{3.48}$$

$$\tilde{\mu}_k^T \mathbf{h}_k = \mathbf{0}, \qquad \tilde{\mu}_k \geq \mathbf{0}, \qquad k = 0, \ldots, N \tag{3.49}$$

$$\frac{\partial L}{\partial \mathbf{x}_k}\left(\mathbf{x}_k, \mathbf{u}_k, \frac{\tilde{\lambda}_k}{w_k}, \frac{\tilde{\mu}_k}{w_k}\right) + \sum_{i=0}^{N} D_{ki}\left(\frac{\tilde{\lambda}_i}{w_i}\right) = \mathbf{0} \qquad k = 1, \ldots, N-1 \tag{3.50}$$

$$\frac{\partial L}{\partial \mathbf{x}_0}\left(\mathbf{x}_0, \mathbf{u}_0, \frac{\tilde{\lambda}_0}{w_0}, \frac{\tilde{\mu}_0}{w_0}\right) + \sum_{i=0}^{N} D_{0i}\frac{\tilde{\lambda}_i}{w_i} = -\frac{\mathbf{c}_0}{w_0} \tag{3.51}$$

$$\frac{\tilde{\lambda}_0}{w_0} + \frac{\partial \tilde{E}_e}{\partial \mathbf{x}_0} = \mathbf{c}_0 \tag{3.52}$$

$$\frac{\partial L}{\partial \mathbf{x}_N}\left(\mathbf{x}_N, \mathbf{u}_N, \frac{\tilde{\lambda}_N}{w_N}, \frac{\tilde{\mu}_N}{w_N}\right) + \sum_{i=0}^{N} D_{Ni}\frac{\tilde{\lambda}_i}{w_i} = \frac{\mathbf{c}_N}{w_N} \tag{3.53}$$

$$\frac{\tilde{\lambda}_N}{w_N} - \frac{\partial \tilde{E}_e}{\partial \mathbf{x}_N} = \mathbf{c}_N \tag{3.54}$$

$$\frac{1}{2}\sum_{i=0}^{N} w_i H\left(\mathbf{x}_i, \mathbf{u}_i, \frac{\tilde{\lambda}_i}{w_i}\right) = -\frac{\partial \tilde{E}_e}{\partial t_N} \tag{3.55}$$

$$\frac{1}{2}\sum_{i=0}^{N} w_i H\left(\mathbf{x}_i, \mathbf{u}_i, \frac{\tilde{\lambda}_i}{w_i}\right) = \frac{\partial \tilde{E}_e}{\partial t_0} \tag{3.56}$$

The KKT conditions for *Problem* B^N are intentionally expressed in a way that makes it easier to define Closure Conditions. Closure Conditions can be described as a collection of restrictions that need to be imposed on *Problem* $B^{N\lambda}$ to ensure that any solution found for this limited problem is equivalent to the solution of *Problem* $B^{\lambda N}$. By comparing the equations of *Problems* $B^{N\lambda}$ and $B^{\lambda N}$, the Closure Conditions can be derived straightforwardly:

$$\mathbf{c}_0 = \mathbf{0} \tag{3.57}$$

$$\mathbf{c}_N = \mathbf{0} \tag{3.58}$$

$$\frac{1}{2}\sum_{i=0}^{N} w_i H\left(\mathbf{x}_i, \mathbf{u}_i, \frac{\tilde{\lambda}_i}{w_i}\right) = H_0 = H_N \tag{3.59}$$

Hence it is possible to state the Covector Mapping Theorem as [88]: *There exist Lagrange multipliers $\tilde{\lambda}_i$, $\tilde{\mu}_i$ that are equal to the Pseudospectral approximations of the covectors $\lambda^N(t_i)$, $\mu^N(t_i)$, at the shifted LGL node t_i multiplied by the corresponding LGL weight w_i. Furthermore, there exists a $\tilde{\nu}$ that is equal to the constant covector ν.* We can represent this relationship as:

$$\lambda^N(t_i) = \frac{\tilde{\lambda}_i}{w_i}, \qquad \mu^N(t_i) = \frac{\tilde{\mu}_i}{w_i}, \qquad \tilde{\nu} = \nu \tag{3.60}$$

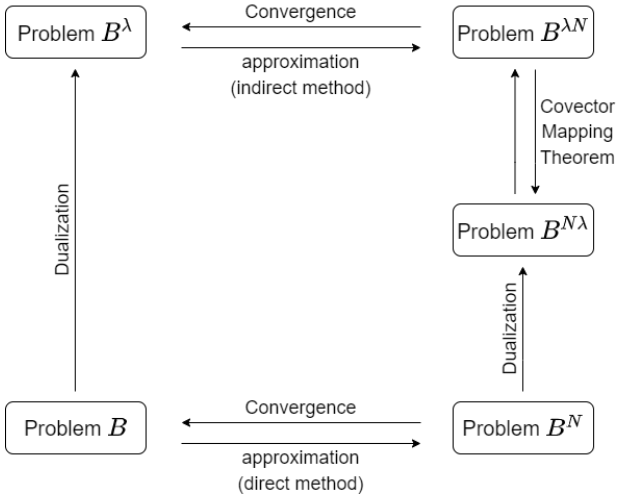


Fig. 3.1 Visual representation of the covector mapping principle, showcasing the unification of direct and indirect methods in optimal control, as initially proposed by Ross and Fahroo [88, 92]

The core of the theorem can be seen in Figure 3.1. It shows how the solution to *Problem $B^{N\lambda}$* can always be obtained from the solution to *Problem $B^{\lambda N}$*, but the reverse is not necessarily true unless Closure Conditions are in place.

3.2 The Indirect Legendre Pseudospectral Method

PS methods can also be used in the context of indirect optimization methods to discretize the resulting boundary value problem [93, 65, 66].

Here a modified Pseudospectral method for an efficient numerical solution of a class of switching optimal control problems is considered. This class of optimal control problems, which are usually named bang-bang optimal control problems, arises in trajectory design of a spacecraft. Moreover, due to the lack of basic information about the optimal control structure, the numerical solution to this kind of problem can be extremely difficult.
In this work, an Indirect Legendre-Gauss-Lobatto Pseudospectral Method to approximate the solution of a class of switching optimal control problems is developed. In subsection 3.1.1 we refereed at this method as *Problem* $B^{\lambda N}$, however, from now on it is identified as Ind.PS for clarity. PS method using a low number of nodes suffers the difficulty to reconstruct the bang-bang thrust profile properly due to the presence of intermediate thrust values; hence it has been decided to solve this problem with an indirect method. First, based on Pontryangin's minimum principle, the first-order necessary condition of optimality is derived. This leads to the definition of a Hamiltonian Boundary Value Problem (HBVP), as described in subsection 2.1.2. Then, the state and costate functions are approximated with interpolating polynomials. Consequently, the problem is converted to the solution of a system of algebraic equations, in which the values of unknown functions at the LGL points are unknown parameters. A very good initial guess to solve the obtained system of algebraic equations for the unknown parameters is needed. Unfortunately, it is also demanding to find. Therefore, a smoothing technique is used to overcome this difficulty. In literature there are different techniques to smooth the time history of the acceleration along the trajectory, regularizing the control discontinuities. Some of them introduce smoothing functions at the level of the cost functional [94], and others apply directly at the control level, using a hyperbolic tangent approximation of the bang-bang control inputs [95].

3.2.1 Continuation Smoothing Approach

Considering the problem presented in subsection 2.1.1, here a perturbed performance index is used, that is: Maximize

$$J(\mathbf{u}(\cdot),\mathbf{x}(\cdot),t_f) = \int_{t_0}^{t_f} \left[-\|u(t)\| + \varepsilon F_u[\mathbf{u}(t)]\right] dt \tag{3.61}$$

subject to:

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t), \quad t \in [t_0, t_f] \tag{3.62}$$

The Hamiltonian function can be written as:

$$H = -\|u(t)\| + \varepsilon F_u[\mathbf{u}(t)] + \boldsymbol{\lambda}^T [f(x(t)) + g(x(t))u(t)] \tag{3.63}$$

Applying the Pontriagyn's minimum principle [4, 79, 96], the extremal control can be expressed as:

$$u^*(t) = \arg\min_u -\|u(t)\| + \varepsilon F_u[\mathbf{u}(t)] + \boldsymbol{\lambda}^T [f(x(t)) + g(x(t))u(t)] \tag{3.64}$$

Resulting in:

$$u^*(t) = u_\varepsilon(t) \frac{g(x(t))^T \boldsymbol{\lambda}(t)}{\|g(x(t))^T \boldsymbol{\lambda}(t)\|} \tag{3.65}$$

In this formulation F is a continuous function satisfying:

$$F_u \geq 0 \qquad \forall u \in [0,1] \tag{3.66}$$

ε is the continuation parameter and it is assumed to be in the interval $]0,1]$. If $F_u(u(t)) \to +\infty$ when u approaches zero or one, then F_u is called a barrier function, otherwise it is a penalty function.

Quadratic Penalty

A quadratic penalty function can be written as:

$$\varepsilon F_u[\mathbf{u}(t)] = \varepsilon \|u(t)\| (1 - \|u(t)\|) \tag{3.67}$$

This leads to a control $u_\varepsilon(t)$:

$$u_\varepsilon(t) = \begin{cases} 1 & if \quad sf \geq \varepsilon \\ \frac{1}{2} + \frac{sf}{2\varepsilon} & if \quad |sf| \leq \varepsilon \\ 0 & if \quad sf \leq -\varepsilon \end{cases} \tag{3.68}$$

where the switching function sf is defined as:

$$sf = \left\| g(x(t))^T \lambda(t) \right\| - 1 \tag{3.69}$$

Logarithmic Barrier

The logarithmic barrier function is expresses as:

$$\varepsilon F_u[\mathbf{u}(t)] = \varepsilon \log u(1-u) \tag{3.70}$$

The corresponding control is:

$$u_\varepsilon(t) = \frac{2\varepsilon}{2\varepsilon - sf + \sqrt{sf^2 + 4\varepsilon^2}} \tag{3.71}$$

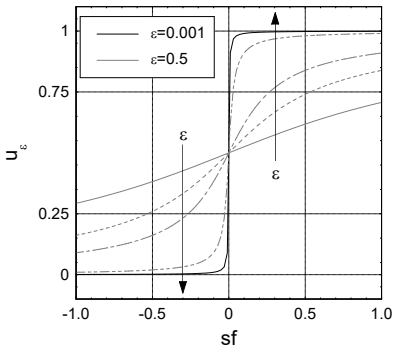
In Figure 3.2 it is possible to see the trends of the control u_ε as a function of a $sf \in [-1,1]$, and of the added costs (εF) as a function of the control $u_\varepsilon \in [0,1]$ for a logarithmic barrier function and a quadratic penalty function. The curves are drawn for different values of the continuation parameter $\varepsilon = [0.5, 0.2, 0.1, 0.01, 0.001]$.

Hyperbolic Tangent Function

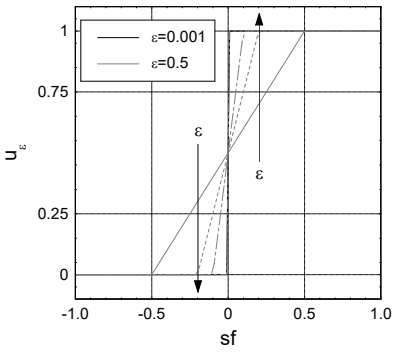
Taheri and Abdelkhalik in [97] used the hyperbolic tangent function for shaping low-thrust trajectories. However, in this work the function is introduced at the level of the control, not affecting the formulation of the necessary conditions for the indirect method. Here the control is expressed as:

$$u_\varepsilon(t) = \frac{1}{2}\left[(u_l + u_u) + (u_u - u_l) \tanh\left(\frac{sf - sf_C}{\varepsilon} \right) \right] \tag{3.72}$$

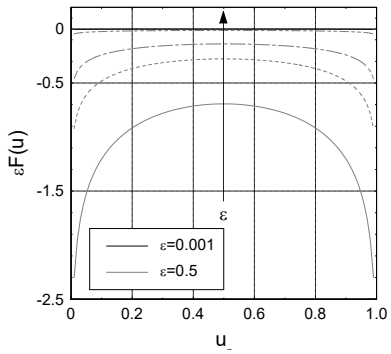
where u_l and u_u define the lower and upper bounds of the control input, and sf_C is the switching point. However, in the problems considered in this work, the switching points correspond to where the switching function crosses zero, hence $sf_C = 0$. The hyperbolic tangent acts as a smoothing filter and its behavior can be seen in Figure 3.3, as a function of the sf and for various values of ε.



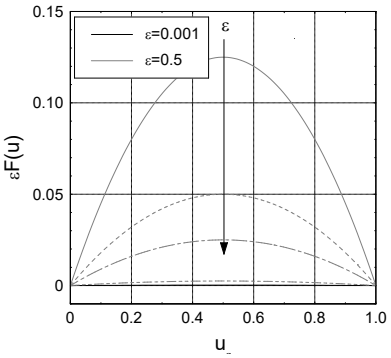
(a) Control for a logarithmic barrier function



(b) Control for a quadratic penalty function



(c) Logarithmic barrier function



(d) Quadratic penalty function

Fig. 3.2 Optimal controls (u_ε) and added costs (εF) for $\varepsilon = [0.5, 0.2, 0.1, 0.01, 0.001]$.

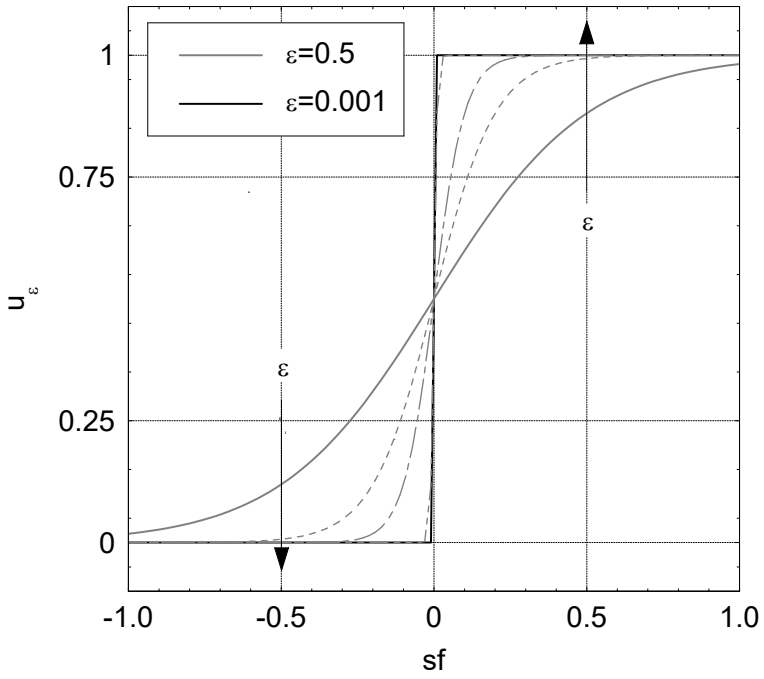


Fig. 3.3 Hyperbolic tangent control for $\varepsilon = [0.5, 0.2, 0.1, 0.01, 0.001]$.

Here it has been decided to use a logarithmic barrier function to perform the continuation smoothing approach. For a given $n > 0$, a sequence of values denoted as ε_i $(i = 1, ..., n)$ is defined with $(\varepsilon_1 > \varepsilon_1 > ... > \varepsilon_n)$. Then, the problem is solved sequentially using the solution obtained at step $(i-1)$ as an initial guess for step i. Finally, assuming that ε_n is sufficiently small, the solution of the last iteration provides a very accurate approximation for the solution of the original problem. Note that if ε_n is too small, the problem may become too stiff or infeasible, as the original one.

Since our problem is discretized at the LGL nodes, it is possible to calculate the value of u^ε at each node and use this value in the discretized differential equation. Moreover, it is possible to approximate the costate by the N^{th} degree polynomial:

$$\lambda \approx \lambda^N(\tau) = \sum_{l=0}^{N} \lambda(\tau_l)\phi_l(\tau) \tag{3.73}$$

Furthermore, the costate differential equation, already presented in Equation 2.14, can be discretized in the LGL nodes using Equation 3.73 and Equation 3.18:

$$\sum_{l=0}^{N} D_{kl}\lambda(\tau_l) + \frac{t_f - t_0}{2}\frac{\partial H}{\partial \mathbf{x}(\tau_l)}[\mathbf{x}(\tau_l), \lambda(\tau_l), \mathbf{u}(\tau_l)] \approx \mathbf{0} \tag{3.74}$$

Hence, the problem is converted in the solution of a system of algebraic equations (Equation 3.75) to find the unknowns: $\mathbf{x}(\tau_l)$, $\lambda(\tau_l)$ and $\mathbf{u}^{\varepsilon}(\tau_l)$.

$$\mathbf{F}\begin{pmatrix} \mathbf{x}(\tau_0) \\ \vdots \\ \mathbf{x}(\tau_{N+1}) \\ \lambda(\tau_0) \\ \vdots \\ \lambda(\tau_{N+1}) \\ \mathbf{u}_{\varepsilon}(\tau_0) \\ \vdots \\ \mathbf{u}_{\varepsilon}(\tau_{N+1}) \end{pmatrix} = \begin{pmatrix} \sum_{l=0}^{N} D_{0l}\mathbf{x}(\tau_l) - \frac{t_f - t_0}{2}\mathbf{f}[\mathbf{x}(\tau_0), \mathbf{u}(\tau_0)] \\ \vdots \\ \sum_{l=0}^{N} D_{N-1l}\mathbf{x}(\tau_l) - \frac{t_f - t_0}{2}\mathbf{f}[\mathbf{x}(\tau_{N-1}), \mathbf{u}(\tau_{N-1})] \\ \sum_{l=0}^{N} D_{0l}\lambda(\tau_l) - \frac{t_f - t_0}{2}\frac{dH}{dx(\tau_0)} \\ \vdots \\ \sum_{l=0}^{N} D_{N-1l}\lambda(\tau_l) - \frac{t_f - t_0}{2}\frac{dH}{dx(\tau_{N-1})} \\ \mathbf{x}(\tau_0) - \mathbf{x}(t_0) \\ \Psi\left(\mathbf{x}(\tau_f), \lambda(\tau_f), \mathbf{u}(\tau_f), \tau_f\right) \end{pmatrix} = \mathbf{0} \tag{3.75}$$

where:

$$\Psi\left(\mathbf{x}(\tau_f), \lambda(\tau_f), \mathbf{u}(\tau_f), \tau_f\right) = \begin{pmatrix} \mathbf{x}(\tau_f) - \mathbf{x_f} \\ H\left(\mathbf{x}(\tau_f), \lambda(\tau_f), \mathbf{u}(\tau_f)\right) + \frac{\partial E_e}{\partial t_f}\left(\mathbf{x}(\tau_f), \tau_f\right) \\ \frac{\partial E_e}{\partial x(\tau_f)}\left(\mathbf{x}(\tau_f), \tau_f\right) - \lambda(\tau_f) \end{pmatrix} \tag{3.76}$$

3.3 Pseudospectral Methods Comparison

The performances of three Pseudospectral methods, namely the Legendre [44], Gauss [48], and Radau [46], are presented in this section. These global methods employ orthogonal collocation based on Legendre polynomials to approximate the state, with N nodes along the trajectory. However, they differ in the choice of discretization points and collocation points where the dynamic equations are transcribed into algebraic conditions. Here, we present these discrepancies, discussing the mathematical

consequences of each point set, and performing a numerical comparison of the three methods concerning the accuracy and convergence rates of the state, control, and costate solutions.
While the Legendre method is already detailed in section 3.1, this section briefly provides a more in-depth explanation of the Gauss and Radau direct transcription methods, emphasizing the key differences between them. To ensure that the resulting NLPs have the same size and to ensure consistency in the indices among the approaches, it should be noted that the indexing utilized in this section may change from that in the previous sections. In fact, in all the approaches used for the numerical comparisons, N generally denotes the total number of discretization points (nodes), and l and k denote the $l-th$ node and $k-th$ collocation point, respectively.

3.3.1 Radau Pseudospectral Method

In this work the Radau method uses the flipped Legendre-Gauss-Radau (LGR) points. They are located in the interval $t \in (-1,1]$ and they include the final point but not the initial point. However, it is preferable for the discretization of optimal control problems to cover the full interval, including both ends. Consequently, utilizing $N-1$ flipped Radau points along with the starting point, $\tau_0 = -1$, the N discretization points are located in order to fully discretize the time window. Using a basis of Lagrange polynomials, the state approximation is generated the same way as the Legendre Pseudospectral method, where the nodes τ_l, $l = 0,\ldots,N-1$ are the initial point plus the $N-1$ LGR points:

$$\mathbf{x} \approx \mathbf{x}^N(\tau) = \sum_{l=0}^{N-1} \mathbf{x}(\tau_l)\phi_l(\tau) \tag{3.77}$$

In comparison to the Legendre Pseudospectral approach, the Radau method uses a different number of collocation points, denoted as K, as distinct from the number of discretization points, denoted as N. Here K is equal to the $N-1$ LGR points, hence it is possible to define the K collocation equations as:

$$\left[\frac{t_f - t_0}{2}\right]\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k) - \sum_{l=0}^{N-1} D_{kl}\mathbf{x}_l = \mathbf{0}, \qquad k = 1,\ldots,K \tag{3.78}$$

The differentiation matrix D is here written as:

$$\dot{\phi}_l(\tau_k) = D_{kl} = \begin{cases} \frac{\dot{g}(\tau_k)}{\tau_k - \tau_l \dot{g}(\tau_l)}, & \text{if } k \neq l \\ \frac{\ddot{g}(\tau_l)}{2\dot{g}(\tau_l)}, & \text{if } k = l \end{cases} \tag{3.79}$$

where $g(\tau_l) = (1+\tau_l)\left[L_K(\tau_l) - L_{K-1}(\tau_l)\right]$. The control is computed by using $N-1$ Lagrange polynomials, $\bar{\phi}_k(\tau)$, as the collocation equations in the Radau technique only concern the control at the Radau points.

$$\mathbf{u} \approx \mathbf{u}^N(\tau) = \sum_{l=1}^{N-1} \mathbf{u}(\tau_l)\bar{\phi}_l(\tau) \tag{3.80}$$

As can be seen in Equation 3.80, the control is not defined at the initial node. Therefore, it is possible to valuate the control at $\tau_l = -1$ using a simple extrapolation.
Also in the Radau Pseudospectral method exists a mapping between the KKT multipliers of the NLP and the costastes of the continuous-time optimal control problem [86]. In particular, it is possible to state that:

$$\lambda^N(t_k) = \frac{\tilde{\lambda}_k}{w_k}, \quad (k = 1, \ldots, K) \qquad \textit{and} \qquad \tilde{\nu} = \nu \tag{3.81}$$

A continuous-time approximation to the costate can be obtained using the discrete values of the costate as illustrated in Equation 3.82

$$\lambda \approx \lambda^N(\tau) = \sum_{l=1}^{K} \lambda(\tau_l)\bar{\phi}_l(\tau) \tag{3.82}$$

However, the NLP does not directly generate an initial costate estimation because there is no collocation at the starting site. Nonetheless, by extrapolating Equation 3.82, it is possible to approximate the initial costate.

3.3.2 Gauss Pseudospectral Method

The Gauss Pseudospectral method utilizes Legendre-Gauss (LG) points, located within the interval $\tau \in (-1,1)$, to determine the position of the nodes. These LG points do not include the first and last points of the time interval, thus, to fully discretize the interval, the N discretization points consist of $(N-2)$ interior

LG points, along with the initial point, $\tau_0 = -1$, and the final point, $\tau_f = 1$. To approximate the state, a basis of $N-1$ Lagrange interpolating polynomials is used:

$$\mathbf{x} \approx \mathbf{x}^N(\tau) = \sum_{l=0}^{N-2} \mathbf{x}(\tau_l)\bar{\phi}_l(\tau) \tag{3.83}$$

Where τ_l is the initial point plus the $N-2$ LG points. It is important to note that while the final point is included in the NLP discretization, it is not considered part of the state approximation. This approach results in a state approximation that is one order lower than the previous methods. However, this is necessary to maintain the equivalence property between the KKT conditions and HBVP conditions. Similar to the Radau Pseudospectral method, the Gauss Pseudospectral method uses a different number of collocation points, K, and discretization points, N, with $K \subset N$. In this method, the collocation points correspond to the LG points, hence $K = N-2$, whereas the discretization points include both the LG points and the initial and final points. Equation 3.84 presents the K collocation equation at the LG points:

$$\left[\frac{t_f - t_0}{2}\right] \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) - \sum_{l=0}^{N-2} D_{kl}\mathbf{x}_l = \mathbf{0}, \qquad k = 1, \ldots, K \tag{3.84}$$

To approximate the control at the $N-2$ collocation points, a basis of Lagrange polynomials, $\tilde{\phi}_l(\tau)$, of order $N-2$ is used:

$$\mathbf{u} \approx \mathbf{u}^N(\tau) = \sum_{l=1}^{N-2} \mathbf{u}(\tau_l)\tilde{\phi}_l(\tau) \tag{3.85}$$

To make sure that the final state abides by the state dynamics, a further constraint must be introduced to the discretization. A Gauss quadrature is used to ensure this and approximate the dynamics' integral throughout the full interval:

$$\mathbf{x}(\tau_f) - \mathbf{x}(\tau_0) - \frac{t_f - t_0}{2} \sum_{k=1}^{K} w_k \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) = \mathbf{0} \tag{3.86}$$

Moreover, the mapping from the KKT multipliers to the HBVP variables can be expressed as [98]:

$$\lambda^N(t_k) = \frac{\tilde{\lambda}_k}{w_k} + \tilde{\lambda}_f, \quad (k = 1, \ldots, K) \qquad \tilde{\nu} = \nu \qquad \lambda_f = \tilde{\lambda}_f \tag{3.87}$$

where $\tilde{\lambda}_k$ are the KKT multipliers associated with the collocational equations in Equation 3.84 while $\tilde{\lambda}_f$ corresponds to the quadrature constraint of Equation 3.86. The continuous-time costate approximation is created from the discrete costate using a slightly modified basis of N - 1 Lagrange interpolating polynomials, denoted as:

$$\lambda \approx \lambda^N(\tau) = \sum_{l=1}^{K+1} \lambda(\tau_l)\bar{\phi}_l(\tau) \tag{3.88}$$

where τ_l, $(i = 1,\ldots,K+1)$ are the LG points plus the final point, τ_f. It is important to note that the state and costate approximations in the Radau Pseudospectral method are based on different sets of Lagrange polynomials. The state is approximated using $(N-1)$ Lagrange polynomials based on the LG points and the initial time, while the costate is approximated using $(N-1)$ Lagrange polynomials consisting of the LG points and the final time. This difference is necessary in order to maintain the mapping between the KKT conditions and the HBVP conditions. Since no initial multiplier in the NLP problem exists, to approximate the initial costate λ_0 it is possible to use:

$$\lambda_0 = \tilde{\lambda}_f - \sum_{k=1}^{K} D_{k0}\tilde{\lambda}_k \tag{3.89}$$

3.3.3 Single State Problem

The three Pseudospectral methods formerly described are now compared in a one-dimensional optimal control problem where $y(t)$ is the state, $u(t)$ is the control, and $t_f = 5$. It can be stated as:

$$Problem\ B = \begin{cases} Minimize : J = -y(t_f) \\ \dot{y}(t) = y(t)u(t) - y(t) - u^2(t) \\ y(t_0) = 1 \end{cases} \tag{3.90}$$

The exact solution for this problem is known:

$$y^*(t) = \frac{4}{1+3e^t} \tag{3.91}$$

$$\lambda^*(t) = \frac{-1}{e^{-5}+6+9e^5}\, e^{(2\log(1+3e^t)-t)} \tag{3.92}$$

$$u^*(t) = 0.5y^*(t) \tag{3.93}$$

and it is also shown in Figure 3.4.

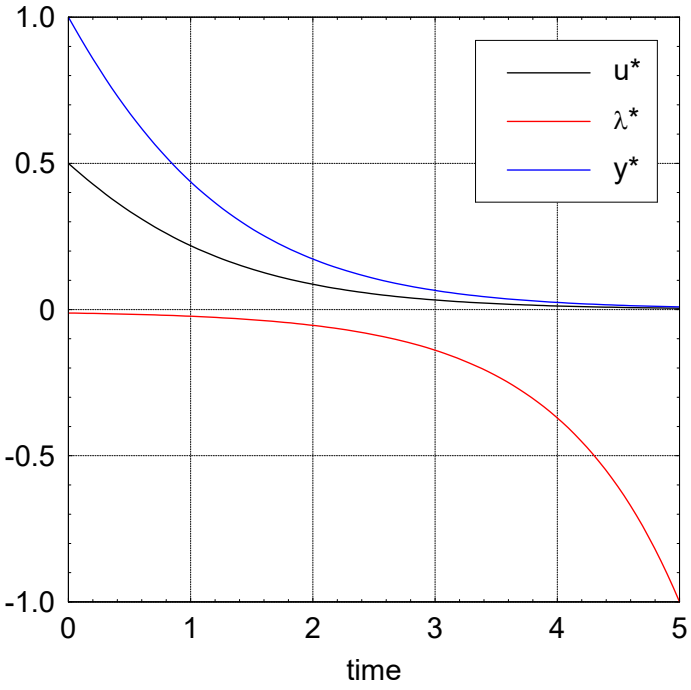
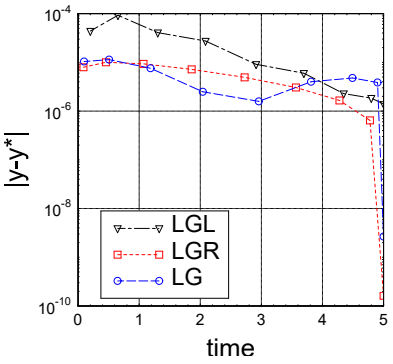
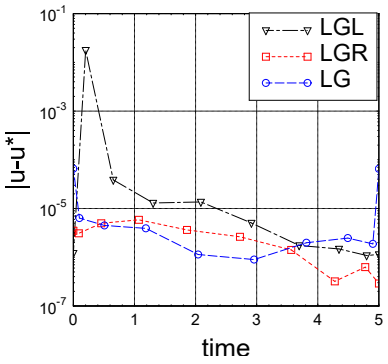


Fig. 3.4 Optimal solution for single-state problem.

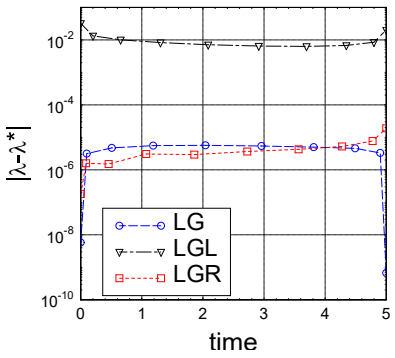
Results for the Legendre, Radau, and Gauss Pseudospectral methods have been found using 10 nodes and the Matlab function *fmincon*, with the *interior-point* algorithm and a step and constraint tolerance equal to 10^{-10}. Figure 3.5 presents the errors between the NLP and the exact solutions for the state, control, and costate in each node. In particular, the errors on the state ($|y-y^*|$) are shown in Figure 3.5a. When considering the whole interval, no method outperforms the others. On the other end, focusing on the final node, it is possible to state that both LG and LGR produce a very accurate final state and it has been claimed that the added quadrature



(a) Errors on the state



(b) Errors on the control



(c) Errors on the costate

Fig. 3.5 Errors on state, control and costate for the single-state problem using different Pseudospectral methods.

constraint in the NLP is what causes the exceptionally precise terminal state for the LG [98]. Figure 3.5b displays the errors on the control. It is worth noting that, as already explained previously, the Radau discretization lacks the initial control, while the Gauss discretization lacks the control at both the initial and final time. Several techniques can be used to compute these missing control values. One commonly used approach is to extrapolate the control approximation equations, as outlined in the literature [98]. Since the extrapolated controls are not explicit variables in the NLP, it is assumed that they may be less accurate than methods that include the boundary controls in the NLP formulation, such as the LGL. As shown in Figure 3.5b, Gauss method's boundary control estimate is the least accurate. Similarly, Radau method's initial control estimate's accuracy is comparable to that of the LGR. However, it is interesting to note that while the largest errors for Gauss and Radau occur at the boundaries, LGL has a larger error at one of the interior points. The costate comparison provided in Figure 3.5c reveals the most notable distinction between the approaches. Overall the trajectory, the LGL costate approximation is significantly less accurate than the other two methods. The approximation tends to bounce around the actual solution, which is a problem with the LGL costate approximation that is highlighted in this case and in the following section, too. It is also interesting that the highest error in the LGL costate is at the boundary. It has been seen that this behavior is due to the impossibility to satisfy contemporary the two conditions required for the KKT multipliers presented in Equation 3.51-Equation 3.54. Whereas the LGR generates an accurate starting costate (because there is no conflict), but a less accurate end costate, the LG creates very accurate boundary costates because it has no constraint conflict.

3.3.4 Orbit Transfer Problem

In this section, we compare results using the three different methods for a more complex problem. It is the orbit-raising problem of Moyer and Pinkham [99] and it has no analytical solution. However, it is a widely-studied [100, 44] classical problem to transfer a spacecraft from an initially circular orbit to the largest feasible circular orbit in a fixed time using a low-thrust engine. The problem's state comprises the spacecraft's radial distance, $r(t)$, the angular distance, $\theta(t)$, the radial velocity component, $u(t)$, and the transverse velocity component, $v(t)$. The control variable is the thrust steering angle β, measured from the local horizontal. The problem can

be stated as:
Minimize:

$$J = -r(t_f) \tag{3.94}$$

Subject to:

$$\begin{cases} \dot{r}(t) = u(t) \\ \dot{\theta}(t) = \frac{v(t)}{r(t)} \\ \dot{u}(t) = \frac{v(t)^2}{r(t)} - \frac{1}{r(t)^2} + A(t)\sin(\beta(t)) \\ \dot{v}(t) = \frac{-v(t)u(t)}{r(t)} + A(t)\cos(\beta(t)) \end{cases} \tag{3.95}$$

With initial and final constraints:

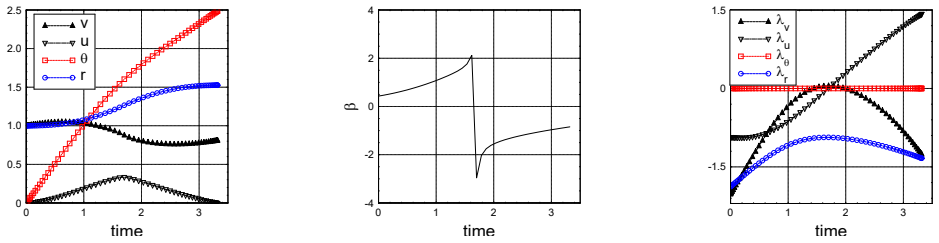
$$\begin{cases} (r(0), \theta(0)) = (1, 0) \\ (u(0), v(0)) = (0, 1) \\ (u(t_f), v(t_f)) = (0, \sqrt{1/r(t_f)}) \\ t_f = 3.32 \end{cases} \tag{3.96}$$

Here $A(t)$ is the continuous acceleration, defined as:

$$A(t) = \frac{T}{m_0 - \dot{m}t} \tag{3.97}$$

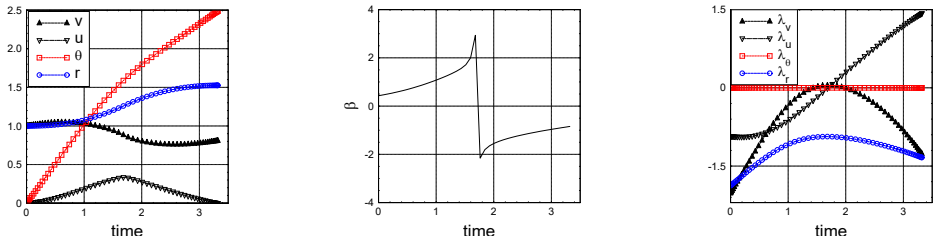
Where $T = 0.1405$, is the continuous thrust, $m_0 = 1$ is the initial mass and $\dot{m} = 0.0749$ is the constant fuel consumption. All these values are given in normalized units.
Results using Gauss, Radau, and Legendre Pseudospectral methods are presented in Figure 3.6, Figure 3.7 and Figure 3.8, respectively, using $N = 64$. As regard the states and control, the three methods yield similar qualitative results. However, upon examining Figure 3.8c, a noticeable dissimilarity is revealed in the costate obtained by the LGL when compared to the LG (Figure 3.6c) and LGR (Figure 3.7c). It is noteworthy that the LG and LGR display smooth behavior and generate an appropriate estimate of λ_θ given the problem's constraint of λ_θ^* being equal to zero, but the LGL produces oscillatory values for both λ_θ and λ_r. This difference in costate accuracy highlights a key distinction between costate estimations made with the LG and LGR and those made with the LGL. It is evident from the results of this problem that the LGL method exhibits significant inaccuracies in the boundary values of the λ_θ and λ_r. The behavior under study has been previously observed in Garg et al.'s work [59], whereas Fahroo and Ross's findings [86] indicate that the LGL technique

can accurately determine the costate without an oscillating pattern. Our results align with those of Garg et al., and we attribute this behavior to *fmincon* inability to impose the double conditions for the Lagrange multipliers in accordance with the KKT conditions, resulting in considerable errors in the λ values. However, this problem is circumvented when employing LGL discretization in an indirect method, which does not encounter this difficulty, as evidenced by the notably accurate costate results shown in Figure 3.9.



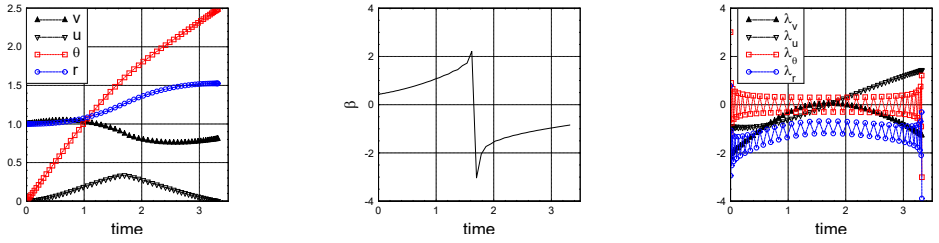
(a) States (b) Control (c) Costates

Fig. 3.6 State, control and λ for the Moyer and Pinkham problem using LG Pseudospectral method



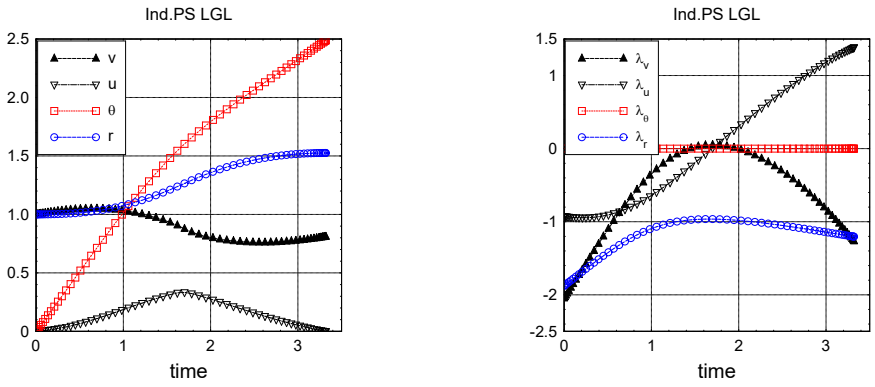
(a) States (b) Control (c) Costates

Fig. 3.7 States, control and costates for the Moyer and Pinkham problem using LGR Pseudospectral method



(a) States (b) Control (c) Costates

Fig. 3.8 States, control and costates for the Moyer and Pinkham problem using LGL Pseudospectral method



(a) States (b) Costates

Fig. 3.9 States and costates for the Moyer and Pinkham problem using Ind.PS method

3.4 Bellman Pseudospectral Method

The PS and Ind.PS methods yield discrete-time values of the state and control variables that satisfy the discretized constraints, resulting in a discrete-time feasible solution. However, to obtain a valid solution for the continuous-time optimal control problem, the discrete solution must be mapped back to the continuous time domain. Although increasing the number of nodes is a simple approach to address the discrepancy between discrete-time and continuous-time feasibility, it may not be the most efficient method available. An alternative is presented by Ross [54], who has proposed an antialiasing algorithm for low-thrust trajectory optimization that incorporates Bellman's principle of optimality. On the other hand the Bellman Pseudospectral method [55] is based on the method presented in Ref. [54] but, it can be applied to a broad range of optimal control problems, with or without the need for antialiasing and it is able to locate discontinuities in controls even when the number of PS nodes are low.

The reason for the difference in feasibility between discrete-time and continuous-time lies in the approximation and interpolation errors. To minimize the error without increasing the number of nodes, Bellman's Principle can be utilized. This principle can be expressed as follows: *If an optimal trajectory from point A to point B is given, then the trajectory from point C, which lies on the optimal trajectory, to point B is also optimal.*

Assuming we have a discrete-time feasible control $\{\mathbf{u_i}\}_{i=0}^{n}$, that is a discrete control resulting from the solution of a *Problem* B^N. It generates a continuous-time trajectory, $\mathbf{x}(t)$, we can measure the continuous-time feasibility using $\|\mathbf{x_i} - \mathbf{x}(t_i)\|$, where

$i = 1, \ldots, n$. If $\|\mathbf{x_i} - \mathbf{x}(t_i)\|$ is small for $i = 1, \ldots, n_s \ll n$, we can resolve the problem with $\mathbf{x}(t_{n_s})$ as the initial condition to reduce the measure of infeasibility. This process can be repeated to obtain a feasible solution using a small number of nodes. However, this method assumes that the system is controllable from each starting point because without this assumption, no matter how large n is, a continuous-time feasible solution cannot be generated.
This algorithm, summarized in the flowchart in Figure 3.10, can be outlined as:

1. Using a low number of nodes, solve the problem through PS or Ind.PS method. $\{\mathbf{x_i}, \mathbf{u_i}\}_{i=0}^{n}$ is the solution corresponding to discrete times $\{t_i\}_{i=0}^{n}$

2. The time interval $[t_0, t_n]$ is partitioned into N_B Bellman segments, $t_0 < t^1 < \ldots < t^{N_B} = t_n$. These segments do not need to be uniformly spaced.

3. The differential equation is propagated from t_0 to t^1 using $\mathbf{x}_0$ as the initial condition and linear interpolation for continuous-time reconstruction of the controls, $\mathbf{u^1}(t)$, $t \in [t_0, t_n]$ based on $\{\mathbf{u_i}\}_{i=0}^{n}$. That is merely the solution to the initial value problem,

$$\dot{\mathbf{x}} = f\left(\mathbf{x}, \mathbf{u}^1(t)\right), \mathbf{x}(t_0) = \mathbf{x_0} \tag{3.98}$$

   This step generates a continuous-time trajectory, $\mathbf{x^1}(t)$, $t \in [t_0, t_n]$. This propagation is here done numerically via the Runge-Kutta 4/5 method.

4. After setting $\mathbf{x_0} = \mathbf{x^1}(t^1)$ and $t_0 = t^1$ go to step 1; that is, set a new initial condition as the value of the integrated state at the end of the period $[t_0, t_1]$ and solve the problem again for n. This generates a new sequence $\{\mathbf{x_i}, \mathbf{u_i}\}_{i=0}^{n}$ corresponding to new discrete times $\{t_i\}_{i=0}^{n}$, etc.

5. The algorithm stops at the N_B^{th} sequence. The candidate optimal trajectory is given by the Bellman chain:

$$\left\{\mathbf{x}^1(t), t \in \left[t_0, t^1\right]; \mathbf{x}^2(t), t \in \left[t^1, t^2\right]; \ldots \mathbf{x}^{N_B}(t), t \in \left[t^{N_B-1}, t^{N_B}\right]\right\} := \mathbf{x_B}(t), t \in \left[t_0, t_f\right]$$

   Similarly, the corresponding controls are given by:

$$\left\{\mathbf{u}^1(t), t \in \left[t_0, t^1\right]; \mathbf{u}^2(t), t \in \left[t^1, t^2\right]; \ldots \mathbf{u}^{N_B}(t), t \in \left[t^{N_B-1}, t^{N_B}\right]\right\} := \mathbf{u_B}(t), t \in \left[t_0, t_f\right]$$

If the proposed method is used with a sufficiently large n, the difference between the discrete solution's interpolation and the continuous optimal solution becomes negligible. In such circumstances, the above algorithm serves as an illustration of Bellman's Principle of Optimality. Moreover, in this method, solutions are adopted only for a short interval, and the full solution is determined by progressively moving the initial condition towards the final conditions, reducing the traveled distance in each iteration. Hence, a continuous-time feasible solution can still be provided by a low number of nodes.

A simpler algorithm can be developed by setting the Bellman time segments' end-points exactly at the node points, $t_k \in \{t_i\}_{i=0}^{n}$, which eliminates step 3 of the algorithm. Hence, instead of propagating the differential equation, we take $\{\mathbf{x}_i, \mathbf{u}_i\}_{i=0}^{m}$, where $m \ll n$, as the solution over the Bellman segment and proceed. In this manner the integration is avoided, however, because the results from numerical propagators are typically regarded as accurate, their integration into the algorithm enhances its practical value as it automatically shows that the "Bellman-sequenced trajectory" is at least feasible from an engineering perspective.

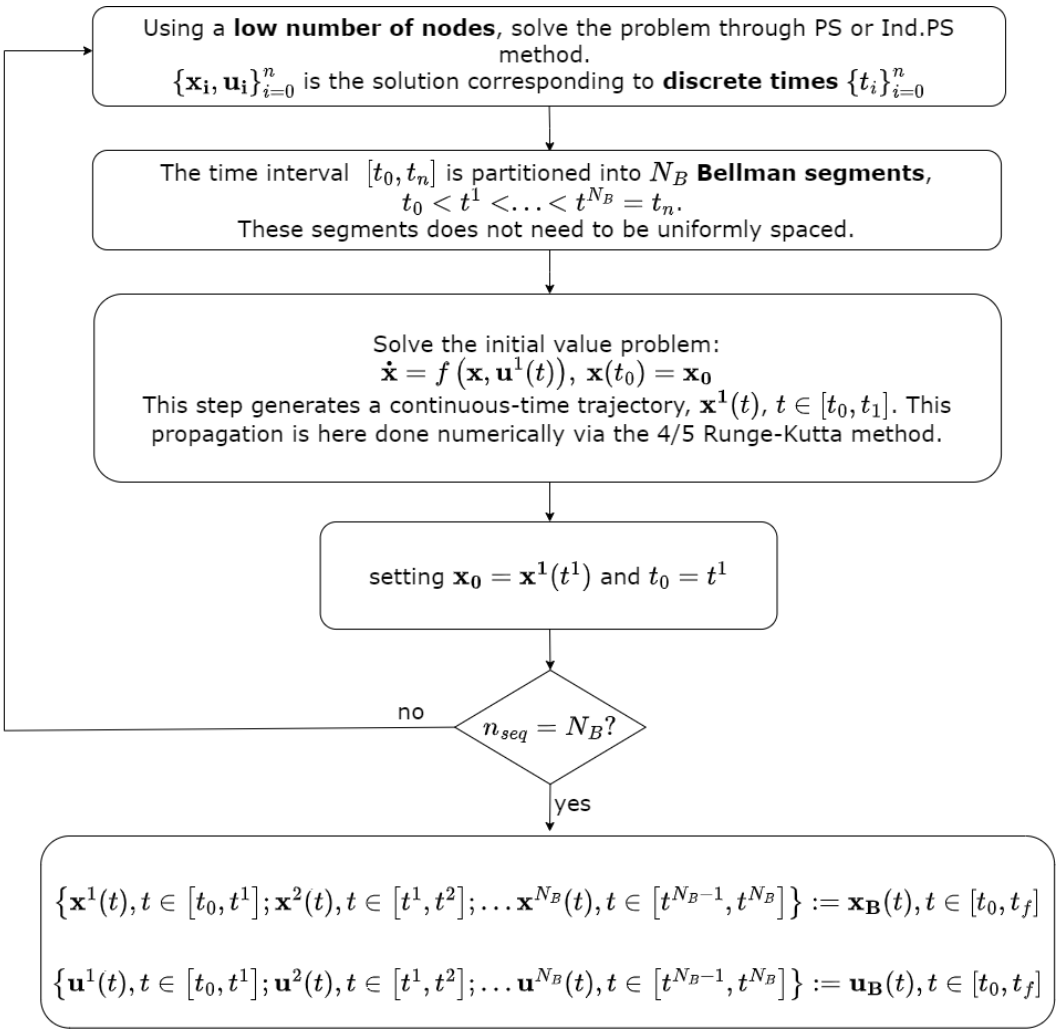


Fig. 3.10 Bellman Pseudospectral Method flowchart

Chapter 4

Results

PS methods have been widely used in the field of space trajectory optimization, but there is a lack of precise characterization regarding the performance and capabilities of these methods. Therefore, we aim to test PS on various orbital problems to understand what factors influence their performance. Specifically, it is intriguing to analyze how using few nodes can still yield satisfactory results, speeding up the convergence/solution process. We will evaluate different sets of PS collocation points, whether it is formulated directly or indirectly, and the effect of the number of nodes, with a particular focus on bang-bang control and the accurate reconstruction of the optimal thrust law. First, we explore the effect of the choice of the number of nodes for the PS and indirect PS method, both using Legendre-Gauss-Lobatto (LGL) and Legendre-Gauss-Radau (LGR). Next, we showcase the results obtained using PS and Ind.PS inserted in the Bellman method.
To evaluate the precision of the methods, we compare the results obtained with PS and Ind.PS with those obtained using an indirect method, which serves as a benchmark. Finally, we apply the same numerical methods to solve a formation reconfiguration problem.

4.1 Orbital Transfer

4.1.1 Problem Formulation

In this section, the fuel-optimal transfer between two circular orbits, representative of an Earth-Mars transfer, is considered. The formulation also deals with rendezvous transfers. All the values are dimensionless, using the Sun-Earth distance, the Earth circular velocity, and the spacecraft mass as reference values. The problem can be stated as: Determine the control function, $\mathbf{u}(\tau) = \{\beta, \sigma, \psi\}$, and the corresponding trajectory, $\mathbf{x}(\tau) = \{r, \theta, \phi, u, v, w\}$, that maximize:

$$J(\mathbf{u}(\cdot), \mathbf{x}(\cdot), t_f) = m_f \tag{4.1}$$

subject to

$$\begin{aligned}
\dot{r} &= u \\
\dot{\theta} &= \tfrac{v}{r\cos\phi} \\
\dot{\phi} &= \tfrac{w}{r} \\
\dot{u} &= \tfrac{v^2+w^2}{r} - \tfrac{1}{r^2} + \tfrac{T\beta}{m}\sin\sigma \\
\dot{v} &= -\tfrac{uv}{r} + \tfrac{vw}{r}\tan\phi + \tfrac{T\beta}{m}\cos\sigma\cos\psi \\
\dot{w} &= -\tfrac{uw}{r} - \tfrac{v^2}{r}\tan\phi + \tfrac{T\beta}{m}\cos\sigma\sin\psi \\
\dot{m} &= -\tfrac{T\beta}{c}
\end{aligned} \tag{4.2}$$

Spherical coordinates, set within an inertial reference system based on the equatorial plane, have been selected for their ability to ensure slowly changing state variables, thereby enhancing the efficiency and robustness of the NLP problem. The use of Cartesian coordinates, while the most straightforward, becomes the least advantageous option. This is primarily due to the need for a large number of discrete points to capture the rapidly changing position and velocity variables [101].
The position of the vehicle is described by the radius r, the longitude θ, and the

latitude ϕ while the speed is described by the components radial (i.e. towards Zenith) u, East v, and North w in a local reference frame. In terms of the controls, $\beta \in [0,1]$ expresses the thrust magnitude, while σ and ψ determine the thrust direction, with σ representing the in-plane thrust angle and ψ representing the out-of-plane thrust angle, measured, respectively, from the horizontal plane with positive angles upward and counterclockwise from the parallel, with positive angles to the North.
The initial and final condition for this problem are:

$$\begin{aligned}
r(0) &= 1 & & \\
\theta(0) &= 0 & r(t_f) &= r_T(t_f + t^*) \\
\phi(0) &= 0 & \theta(t_f) &= \theta_T(t_f + t^*) \\
u(0) &= 0 & \phi(t_f) &= \phi_T(t_f + t^*) \\
v(0) &= 1 & u(t_f) &= u_T(t_f + t^*) \\
w(0) &= 0 & v(t_f) &= v_T(t_f + t^*) \\
m(0) &= 1 & w(t_f) &= w_T(t_f + t^*) \\
t_0 &= 0 & &
\end{aligned} \tag{4.3}$$

where the subscript T denotes the target. The parameter t^* is a problem optimization variable for orbit transfer, while it must instead be set to 0 for rendezvous.
In this section, our focus is on a transfer to a circular target orbit with fixed radius. As a result, the final boundary conditions are reduced to:

$$r(t_f) = r_T = 1.524,\ u(t_f) = u_T = 0,\ v(t_f) = v_T = 1/\sqrt{1.524},\ w(t_f) = w_T = 0 \tag{4.4}$$

The Hamiltonian of the problem is:

$$\begin{aligned}
H = &\lambda_r u + \lambda_\theta \frac{v}{r\cos\phi} + \lambda_\phi \frac{w}{r} + \lambda_u \left[\frac{v^2 + w^2}{r} - \frac{1}{r^2} + \frac{T\beta}{m} \sin\sigma \right] + \\
&\lambda_v \left[-\frac{uv}{r} + \frac{vw}{r} \tan\phi + \frac{T\beta}{m} \cos\sigma \cos\psi \right] + \\
&\lambda_w \left[-\frac{uw}{r} - \frac{v^2}{r} \tan\phi + \frac{T\beta}{m} \cos\sigma \sin\psi \right] - \lambda_m \frac{T\beta}{c}
\end{aligned} \tag{4.5}$$

The optimal value for the thrust angles, σ and ψ can be found nullifying the partial derivatives of the Hamiltonian with respect to the thrust angles. It leads to:

$$\sin\sigma = \frac{\lambda_u}{\lambda_V} \tag{4.6}$$

$$\cos\sigma\cos\psi = \frac{\lambda_v}{\lambda_V} \tag{4.7}$$

$$\cos\sigma\sin\psi = \frac{\lambda_w}{\lambda_V} \tag{4.8}$$

where:

$$\mathbf{p} = \sqrt{\lambda_u^2 + \lambda_v^2 + \lambda_w^2} \tag{4.9}$$

is the primer vector and it is parallel to the optimal thrust direction.
H is linear with respect to β, then $dH/d\beta = 0$ does not depend on β, hence it is indeterminate. In such situations, PMP provides assistance by indicating that the control value that maximizes H should be adopted. However, the problem is meaningful only if β is bounded. In this case $0 \leq \beta \leq 1$. Additionally, the switching function $sf = dH/d\beta$ is introduced:

$$sf = \frac{\lambda_V}{m} - \frac{\lambda_m}{c} \tag{4.10}$$

and a bang-bang solution is found where the control magnitude β that maximizes the Hamiltonian is the maximum (allowed) value if the switch function is positive, the minimum if negative:

$$\beta = \begin{cases} 0 & if \quad sf < 0 \\ 1 & if \quad sf > 0 \end{cases} \tag{4.11}$$

The case of a switch function identically null over a finite time interval is not considered in Equation 4.11. In that occurrence, a singular arc arises: along the portion of the optimal trajectory where $sf = 0$, the Hamiltonian is not an explicit function of the control variables and higher-order necessary conditions are needed to determinate the optimal control. In particular, all the time derivatives of the switch function up to the least order at which the control appears explicitly have to be posed equal to zero [79]. However, singular arcs do not exist in the numerical problems considered in this thesis; thus Equation 4.11 encompasses all the possible instances.
The differential equations for the costates are provided by the Euler-Lagrange equa-

tions (Equation 2.14):

$$\begin{aligned}\dot{\lambda}_r =& \frac{1}{r^2}\left[\lambda_\theta \frac{v}{\cos\phi} + \lambda_\phi w + \lambda_u\left(-\frac{2}{r} + v^2 + w^2\right) + \lambda_v\left(-uv + vw\tan\phi\right) + \right. \\ &\left. + \lambda_w\left(-uw - v^2\tan\phi\right)\right]\end{aligned} \tag{4.12}$$

$$\dot{\lambda}_\theta = 0 \tag{4.13}$$

$$\dot{\lambda}_\phi = \frac{1}{r\cos\phi^2}\left(-\lambda_\theta v\sin\phi - \lambda_v vw + \lambda_w v^2\right) \tag{4.14}$$

$$\dot{\lambda}_u = \frac{1}{r}\left(-\lambda_r r + \lambda_v v + \lambda_w w\right) \tag{4.15}$$

$$\dot{\lambda}_v = \frac{1}{r}\left(-\lambda_\theta \frac{1}{\cos\phi} - 2\lambda_u v + \lambda_v(u - w\tan\phi) + 2\lambda_w v\tan\phi\right) \tag{4.16}$$

$$\dot{\lambda}_w = \frac{1}{r}\left(-\lambda_\phi - 2\lambda_u w - \lambda_v v\tan\phi + \lambda_w u\right) \tag{4.17}$$

$$\dot{\lambda}_m = \frac{T\beta}{m^2}\lambda_V \tag{4.18}$$

Since the final mass is maximized, the boundary condition for optimality results in:

$$\lambda_m(t_f) = 1 \tag{4.19}$$

Moreover, boundary conditions for optimality require also:

$$\nu^T \frac{\partial \mathbf{e}}{\partial t^*} = \mathbf{0} \tag{4.20}$$

where **e** are the boundary conditions expressed in Equation 4.3 and Equation 4.4. This results in the condition:

$$H_f - T\beta_f s f_f = 0 \tag{4.21}$$

In this work, three optimal phasing control problems are studied. They differ for the fixed final time and the adimensional thrust used:

- Case 1: $t_f = 5.3257$ and $T = 0.1$
- Case 2: $t_f = 20$ and $T = 0.01$

- Case 3: $t_f = 15.5$ and $T = 0.1$

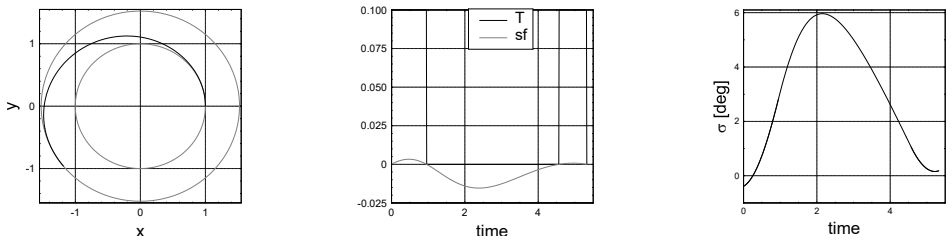
Case 1 has the final time equal to the minimum-fuel, optimal transfer time of the Hohmann-like transfer, which was obtained with an indirect procedure [102]. The other cases assume larger transfer time and/or different thrust magnitude.

4.1.2 Influence of the number of nodes

In this section, we discuss how the number of nodes used for the solution of PS and Ind.PS influences the results, what are the limitations, and how to improve the accuracy of the results. It has been decided to focus the research on LGL and LGR nodes, due to their relevance in past and present literature.
The tentative solution is easily constructed using linearly spaced vectors, from the initial values to the final ones, for r, θ, and m. Since the final mass value is unknown, it is considered to be the 80% of the initial mass. Null values of ϕ, u, w are used in the tentative solution and a constant tentative v, equal to one, is adopted. When the Ind.PS method is used, tentative values for the costates are also needed. It has arbitrarily been decided to use unit values for $\lambda_r, \lambda_u, \lambda_v$ and λ_m, while a null value is assigned to $\lambda_\theta, \lambda_\phi$ and λ_w.
The value of m_f obtained relies on both the optimality of the trajectory and the approximations introduced by PS, which lack exact integration. Subsequently, it is crucial to validate these findings using Bellman, considering not only the final mass but also the error associated with the final conditions.

Case 1

The results for Case 1, obtained using an indirect method, are displayed in Figure 4.1. The solution exhibits two thrusting arcs, occurring at the start and end of the transfer, which correspond to positive values of the switching function. Additionally, since it is a planar scenario, only the in-plane thrust angle is represented in Figure 4.1c and it indicates that the control is mostly applied in the tangential direction. The final mass achieved through the indirect method is $m_f = 0.828606$, with the consumed propellant mass (m_p) amounting to 17.14% of the total initial mass.



(a) Trajectory (b) Thrust and sf (c) In-plane thrust angle

Fig. 4.1 Trajectory, thrust, switching function, and in-plane thrust angle for Case 1 solved with the indirect method

We begin by analyzing Case 1 using Pseudospectral methods, where we keep the tentative solution fixed and vary the number of nodes. Moreover, it has been decided to conduct this study considering both the LGL and the LGR methods to be able to compare their behavior. Figure 4.2 displays how the number of nodes considered affects the absolute error between the final mass obtained using Pseudospectral methods and the optimal one (X^*), obtained using an indirect method.

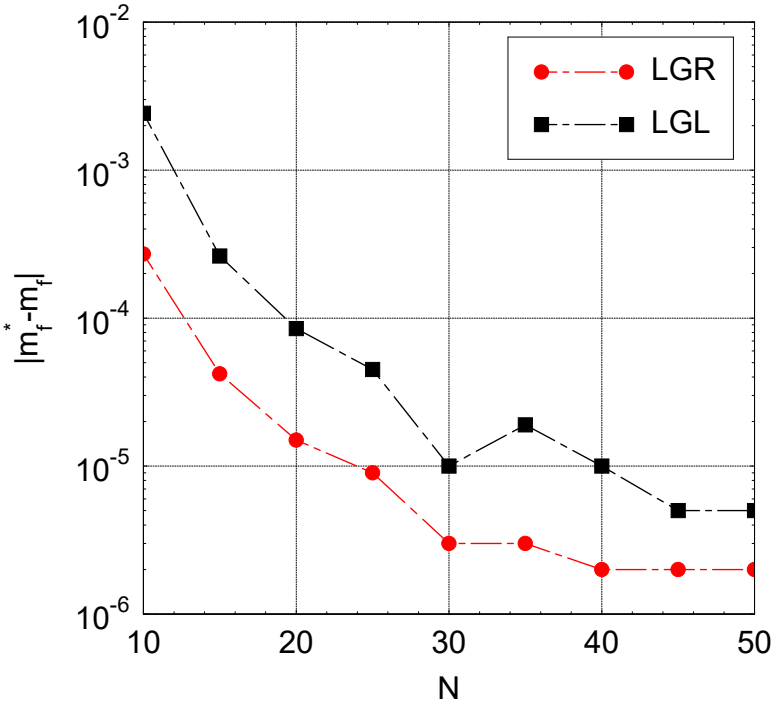
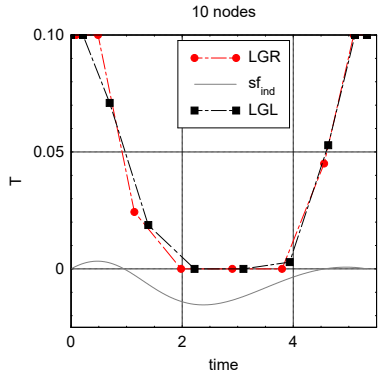
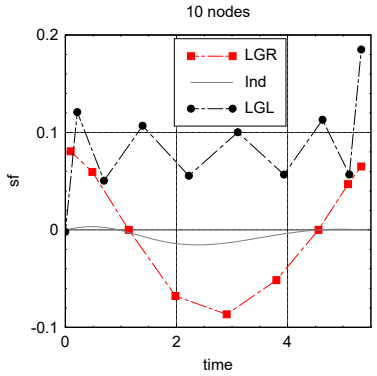


Fig. 4.2 Error on the final mass with different number of nodes for Case 1 considering LGL and LGR methods

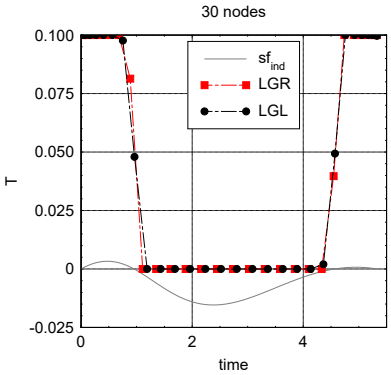


(a) Thrust reconstruction using 10 nodes

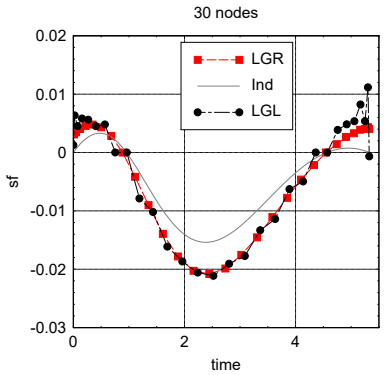


(b) sf reconstruction using 10 nodes

Fig. 4.3 Thrust and switching function reconstruction for Case 1 using 10 nodes, considering LGL and LGR method

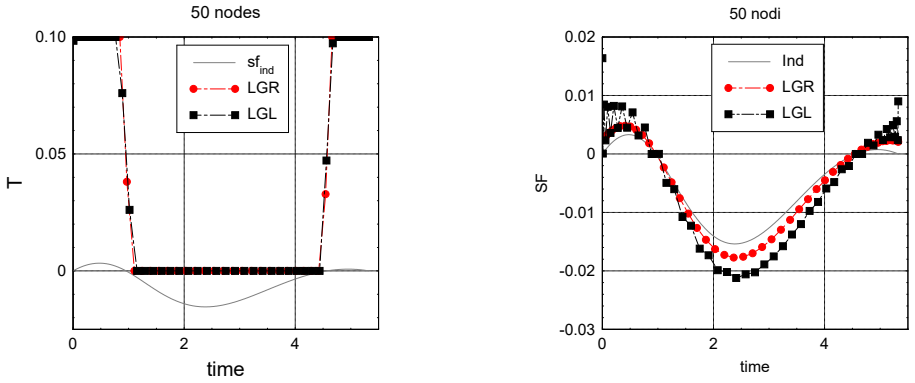


(a) Thrust reconstruction using 30 nodes



(b) sf reconstruction using 30 nodes

Fig. 4.4 Thrust and switching function reconstruction for Case 1 using 30 nodes, considering LGL and LGR method



(a) Thrust reconstruction using 50 nodes

(b) sf reconstruction using 50 nodes

Fig. 4.5 Thrust and switching function reconstruction for Case 1 using 50 nodes, considering LGL and LGR method

The accuracy of the final mass improves as the number of nodes grows. Moreover, LGR presents better results for each number of nodes used. Since the objective is to not only estimate the mass but also practically implement the obtained control law in a real application, it becomes crucial to examine the reconstruction of thrust along the trajectory, together with the calculation of the switching function using the Covector Mapping Theorem. Results obtained from simulations using 10 nodes (Figure 4.3) are not enough accurate both in terms of final mass and thrust reconstruction. When a limited number of nodes is utilized, PS is unable to accurately capture the precise switching points. As a result, non-optimal intermediate thrust values ($0 < \beta < 1$) are found, in contrast to the bang-bang control of the optimal reference solution (shown in Figure 4.3a, where $T = T_{MAX}$ when sf is positive and 0 when negative). Additionally, in Figure 4.3b, the reconstruction of sf for the LGL points seems not correct, as it remains positive even in the coasting arc, where the engine is switched off.

On the other hand, with 30 nodes both final mass and thrust profile are in agreement with the optimal solution of the indirect method (see Figure 4.4). However, it is noteworthy that LGL presents sf that oscillates at the boundaries. This behavior is similar to the one presented in Figure 3.8c. Also in this case the costates λ_r, λ_θ, and λ_u have oscillatory trend with larger range of values at the boundaries, resulting in and oscillating sf in the same region. Moreover, at the switching points sf presents values close to zero in correspondence of intermediate thrust values found by the LGL PS. However, this issue does not occur when using the LGR method. In fact, it presents fewer points where the thrust has intermediate values and a smoother sf.

When using 50 nodes, the best results are obtained (Figure 4.5), but sf for LGL still shows a peculiar behavior: two small region where sf is close to zero appear when $t \in [0.88, 1.01]$ and $t \in [4.56, 4.67]$, in correspondence of the switching region. Moreover, LGL sf retains the oscillatory trend, presented also in Figure 4.4b, at the initial and final region of the transfer, due to the oscillating KKT multipliers found with $fmincon$. Overall, it can be concluded that using a higher number of nodes yields better results, however, 30 nodes already provide a satisfying accuracy and a fast computational time, in the order of ten seconds. Furthermore, LGR seems to be more suitable than LGL for thrust reconstruction and it is also able to find smoother sf.

This analysis, which was carried out using the LGL and LGR Pseudospectral method was also conducted with an Indirect Pseudospectral method (Ind.PS) with LGL as discretization and collocation points.

The results obtained are presented in Figure 4.6. Visibly, adding more nodes has a positive effect on the results. However, the cases with 40 and 50 nodes are worth a deeper analysis.

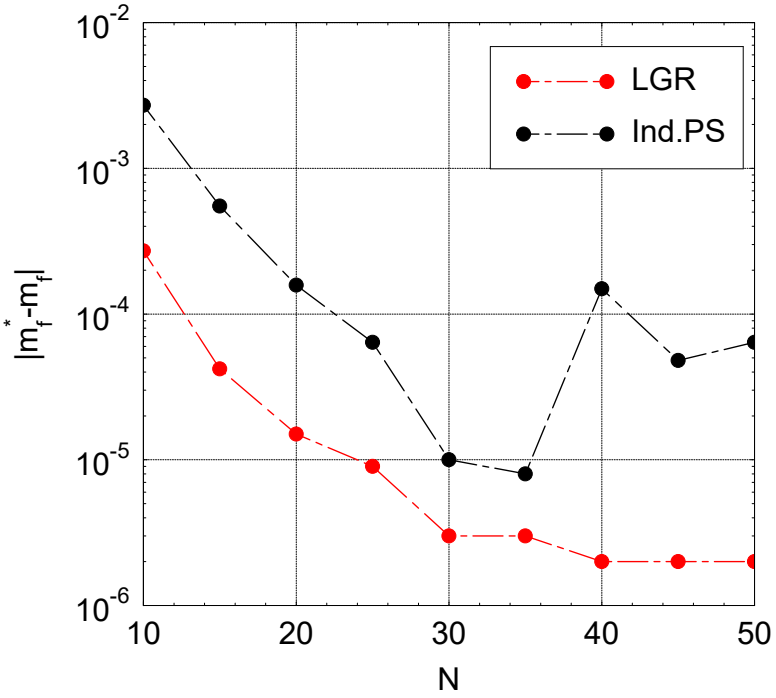
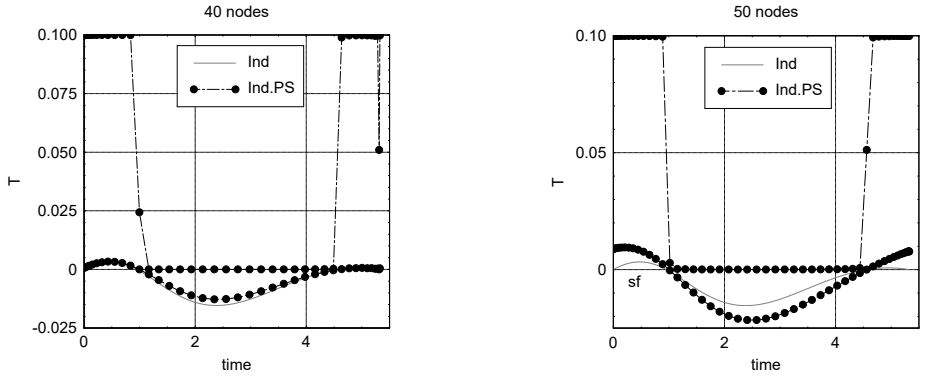


Fig. 4.6 Error on the final mass with different number of nodes for Case 1 considering Ind.PS method

Figure 4.7 displayed the thrust and switching function reconstruction for the two cases. When considering 40 nodes, there is a point in the final thrusting arc where the thrust value is intermediate. This occurs because, at that point, the switching function has a value still positive but close to zero. Consequently, when we use the smoothing technique to find the thrust at that particular point, the almost null switching function causes the thrust to have an intermediate value that worsens the final mass. In order for the thrust to be completely nullified, ε should be reduced to a magnitude of $1e^{-14}$, which significantly increases the calculation time.



(a) Thrust and sf reconstruction using 40 nodes (b) Thrust and sf reconstruction using 50 nodes

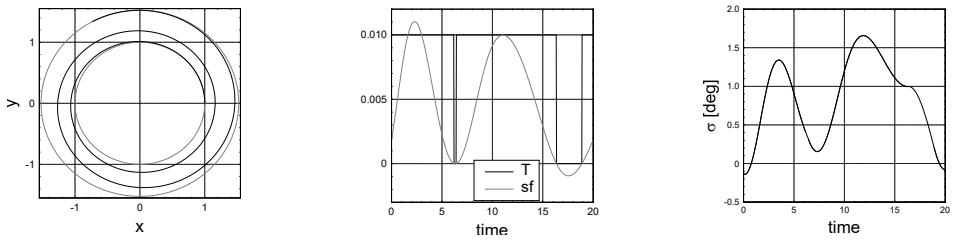
Fig. 4.7 Thrust and switching function reconstruction for Case 1 using 40 and 50 nodes, considering Ind.PS method

When using 50 nodes this issue does not occur and the results are in great agreement with the indirect solution. Moreover, comparing Figure 4.7b with Figure 4.5a, it is possible to state that using the Ind.PS avoids having many intermediate thrust values thanks to the smoothing technique that expresses the thrust value in function of sf and ε parameter, as explained in subsection 3.2.1. Furthermore, sf found through the Ind.PS does not have that oscillatory trend shown in Figure 4.5b.
These results show that both the LGR/LGL Pseudospectral method and Indirect Pseudospectral method are capable of producing good results for this case even with a relatively low number of nodes. However, it should be noted that using the Ind.PS method results in a smoother switching function and fewer instances of intermediate thrust values and therefore a solution that more closely replicates the optimal one. Nevertheless, this method requires slightly more computational time (approximately

40 seconds when using 50 nodes) due to the additional iterations required by the smoothing technique.

Case 2

Case 2 involves a longer transfer with two revolutions, as depicted in Figure 4.8, showcasing the results obtained using the indirect method. In this scenario, three thrusting arcs are observed, which contributes to a propellant consumption equivalent to 17.29% of the initial mass. Consequently, the final mass is calculated to be 0.827087. Figure 4.8c displays the in-plane thrust angle obtained through the indirect method, showcasing its optimal values within the range of $-0.14\,\text{deg}$ to $1.6\,\text{deg}$. This indicates that the thrust exerted is predominantly directed tangentially along the trajectory.



(a) Trajectory (b) Thrust and sf (c) In-plane thrust angle

Fig. 4.8 Trajectory, thrust, switching function, and in-plane thrust angle for Case 2 solved with the indirect method

When Pseudospectral methods are used to solve Case 2, the presence of the first brief coasting arc is difficult to capture. This is a predictable problem that arises when the time step length of the PS method becomes comparable to or greater than the length of the relevant thrust/coast arcs. Figure 4.9 shows the error on the final mass for LGL and LGR.

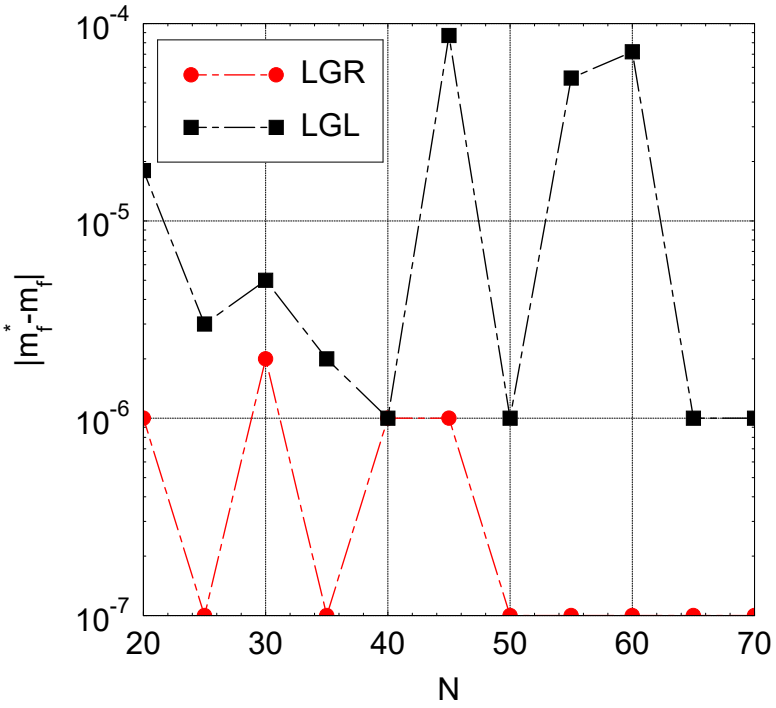
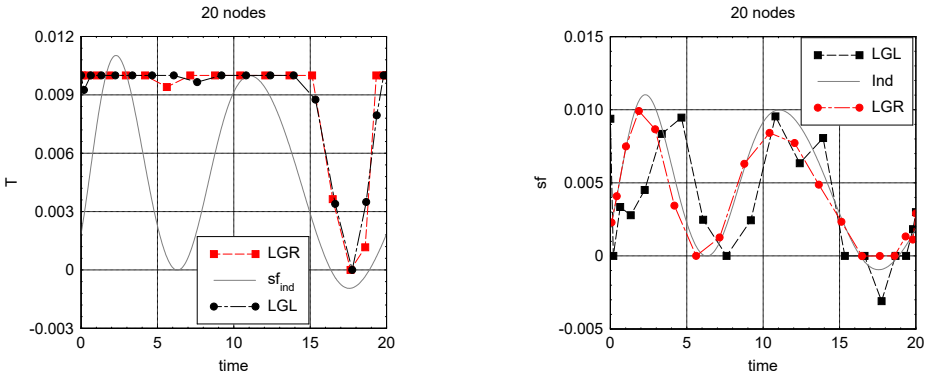


Fig. 4.9 Error on the final mass with different number of nodes for Case 2 considering LGL and LGR method

Unlike the previous case, the trend of decreasing error is not observed as the number of nodes increases beyond 40 for LGL. This is due to the method's inability to properly reconstruct the thrust in certain cases, leading to suboptimal results when the nodes are not properly spaced. However, regardless of whether the LGL or LGR method is used, the errors in the final mass never exceed 10^{-4}. The largest error occurs with LGL using 45 nodes, where the final mass is just 0.2% smaller than the one obtained through the indirect method. The results obtained for 20 nodes, Figure 4.10, indicate that this number of nodes is insufficient to correctly place the first coasting arc along the trajectory, as both LGL and LGR do not place any node in the position when the switch occurs for the indirect solution. Furthermore, the second coasting arc identified through the Pseudospectral methods is considerably smaller in comparison to the one obtained from the indirect solution. Also in this case, the transition from maximum thrust to zero thrust occurs gradually, including nodes with intermediate thrust values. Consequently, just a single point within the second coasting arc has a thrust value of $T = 0$ both for LGL and LGR. As regard the switching function, Figure 4.10b shows its trend. As already noted, LGR presents

sf in greater accordance with the one found by the indirect method, while the one related to LGL fluctuates more. Neither the LGL nor the LGR methods are capable of identifying a negative switching function within the first coasting arc. Instead, they both exhibit a point where sf approaches zero. Specifically, the LGL method yields sf value of $6.9e^{-8}$ at $t = 7.6$, while the LGR method assigns a sf value of $1e^{-6}$ at $t = 5.6$. Both of them remain positive. However, LGL found a negative sf in the second coasting arc preceded and followed by two regions of positive but close to zero sf. LGR, on the other hand, found a negative sf for $t = 16.47$ and $t = 17.64$, while the last node inside the indirect coasting arc has a positive, albeit small, sf.

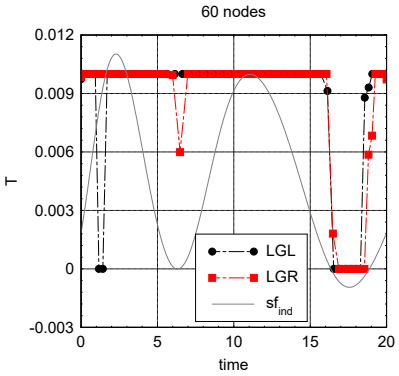


(a) Thrust reconstruction using 20 nodes

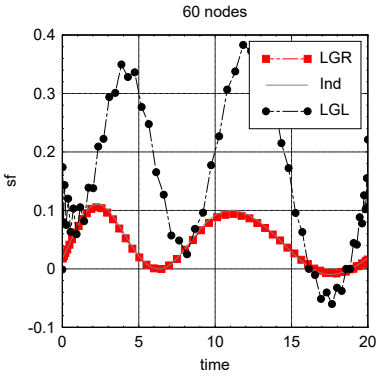
(b) sf reconstruction using 20 nodes

Fig. 4.10 Thrust and switching function reconstruction for Case 2 using 20 nodes, considering LGL and LGR method

Upon examining Figure 4.9, it becomes evident that LGL exhibits higher errors for N values of 45, 55, and 60. To clarify the reason behind this, we present the results for $N = 60$, although this behavior is consistent also across the other aforementioned cases. Figure 4.11a illustrates that LGL places the first coasting arc early in the trajectory, compared to the first region where sf_{ind} becomes negative. This leads to a smaller final mass. Moreover, the second coasting arc, found by LGL, has a shorter duration compared to the indirect solution. Figure 4.11b shows that the switching function still presents oscillatory trend at the boundaries, has already found in Case 1. Additionally, it is is not aligned with the expected coasting arc, in fact, sf stay positive where the LGL thrust is null for $t = 1.2$ and $t = 1.4$.



(a) Thrust reconstruction using 60 nodes

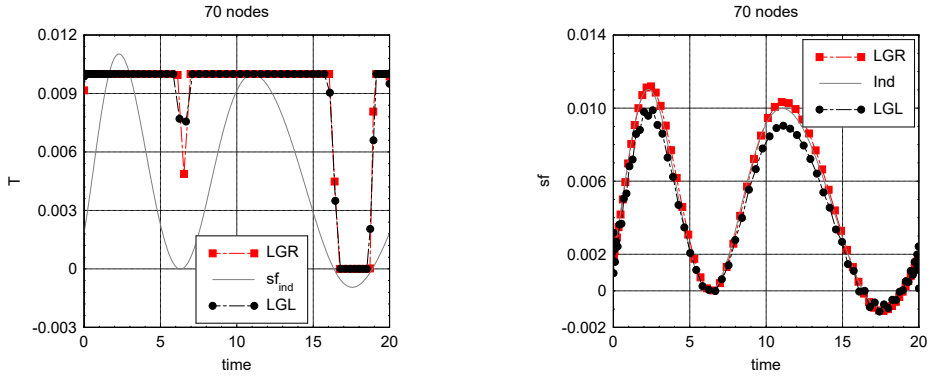


(b) sf reconstruction using 60 nodes

Fig. 4.11 Thrust and switching function reconstruction for Case 2 using 60 nodes, considering LGL and LGR method

In contrast, LGR produces results that closely match those obtained through the indirect method and it is also able to place a negative sf for $t = 6.5$. However, Figure 4.11 demonstrates that even for negative sf, the thrust found by LGR PS is not null but rather has an intermediate value. This is due to the LGR collocation, which does not place enough nodes inside the coasting arc. In fact, as we have already observed, the Pseudospectral method requires some nodes to transition between the maximum and minimum value of the thrust.

Results using 70 nodes, shown in Figure 4.12, indicate that both LGL and LGR yield satisfactory outcomes. Within the first coasting arc, the LGL method discovers a higher thrust value compared to LGR. However, even with 70 nodes, the thrust does not reach a null value. It is important to consider that the duration of the first coasting arc is $\Delta t = 0.134$, whereas the collocation points of the LGL/LGR method in that region are spaced with a larger interval of $\Delta t = 0.34$. This necessitates an extensive number of nodes to accommodate multiple nodes within this arc and it is outside the scope of this work. In contrast, the LGR method effectively reconstructs the switching function, aligning closely with the indirect solution (see Figure 4.12b). Although the oscillations in the switching function observed in the LGL method are diminished, some remnants are still present at the boundaries for this case.



(a) Thrust reconstruction using 70 nodes

(b) sf reconstruction using 70 nodes

Fig. 4.12 Thrust and switching function reconstruction for Case 2 using 70 nodes, considering LGL and LGR method

Results analyzing Case 2 using the Ind.PS technique are illustrated in Figure 4.13. The number of nodes has little impact on the error in the final mass estimation. The method successfully determines a final mass that, even in the worst-case scenario with 20 nodes, is only 0.007% higher than the mass obtained through the indirect method. Please be aware that Ind.PS does not provide exact integration, which means that there is a possibility of obtaining higher final mass values compared to the optimal solution obtained through the indirect method. Figure 4.14 presents two examples, one with 25 nodes and the other with 60 nodes. Even with 25 nodes the switching function is highly consistent with the results obtained from the indirect method. Additionally, using 60 nodes allows for accurate placement of the first coasting arc, which was not possible with LGL as demonstrated in Figure 4.11a. To reconstruct the thrust, a smoothing technique is employed, as detailed in subsection 3.2.1. However, sf within the coasting arc exhibits a small negative value that is inadequate for completely nullifying the thrust. Consequently, within this coasting phase, intermediate thrust values persist regardless of whether 25 or 60 nodes are utilized.

An additional observation to make is that the Ind.PS approach yielded slightly higher errors on the final masses in comparison to the solutions obtained with LGL/LGR PS method. Nevertheless, utilizing Ind.PS ensures improved thrust reconstruction, where the coasting arc is consistently positioned accurately along the trajectory, and a smoother switching function is also achieved.

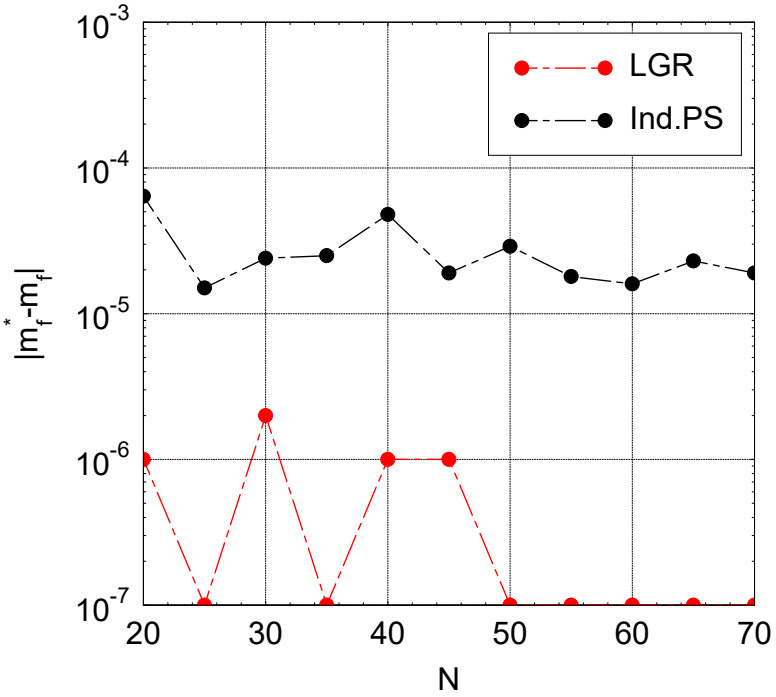
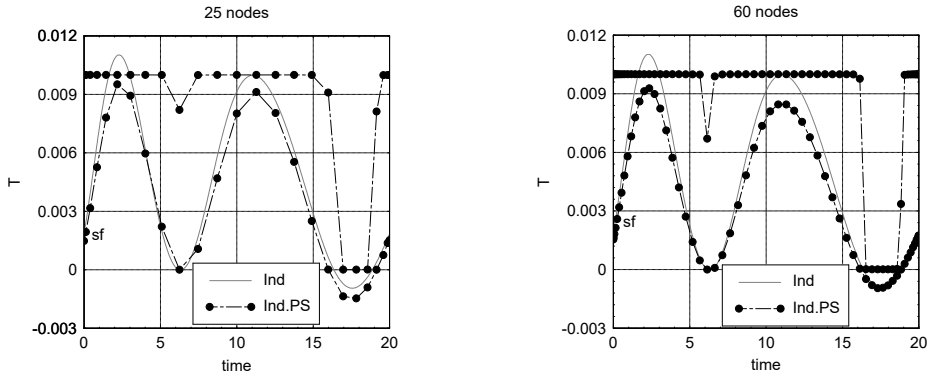


Fig. 4.13 Error on the final mass with different number of nodes for Case 2 and considering Ind.PS method

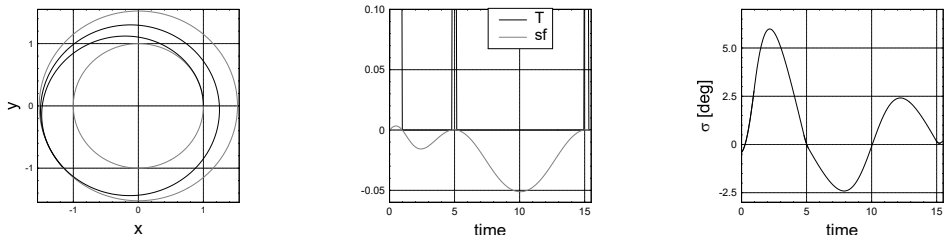


(a) Thrust and sf reconstruction using 25 nodes (b) Thrust and sf reconstruction using 60 nodes

Fig. 4.14 Thrust and switching function reconstruction for Case 1 using 25 and 60 nodes, considering Ind.PS method

Case 3

Figure 4.15 illustrates results obtained for Case 3 using the indirect method. The solution demonstrates the presence of three brief thrusting arcs, aligning with positive values of the switching function. Additionally, the in-plane thrust angle is depicted in Figure 4.15c, indicating that, similar to previous cases, the control predominantly applies in the tangential direction (highlighted regions), where σ presents smaller values. The final mass attained through the indirect method is $m_f = 0.828618$, which is only slightly greater than Case 1 by 0.0014%. Furthermore, the consumed propellant mass (m_p) accounts for approximately 17.14% of the total initial mass.



(a) Trajectory (b) Thrust and sf (c) In-plane thrust angle

Fig. 4.15 Trajectory, thrust, switching function, and in-plane thrust angle for Case 3 solved with the indirect method

The issue observed in the initial coasting arc of Case 2 could also manifest in a scenario featuring three small thrusting arcs akin to those in Case 3. The challenge of reconstructing the thrust trend arises from the brief duration of the thrusting arcs ($\Delta t_1 = 0.956$, $\Delta t_2 = 0.362$, $\Delta t_3 = 0.335$), necessitating closer node placement than these intervals for precision. However, LGL and LGR points have the characteristic that they are more dense at the beginning and at the end of the trajectory and more sparse in the middle. This leads to difficulty to reconstruct the second thrust arc since the number of nodes inside this arc is low.

The error in the final mass for various N is depicted in Figure 4.16. Increasing the number of nodes does not necessarily lead to more precise outcomes, and there could be multiple explanations for this phenomenon. For instance, in the case of LGL, utilizing 25 nodes results in suboptimal outcomes because the method identifies four thrusting arcs instead of three, with thrust values that are below the maximum, as can be seen in Figure 4.17a.

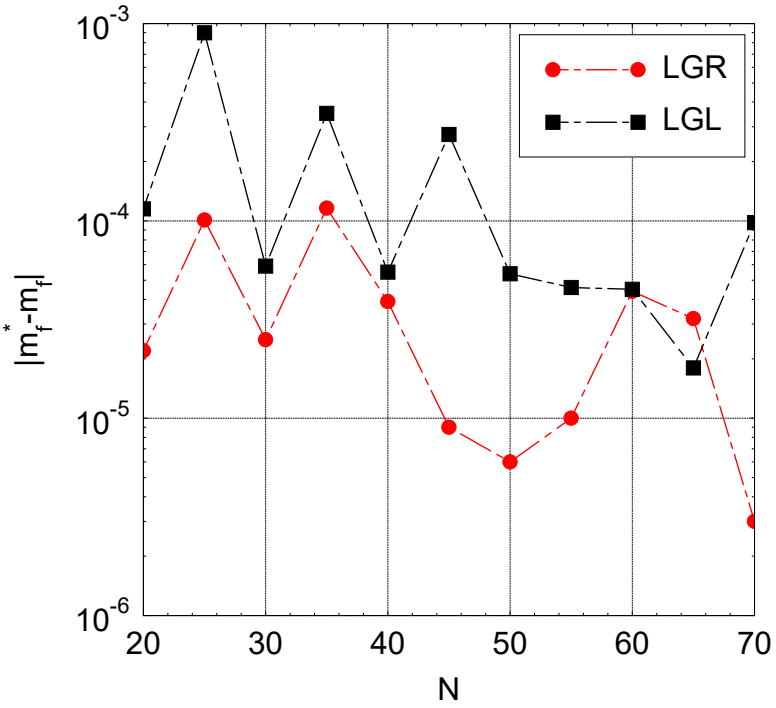
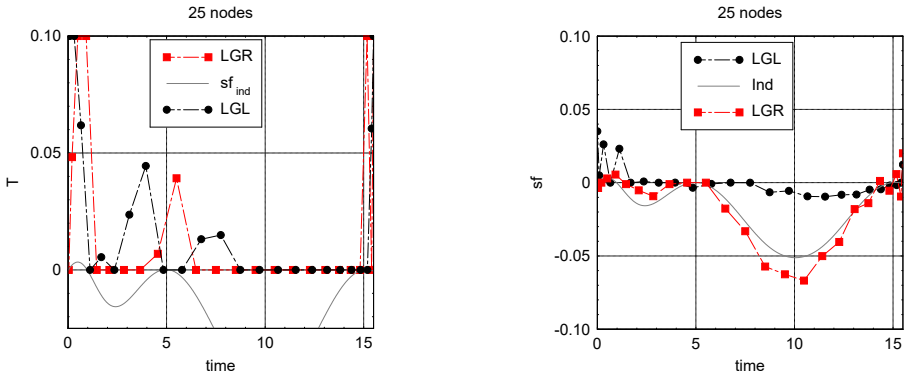


Fig. 4.16 Error on the final mass with different number of nodes for Case 3 and considering LGL and LGR method

LGR detects three thrusting arcs, however, also in this case, the second firing does not feature maximum thrust and in the last thrusting arc there is also an instantaneous thrust switch. sf, as depicted in Figure 4.17b, exhibits oscillations at the boundary when employing LGL, displaying similar behavior observed in previous cases.

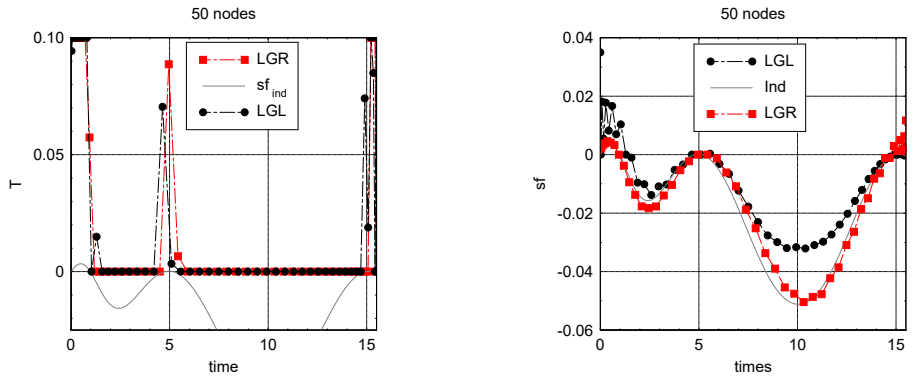


(a) Thrust reconstruction using 25 nodes

(b) sf reconstruction using 25 nodes

Fig. 4.17 Thrust and switching function reconstruction for Case 3 using 25 nodes, considering LGL and LGR method

However, in this scenario, also when utilizing LGR, sf experiences abrupt switches between positive and negative values in the final portion of the transfer. For these reasons, it is possible to state that 25 nodes are not sufficient to have accurate and reliable results.
When considering 50 nodes, LGL exhibits a distinct behavior as shown in Figure 4.18. At $t = 1.5$, the thrust assumes a $T = 0.015$ value, adding a burn that however corresponds to a negative switching function. Additionally, in the final thrusting arc, there is an intermediate thrust value between zero and the maximum for 3 nodes, corresponding to a switching function with a positive value, albeit very close to zero. In this case as well, oscillations of sf can be observed at the boundaries of the problem. Conversely, LGR outperforms LGL in terms of both functional and thrust reconstruction. LGR achieves a higher thrust value in the second firing arc, and the final arc is depicted without the switches observed in LGL. sf closely aligns with the indirect solution. However, it exhibits an oscillating trend in the latter part of the trajectory, maintaining a positive value even for $t \in [15.3, 15.5]$, where sf_{ind} becomes negative.



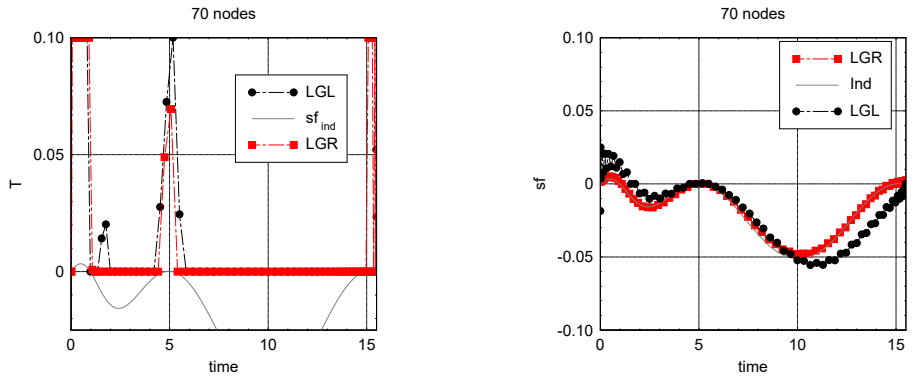
(a) Thrust reconstruction using 50 nodes

(b) sf reconstruction using 50 nodes

Fig. 4.18 Thrust and switching function reconstruction for Case 3 using 50 nodes, considering LGL and LGR method

Results for 70 nodes are depicted in Figure 4.19. In this case, LGR performs well, but there is a minor imperfection whereby the thrust is null at three nodes in the final firing arc, corresponding to a negative switching function. Conversely, LGL exhibits the same behavior as in the previous results at $t = 1.5$, with a small additional arc with $T = 0.02$. Additionally, it can identify the second thrusting arc, but since it

lasts longer than the arc found using the indirect method, the third thrusting arc disappears. Also in this case, the switching function presents greats fluctuations at the beginning of the transfer, still remaining always positive. In the final nodes, instead, it is negative, in accordance with the null thrust found by the PS.



(a) Thrust reconstruction using 70 nodes

(b) *s* reconstruction using 70 nodes

Fig. 4.19 Thrust and switching function reconstruction for Case 3 using 70 nodes, considering LGL and LGR method

As regards the Ind.PS method, it is impossible to find good results. This could be due to the fact that in this case, we have a system of algebraic equations that must be satisfied and if the nodes are not placed correctly, convergence is not achieved. Hence the Ind.PS is not an adequate method to solve problems with very short burning arcs. This consideration on the thrusting arcs can be done just when the arc distribution is known and it cannot be done in advance. Moreover, due to the nodes distribution, this method seems to be more suited for problems with longer thrusting arcs, so that one can be sure to reconstruct the thrust trend correctly.

4.1.3 Comparative analysis between Pseudospectral Methods and Trapezoidal Collocation Method

This section undertakes a comparative analysis between Pseudospectral Methods and the Trapezoidal Collocation (TC) Method. TC transforms a continuous-time trajectory optimization problem into a NLP through the application of trapezoidal quadrature, also known as the trapezoid rule for integration. As a result, each continuous element of the problem is converted into a discrete approximation [13, 103]. In this work results using the TC Method have been obtained using *OptimTraj*, a matlab

library designed for solving continuous-time single-phase trajectory optimization problems [104].

The study is conducted for Case 1 and Case 2 shown in previous sections, with a focus on analyzing the computational time and the accuracy of the results. Table 4.1 compares LGR, LGL, Ind.PS, and TC methods in terms of computational time required to achieve the solution in Case 1, for different number of nodes. Notably, as the number of nodes increases, LGR emerges as the most efficient method for problem resolution; on the other hand, TC is much slower and has computational times comparable to or even greater than Ind.PS (where iterations are required by the smoothing technique). As regard the control solution, TC effectively places the two thrusting arcs, in line with the results obtained in subsection 4.1.2. Considering Figure 4.2 and Figure 4.6, it is evident that LGR stands out as the method able to find cost function values in great accordance with the results of the indirect optimal solution. This observation holds true even when considering TC, as TC exhibits a behavior similar to that of LGR. Specifically, when $N > 20$ both TC and LGR present errors below 0.001%, then for $N > 40$ LGR is able to provide smaller errors ($2e^{-6}$) compared to $4e^{-6}$ found by TC.

Table 4.1 Computational time for Case 1 using LGL, LGR, Ind.PS, and trapezoidal collocational (TC) method.

Computational time [s]	LGR	LGL	Ind.PS	TC
$N = 10$	0.5833	0.5376	3.3848	1.5046
$N = 20$	3.4026	2.8493	9.8410	6.3858
$N = 30$	6.5833	6.3400	21.0159	15.7023
$N = 40$	14.7675	13.3935	38.7836	43.0655
$N = 50$	23.1853	26.4501	35.8338	89.9135

In Table 4.2, the computational time for solving Case 2 is presented. Similar to Case 1, Ind.PS and TC exhibit a lower convergence rate, leading to an increase in computational time. Additionally, for $N \leq 30$, TC shows the highest errors $|m_f^* - m_f|$ among the methods. However, for $N > 30$, the error decreases and remains below 0.002%. Despite this, an examination of the control results from the TC Method in Figure 4.20 reveals a notable issue: the method fails to accurately position the first brief coasting arc and is unable to find a $T = 0$ for $t \in [16.333, 18.867]$. These issues persist for all the numbers of nodes considered. In conclusione PS methods

seem superior to TC, both in terms of speed of calculation and, when short arcs are involved, of solution accuracy.

Table 4.2 Computational time for Case 2 using LGL, LGR, Ind.PS, and trapezoidal collocational (TC) method.

Computational time [s]	LGR	LGL	Ind.PS	TC
$N = 20$	5.1764	4.8180	9.6195	10.9729
$N = 30$	12.97933	17.3703	19.6586	21.2379
$N = 40$	16.3906	22.8714	33.8045	58.6003
$N = 50$	26.6686	49.0545	57.0068	94.0222
$N = 60$	44.1000	59.4665	64.9420	183.1912
$N = 70$	83.9097	87.3692	93.6745	260.6818

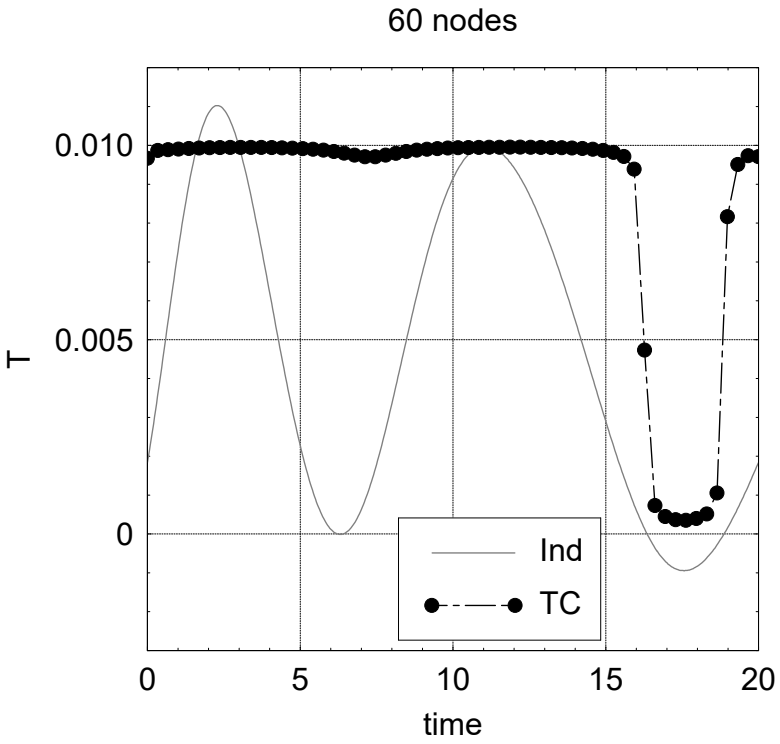


Fig. 4.20 Thrust reconstruction for Case 2 using 60 nodes, considering TC method

4.1.4 Bellman Pseudospectral Method

PS and Ind.PS methods provide solutions for the state and control variables at discrete time intervals. To obtain a valid solution to the original continuous-time optimal control problem, it is necessary to map the discrete solution to the continuous-time domain. This is accomplished using the Bellman Pseudospectral method, in conjunction with LGL, LGR, and Ind.PS methods, for Case 1 and 2 that have been already presented. As previously explained in section 3.4, the control variables must be interpolated from the discrete results during the propagation of the differential equations in a segment. It has been seen, for Case 2 specifically, that the Pseudospectral method finds intermediate thrust values. In the case where the thrust cannot be modulated, we can utilize the switching function to reconstruct the bang-bang solution. In fact, in line with Pontryagin's Maximum Principle (PMP):

$$T = \begin{cases} T_{MAX} & if \quad sf > 0 \\ 0 & if \quad sf < 0 \end{cases} \tag{4.22}$$

In subsection 4.1.2 we have observed that the switching function presented a smoother trend when employing LGR nodes. Based on these findings, it has been determined that implementing the control reconstruction using the switching function based on the discrete solution of the LGR Pseudospectral method is the most effective approach, compared to LGL.

Case 1

In light of the outcomes achieved through the LGL and LGR Pseudospectral approaches as well as the Ind.PS method in Case 1, we are now presenting the solution attained using the Bellman approach to build a continuous solution. Four Bellmann segments, equally space throughout the transfer, are used here.
The thrust applied during integration is depicted in Figure 4.21, which has been obtained through interpolation of the discrete control solutions using LGL, LGR, and Ind.PS methods for solving the problem. All the methods are able to reconstruct the two thrusting arcs. The first one has a longer duration compared to the indirect solution, because the transition from a maximum thrust to a null one occurs gradually for all the methods. In addition, Table 4.3 illustrates the errors between the adimensional final integrated states and the target conditions, as well as the adimensional final

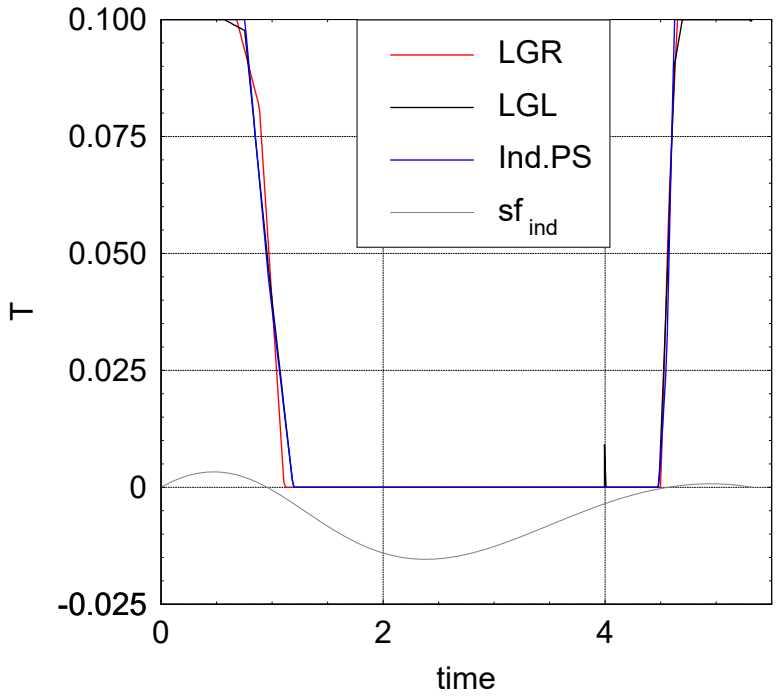


Fig. 4.21 Thrust reconstruction for Case 1 using 30 nodes and $N_B = 4$, considering Bellman algorithm with LGR, LGL and Ind.PS. The thrust was reconstructed interpolating the discrete control

mass. The small errors observed across all the methods provide compelling evidence of their capability to attain the desired target conditions. Among the methods, LGL yields the smallest error in reaching the final radius, while Ind.PS exhibits the largest discrepancies from the target conditions.

However, Ind.PS has the highest final mass, whereas LGL and LGR achieve final masses that are 0.04% and 0.03% lower (even though very close) than the indirect solution, respectively. It should be noted that, as mentioned earlier, the thrust is interpolated from discrete outcomes in this analysis, meaning that intermediate values between maximum and zero thrust can occur during integration. Furthermore, the LGL solution presents an instantaneous thrust value of 0.01 at $t = 4$ (see Figure 4.21). This can occur during the transition between two segments, where the continuation of control is not applied, and the control law in the subsequent segment may differ from the one determined previously.

Table 4.3 Errors for the final states for Case 1 using Bellman + PS with $N_B = 4$. The thrust was reconstructed interpolating the discrete control.

$\Delta x = \lvert x_T - x_B \rvert$	LGL	LGR	Ind.PS
Δr_f	$3.2e^{-5}$	$4.6e^{-5}$	$5.7e^{-5}$
Δu_f	$8.2e^{-5}$	$5.2e^{-5}$	$1.3e^{-4}$
Δv_f	$9.1e^{-5}$	$9.7e^{-6}$	$2.2e^{-4}$
m_f	0.828246	0.828314	0.828638

Alternatively, if a purely bang-bang control is desired, the knowledge of sf can be exploited to assess the thrust, as previously described. The results of this type of reconstruction, using the Bellman algorithm in conjunction with LGR and Ind.PS, can be observed in Figure 4.22. Here, we have illustrated both the thrust and the switching function. Visibly, the switching function is not continuous: each segment is, in fact, solved independently from the previous one, except for state continuity.

ùà

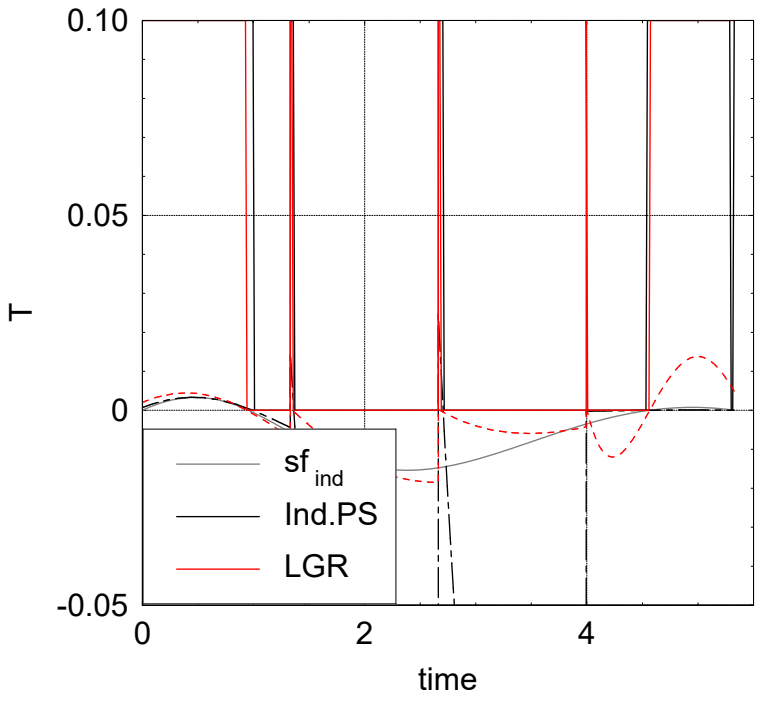


Fig. 4.22 Thrust reconstruction for Case 1 using 30 nodes and $N_B = 4$, considering Bellman algorithm with LGR and Ind.PS method. The thrust was reconstructed using sf

Therefore, the adjoint variables (both reconstructed/mapped for LGR PS methods or obtained by the Ind.PS solution) may be discontinuous at segment junctures. Additionally, when utilizing LGR, there are three extra thrusting arcs, while Ind.PS requires just two additional burns for $t \in [1.33, 1.26]$ and $t \in [2.66, 2.7]$. To gain a better understanding of the necessity for these extra arcs, Figure 4.23 presents the evolution of periapsis (r_p) and apoapsis (r_a) with time.

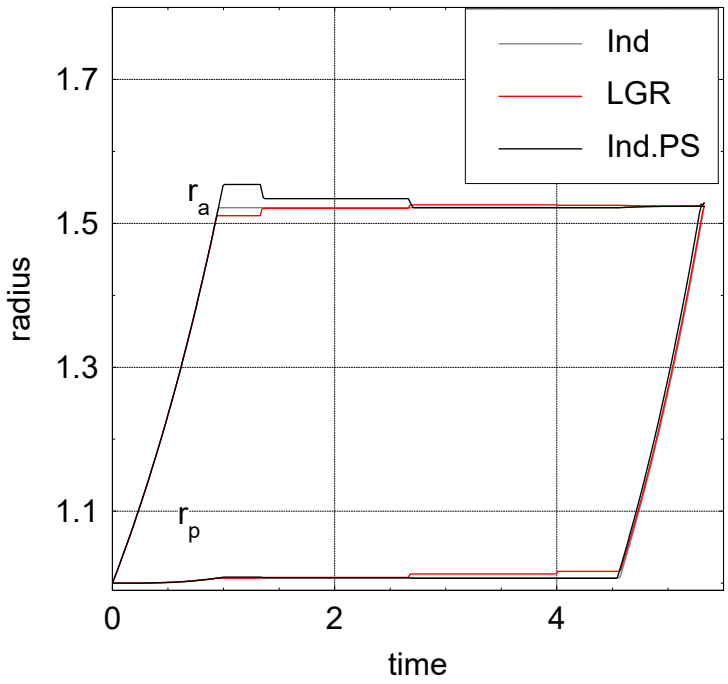


Fig. 4.23 Periapsis and apoapsis evolution for Case 1 using 30 nodes and $N_B = 4$, considering Bellman algorithm with LGR and Ind.PS method. The thrust was reconstructed using sf

When considering Bellman+LGR solution, after the initial burning arc, the apoapsis has not attained the optimum value (as represented by the indirect solution). Therefore, an additional arc is required to increase it as quickly as possible since it is more convenient to modify it when closer to the periapsis. The other two burns are instead used to slightly increase the periapsis. Conversely, when employing Ind.PS, the apoapsis is already greater after the first burning arc, since it has a longer duration; hence, two small braking arcs are incorporated. As a consequence, the final mass is reduced, but the decrease amounts to only 0.28% for LGR and 1.3%

for Ind.PS. Table 4.4 illustrates the errors between the adimensional final integrated state and the target state.

Table 4.4 Errors for the final states and final mass for Case 1 using Bellman + LGR and Ind.PS with $N_B = 4$. The thrust was reconstructed using sf.

$\Delta x = \lvert x_T - x_B \rvert$	Ind.PS	LGR
Δr_f	$2.5e^{-4}$	$3.6e^{-4}$
Δu_f	$2.2e^{-4}$	$9.7e^{-5}$
Δv_f	$6.7e^{-4}$	$7.6e^{-4}$
m_f	0.817545	0.826287

Remarkably, the Ind.PS method demonstrates a propensity to attain the final radius with more accuracy, despite the greater propellant requirement to achieve this outcome. However, interpolating the control form the discrete solution yields smaller errors on the final state, as can be seen comparing Table 4.3 and Table 4.4.
The Bellman method offers two distinct ways of utilization. The first approach occurs when the evolution of states after integration does not deviate significantly from the discrete solution. In this scenario, the Bellman solution closely resembles the discrete solution, effectively transitioning from a discrete-time solution to a continuous-time solution. This is exemplified by the solution obtained through the interpolation of discrete controls (Figure 4.21). Conversely, when the switching function is employed, a strict bang-bang control is imposed. This control law differs from the one used to achieve the PS solution. PS methods have in fact solutions with intermediate thrust values and cannot attain a strict bang-bang control history. As a result, the evolution of integrated states may differ from the discrete solution. In such cases, the Bellman algorithm starts a new optimization process from the updated initial states, allowing for the discovery of a new control law that may differ from the solution obtained in previous iterations.

Case 2

When analyzing Case 2, different segment numbers (N_B =4, 6, and 8) are used in the Bellman method. Table 4.5 and Table 4.6 show the error between the obtained final states (adimensional) and the target one, together with the final mass resulted

after the integration. The errors exhibit a decreasing trend as the number of segments increases for LGL, LGR, and Ind.PS, with all the considered cases yielding final masses greater than those obtained through the indirect method (but do not exactly attain the required final orbit). However, two exceptions, where the obtained final masses are lower than the indirect one, are observed for LGL with $N_B = 4$ and LGR with $N_B = 8$. In the former case, the final mass is 0.827024, requiring only an additional 0.04% of propellant compared to the optimal required propellant mass. As for LGR with $N_B = 8$, the percentage of additional propellant decreases to just 0.01% when compared to m_p^*.

Table 4.5 Errors for the final states for Case 2 using Bellman + PS with $N_B = 4, 6, 8$. The thrust was reconstructed interpolating the discrete control.

	$N_B = 4$		$N_B = 6$		$N_B = 8$	
$\Delta x = \lvert x_T - x_B \rvert$	LGL	LGR	LGL	LGR	LGL	LGR
Δr_f	$1.5e^{-4}$	$5.3e^{-4}$	$2.4e^{-5}$	$1.7e^{-5}$	$2.6e^{-7}$	$2.6e^{-5}$
Δu_f	$1.5e^{-4}$	$3.4e^{-5}$	$7.5e^{-6}$	$1.7e^{-6}$	$3.4e^{-6}$	$3.5e^{-5}$
Δv_f	$4.9e^{-6}$	$3.7e^{-4}$	$2.2e^{-5}$	$4.1e^{-5}$	$2.2e^{-6}$	$9.2e^{-6}$
m_f	0.827024	0.827160	0.827094	0.827112	0.827088	0.827067

Table 4.6 Errors for the final states for Case 2 using Bellman + Ind.PS with $N_B = 4, 6, 8$. The thrust was reconstructed interpolating the discrete control.

$\Delta x = \lvert x_T - x_B \rvert$	$N_B = 4$	$N_B = 6$	$N_B = 8$
Δr_f	$1.2e^{-3}$	$8.1e^{-5}$	$2.5e^{-5}$
Δu_f	$4e^{-4}$	$1.5e^{-5}$	$7.3e^{-5}$
Δv_f	$6.3e^{-4}$	$1.6e^{-4}$	$2.2e^{-5}$
m_f	0.827099	0.827256	0.827113

Figure 4.24 displays the thrust behavior achieved with Ind.PS, LGL, and LGR in the Bellman algorithm using 8 segments and shows that, within the first coasting arc of the indirect optimal solution, Ind.PS detects the lowest thrust value. However, as already shown in Figure 4.11a and Figure 4.14b, the PS methods, both in its direct or indirect formulation, are unable to find that first coasting arc and this leads to an integration that considers intermediate thrust value in this transfer region.

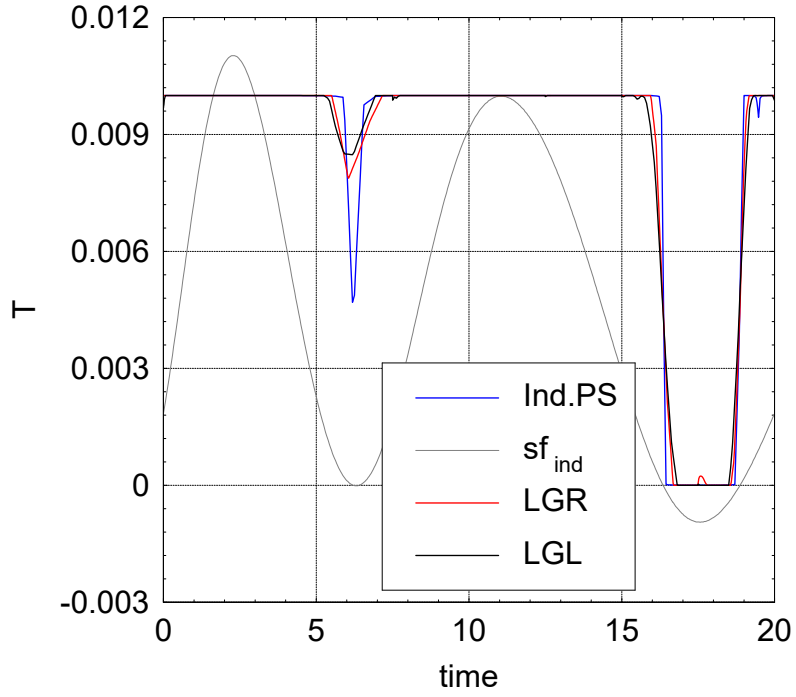


Fig. 4.24 Thrust reconstruction for Case 2 using 40 nodes and $N_B = 8$, considering Bellman algorithm with LGL and LGR method. The thrust was reconstructed interpolating the discrete control.

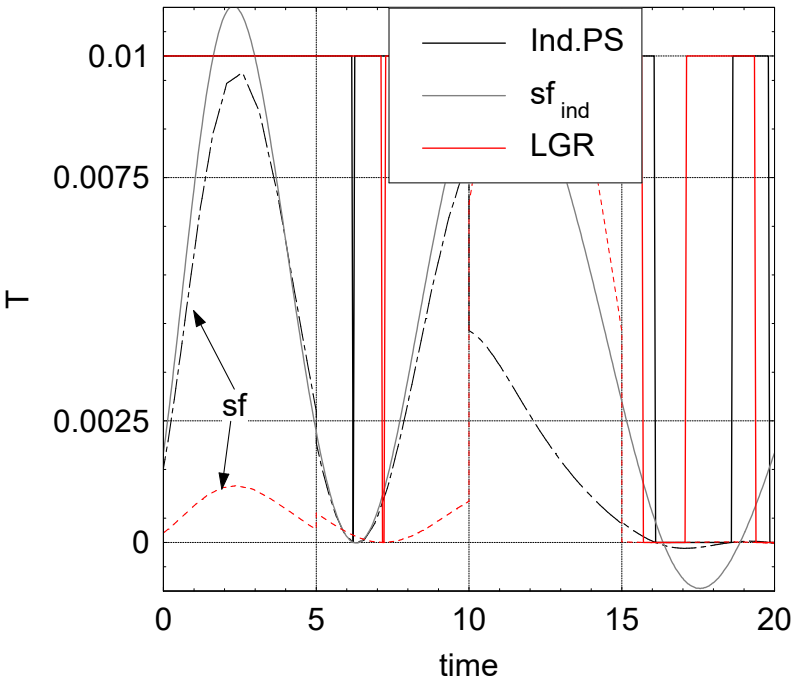
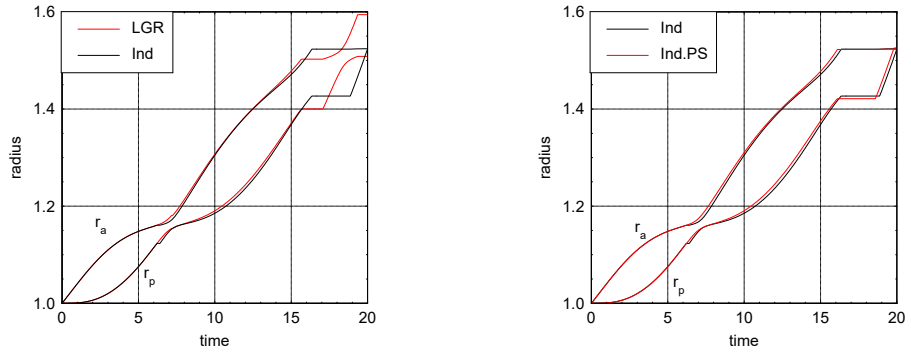


Fig. 4.25 Thrust reconstruction for Case 2 using 40 nodes and $N_B = 4$, considering Bellman algorithm with Ind.PS and LGR method. The thrust was reconstructed using sf.

To identify the initial coasting arc, where the thrust is null, one approach is to leverage the information regarding the value of sf. As explained in subsection 4.1.4, if sf at a given moment is negative, the use of zero thrust becomes optimal and $T = 0$ is imposed during the integration process. This specific technique yields the results presented in Figure 4.25. Bellman+Ind.PS approach successfully determines the placement of the first coasting arc, exhibiting a remarkable alignment with the solution obtained through the indirect method. Nevertheless, a slight variation arises in the positioning of the second coasting arc, which occurs slightly earlier in the transfer. Furthermore, this method exhibits engine switch-off during the latter part of the transfer, specifically in the time interval $t \in [19.8, 20]$. This contrasts with the behavior of the indirect solution, which continues to thrust throughout this period. However, when utilizing the Bellman+LGR approach, the position of the coasting arc differs compared to the indirect solution. In this case, the first coasting arc occurs within the time interval $t \in [7.1, 7.2]$, which is later compared to the timing in the indirect solution (refer to Figure 4.25). Furthermore, the second coasting arc takes place earlier and has a shorter duration. These variations in the control strategy impact the evolution of periapsis and apoapsis, as depicted in Figure 4.26. While the Ind.PS method closely follows the behavior of the indirect solution, the Bellman+LGR approach exhibits a distinct pattern in the latter part of the transfer. This is evident from the significantly different values of the final periapsis and apoapsis compared to the target conditions. These discrepancies are also reflected in Table 4.7, where the errors in the final state for LGR prove to be greater than those obtained with Ind.PS. Regarding the final mass, the Bellman+LGR approach yields m_f that is 0.7% lower than the optimal final mass (m_f^*). Conversely, the final mass obtained by Bellman+Ind.PS aligns closely with the solution derived from the indirect method. One should also note the larger error on the required velocity of the Bellman+LGR approach, and the higher accuracy when Ind.PS is employed.

Table 4.7 Errors for the final states and final mass for Case 2 using Bellman + Ind.PS $N_B = 4, 6, 8$. The thrust was reconstructed interpolating the discrete control.

$\Delta x = \lvert x_T - x_B \rvert$	LGR	Ind.PS
Δr_f	$6.5e^{-3}$	$1e^{-3}$
Δu_f	$1.9e^{-2}$	$3.3e^{-4}$
Δv_f	$3.4e^{-3}$	$6.2e^{-4}$
m_f	0.821296	0.827020



(a) r_p and r_a evolution with Bellman and LGR (b) r_p and r_a evolution with Bellman and Ind.PS

Fig. 4.26 Periapsis and apoapsis evolution for Case 2 using 40 nodes and $N_B = 4$, considering Bellman algorithm with LGR and Ind.PS method. The thrust was reconstructed using sf

In this particular case, employing the switching function to reconstruct the thrust profile leads to a control law that, after integration, fails to precisely satisfy the final target conditions when using Bellman+LGR (Table 4.7). However, results improve when Bellman is used in conjunction to Ind.PS.

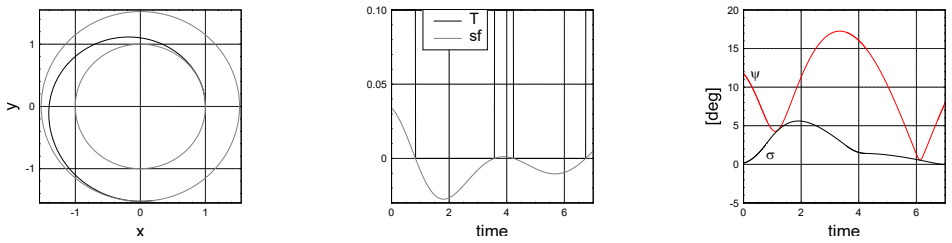
4.1.5 3D Case

So far, only two-dimensional scenarios have been taken into account. However, in this particular section, a three-dimensional case will be examined. In this scenario, a modification of inclination is also necessary. Initial and final orbits have orbital parameters:

$$a(0) = 1,\ e(0) = 0,\ i(0) = 0,\ \omega(0) = 0,\ t_p(0) = 0,\ \Omega(0) = 0,\ m(0) = 1 \quad (4.23)$$

$$a(t_f) = 1.524,\ e(t_f) = 0,\ i(t_f) = 2\,\text{deg},\ \omega(t_f) = 0,\ t_p(t_f) = 0,\ \Omega(t_f) = 0 \quad (4.24)$$

The required transfer must be accomplished in $t = 7$ with a thrust of $T = 0.1$ and $c = 1$.



(a) Trajectory (b) Thrust and sf (c) Control angles

Fig. 4.27 Trajectory, thrust, switching function, and control angles for Case 3D solved with the indirect method

The trajectory and control results obtained from solving the problem using the indirect method are presented in Figure 4.27. The solution showcases three regions of thrust application. During the first two arcs, the thrust is primarily exerted in the tangential and normal directions, as illustrated in Figure 4.27c. The angles σ and ψ are depicted throughout the transfer, with the highlighted regions indicating the firing arcs. The majority of the plane change maneuver occurs within the second arc, where ψ reaches its highest value. Regarding the final thrusting event, it is executed with a nearly zero σ angle and a ψ value ranging between 5.7 deg and 7.9 deg. This transfer requires a propellant mass equal to the 17.5% of the initial total mass and at the end of the maneuvers $m_f = 0.824977$.

First, we perform an analysis utilizing 40 nodes for the LGL, LGR, and Ind.PS techniques. The resultant thrust is depicted in Figure 4.28. All three methods accurately position the three firing arcs along the trajectory. LGL found a longer second firing arc where it places just one node with maximum thrust value while the other three have intermediate values. This results in a slightly different trend of the inclination, that starts to change earlier in the transfer, see Figure 4.29. All the methods ensure that the orbital parameters follow the same evolution as the optimal indirect solution, as evident from Figure 4.29, satisfying the required final conditions. Furthermore, in terms of maximizing the final mass, all the methods yield a close approximation of the exact value. Specifically, Ind.PS obtains a final mass (m_f) that is only 0.04% lower than the optimal solution obtained via the indirect method, and this discrepancy reduces further to 0.005% and 0.002% with LGL and LGR, respectively.

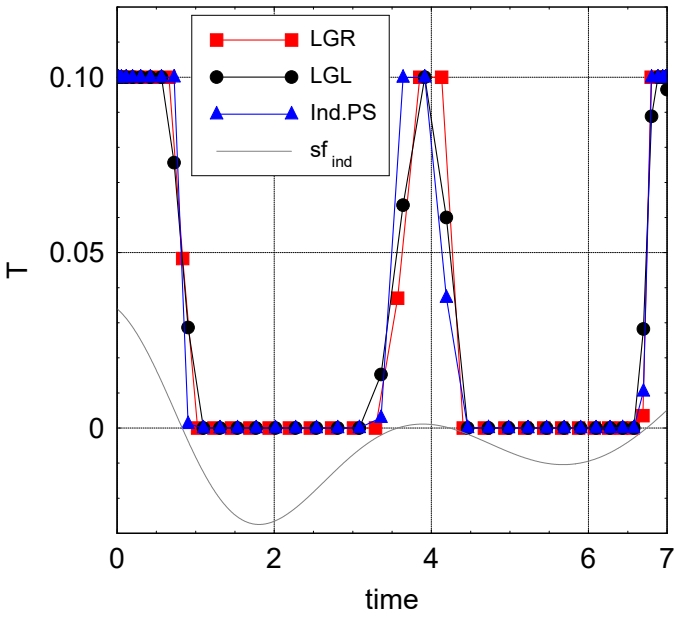


Fig. 4.28 Thrust reconstruction for 3D Case using LGR, LGL, and Ind.PS method with 40 nodes

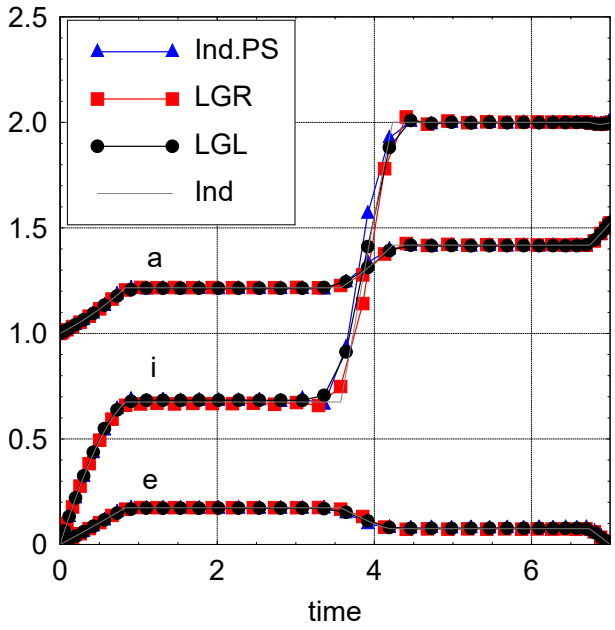


Fig. 4.29 Semi-major axis, eccentricity and inclination for 3D Case using LGR, LGL, and Ind.PS method with 40 nodes

The transition from a discrete-time solution to a continuous-time solution, uses Bellman method along with LGL, LGR, and Ind.PS. In this problem, the system is divided into four segments, and the control values are interpolated from the discrete solution during integration.

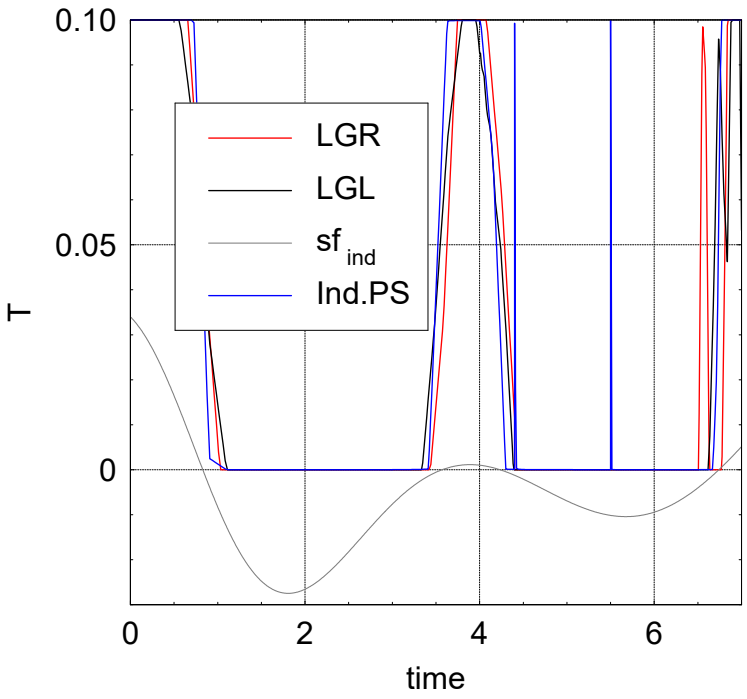


Fig. 4.30 Thrust reconstruction for 3D Case using Bellman algorithm with LGR, LGL, and Ind.PS with 40 nodes and $N_B = 4$. The thrust was reconstructed interpolating the discrete control.

Figure 4.30 shows that all three methods can identify the first two thrusting arcs, but due to the control interpolation, the arcs are longer and have an intermediate region where the thrust gradually decreases from its maximum value to zero. As a result, integrated states deviate from the solutions obtained by Pseudospectral methods. After the second burn, all methods behave differently from the optimal indirect solution, as seen in Figure 4.30. Ind.PS identifies an additional brief burn at $t = 4.4$. LGL finds a thrust that oscillates with values below T_{max} at the beginning of the third arc and then uses T_{max} after $t = 6.87$. On the other hand, LGR finds an additional firing arc with $T < T_{max}$ for $t \in [6, 6.27]$ to increase inclination and periapsis, and decrease eccentricity, as shown in Figure 4.31. The use of interpolated

controls results in a smooth variation of the orbital parameters.

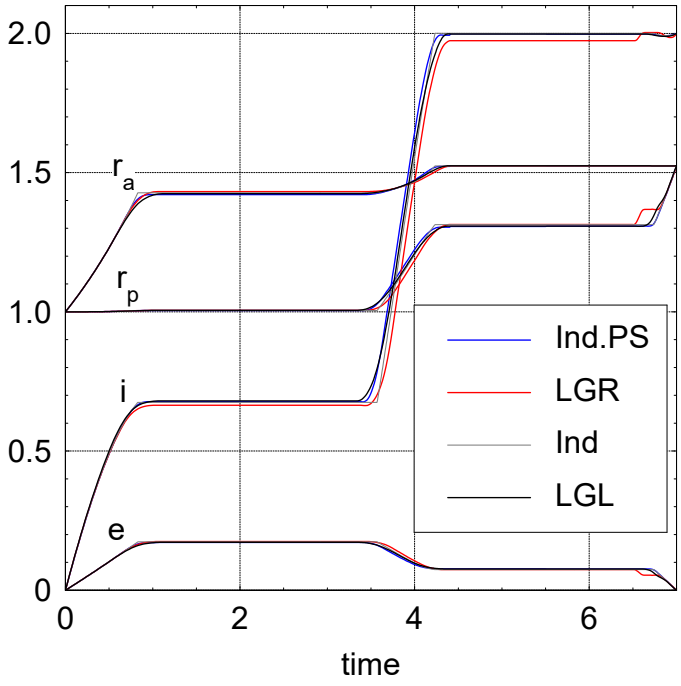


Fig. 4.31 Periapsis, apoapsis, inclination and eccentricity evolution for the 3D Case using 40 nodes and $N_B = 4$, considering Bellman algorithm with LGL, LGR, and Ind.PS. The thrust was reconstructed interpolating the discrete control.

The errors between the final and target states, as well as the final mass obtained after integration, are presented in Table 4.8. In general, LGL method exhibits the largest errors, which can be attributed to the intermediate values in the final thrusting arcs, as previously explained. However, it manages to achieve a final mass that is only 0.05% lower than m_f^*. On the other hand, the LGR method demonstrates the greatest error in the final velocity v_f and requires approximately 0.3% more propellant. Despite the inclusion of an additional brief firing arc, the Ind.PS method achieves a final mass that is merely 0.04% lower than the optimal solution. Moreover, it proves to be the most effective method in avoiding abrupt switches in the thrust reconstruction.

Table 4.8 Errors for the final states and final mass for the 3D Case using Bellman + PS/ Ind.PS with $N_B = 4$. Results are shown for the two techniques: thrust reconstructed interpolating the discrete control (T interpolated) and thrust reconstructed using the switching function value (sf).

	T interpolated			sf	
$\Delta x = \lvert x_T - x_B \rvert$	LGL	LGR	Ind.PS	LGR	Ind.PS
Δr_f	$2.1e^{-5}$	$7.5e^{-5}$	$5.4e^{-5}$	$1.9e^{-3}$	$1.2e^{-3}$
Δu_f	$1.2e^{-4}$	$9.2e^{-5}$	$4e^{-5}$	$1.9e^{-3}$	$1.1e^{-3}$
Δv_f	$1.5e^{-4}$	$4.4e^{-4}$	$9.6e^{-5}$	$1.7e^{-3}$	$1.9e^{-4}$
Δw_f	$2.3e^{-4}$	$3.5e^{-5}$	$2.6e^{-4}$	$2.4e^{-4}$	$2.1e^{-3}$
m_f	0.824546	0.824433	0.824627	0.821759	0.823821

When the thrust value is determined based on the switching function value, the outcomes differ as shown in Figure 4.32. In this scenario, the Bellman+LGR approach yields a solution similar to the indirect method, but with an additional brief thrusting interval between $t = 5.25$ and $t = 5.3$. This thrusting arc is primarily employed to increase the inclination since, after the second burn, the spacecraft is unable to achieve the desired final inclination of 2 deg (Figure 4.33). Bellman+Ind.PS discovers a distinct thrust profile with some notable differences. It includes a slightly lengthier initial thrusting phase, followed by a second burn executed later during the transfer, specifically between $t = 3.96$ and $t = 4.58$. In this case, between $t = 5.25$ and $t = 5.38$, the engine thrust is employed to decrease the inclination and eccentricity while increasing the periapsis, as depicted in Figure 4.33. Then, for $t \in [6.66, 6.8]$ the spacecraft performs the final thrusting arc. The discrepancies between the target and final states, along with the final mass, are presented in Table 4.8. Ind.PS achieves marginally more accurate results and exhibits a final mass of 0.823821, which demonstrates a 0.14% reduction compared to the optimal solution obtained through the indirect method ($m_f^* = 0.824977$). Although this approach deviates from the indirect method's outcome, it solves the problem without significantly increasing propellant consumption. Bellman+LGR produces results that align more closely with the indirect method, yielding a final mass 0.39% smaller than the indirect method's solution. Also in this case, the errors on the final states are greater compared to the one obtained using the interpolated discrete control.

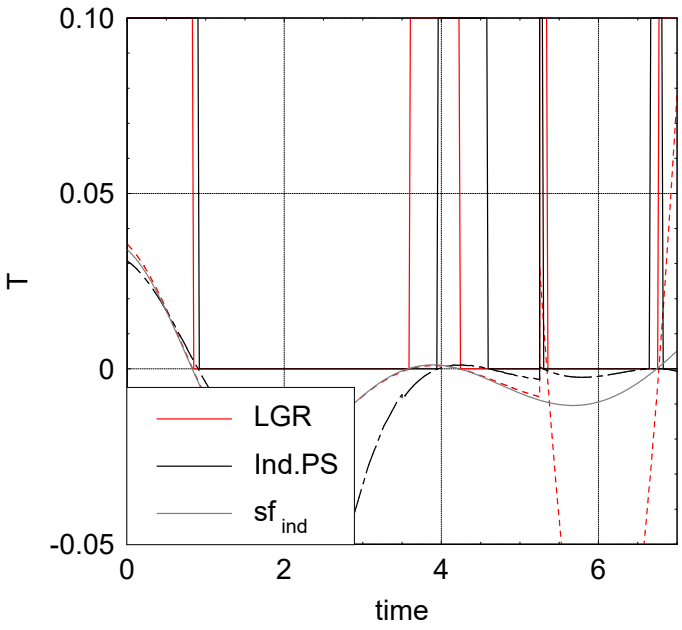


Fig. 4.32 Thrust reconstruction for 3D Case using Bellman algorithm with LGR and Ind.PS with 40 nodes and $N_B = 4$. The thrust was reconstructed using sf.

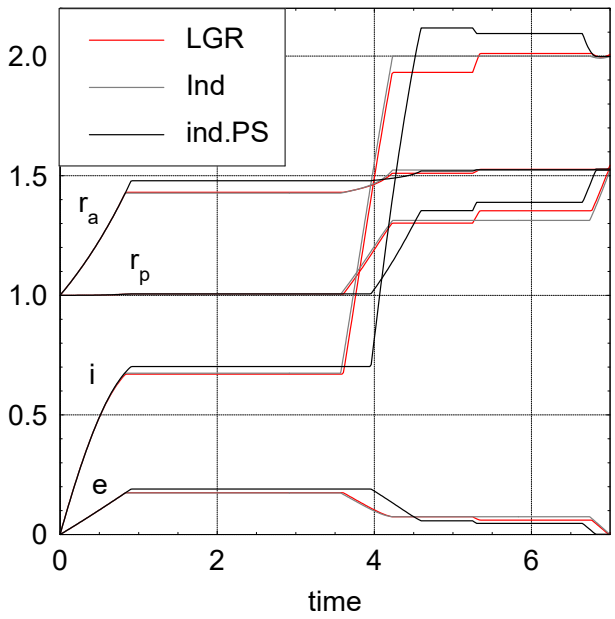


Fig. 4.33 Periapsis, apoapsis, inclination and eccentricity evolution for the 3D Case using 40 nodes and $N_B = 4$, considering Bellman algorithm with LGR and Ind.PS. The thrust was reconstructed using sf.

The cases examined thus far show that the control law derived from Pseudospectral methods, which allow for the utilization of intermediate thrust values, ensure a higher likelihood of reaching the desired target. In contrast, reconstructing the control during integration based on the switching function does not always guarantee the attainment of the desired target conditions. Specifically, in Case 2 solved with Bellman+LGR, significant errors are observed in the final velocity. Nonetheless, notable improvements are observed when the Bellman algorithm is combined with Ind.PS, leveraging the sf value.
Interpolating the discrete control during the integration appears to be the most suitable method for effectively solving these types of problems. However, when a bang-bang control is required, employing the switching function value to reconstruct the control law using Bellman+Ind.PS allows for results that are still sufficiently accurate.

4.2 Formation Reconfiguration

In the preceding section, we employed Pseudospectral methods to solve orbital transfer problems. However, these methods can be utilized in a wide range of applications [105, 49]. In order to highlight their versatility, we have chosen to apply the LGL, LGR, and Ind.PS methods to a formation reconfiguration problem. This section is based on the work done in Ref.[106] and it considers a finite-thrust reconfiguration of the relative motion between two spacecraft. One spacecraft, which defines the reference orbit, is designated the "chief" and it is uncontrolled. The other spacecraft is designated the "deputy" and it is controlled by a three-component thrust input. The procedure searches to reconstruct the thrust profile to reach an aimed final relative configuration, in a specified fixed time window. The study considers Keplerian motion for the two spacecraft and the dynamics equations are developed in terms of Relative Orbital Elements (ROE) [107, 108], that are nonlinear combination of absolute orbital elements of the chief and deputy spacecraft.
In the Earth Centered Inertial reference frame (ECI), a spacecraft orbit can be described using the absolute orbital elements:

$$\mathbf{x} = OE = \begin{pmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ M \end{pmatrix} \tag{4.25}$$

where a is the semi-major axis, e is the eccentricity, i is the inclination, Ω is the right ascension of the ascending node, ω is the argument of perigee and M is the mean anomaly. Considering the chief's orbit as the reference one, it is possible to express the ROE as:

$$\begin{pmatrix} \delta a \\ \delta \Lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \end{pmatrix} = \begin{pmatrix} \dfrac{a_d - a_c}{a_c} \\ (\vartheta_d - \vartheta_c) + (\Omega_d - \Omega_c)\cos i_c \\ e_d \cos\omega_d - e_c \cos\omega_c \\ e_d \sin\omega_d - e_c \sin\omega_c \\ i_d - i_c \\ (\Omega_d - \Omega_c)\sin i_c \end{pmatrix} \tag{4.26}$$

Equation 4.26 can be used for circular reference orbits, but it is still singular for equatorial ones (not used in this paper). Here the subscripts c and d correspond to chief and deputy, respectively, and $\vartheta = \omega + M$ is the mean argument of latitude. For near-circular reference orbit, the relative semi-major axis δa and the relative mean longitude $\delta\Lambda$ represent the offsets in radial and tangential directions in the Radial-Tangential-Normal (RTN) reference frame. Three unit vectors define the RTN frame: $\mathbf{i_r}$ in the radial direction positive outwards from the center of the Earth to the chief, $\mathbf{i_t}$ in the direction of the spacecraft motion and $\mathbf{i_n}$ in the normal direction positive in accordance to the direction of the angular momentum vector. This moving reference frame is centered at the center of mass of the chief satellite, whose motion again defines the reference orbit. δe is the relative eccentricity vector and it describes amplitude and phase motion on the plane of the chief's orbit, whereas δi is the relative inclination vector and it is related to the out-of-plane motion [107].
The problem can be stated as: Maximize:

$$\mathbf{J}(\mathbf{u}) = -\int_{t_0}^{t_f} \sqrt{\mathbf{u}^2} dt \tag{4.27}$$

Subject to:

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} \tag{4.28}$$

$$\mathbf{x}(t_0) = \mathbf{x_0} \tag{4.29}$$

$$\mathbf{x}(t_f) = \mathbf{x_f} \tag{4.30}$$

$$\mathbf{u}(t) \leq 1, t \in [t_0, t_f] \tag{4.31}$$

Here we consider just the in-plane motion, hence the dynamics can be expressed using:

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 \\ -\frac{3}{2}n & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{4.32}$$

where n is the chief's mean motion. Equation 4.32 expresses the Keplerian motion for both spacecraft, since perturbation can be neglected if the time control window is short enough [109]. The simplified Gauss Variational Equations (GVE) [107] map the effects of thrusting vector $\mathbf{u} = u_r\mathbf{i_r} + u_t\mathbf{i_t}$ to variations of ROE, when the

reference orbit is near-circular:

$$B = \begin{pmatrix} 0 & 2 \\ -2 & 0 \\ \sin\vartheta & 2\cos\vartheta \\ -\cos\vartheta & 2\sin\vartheta \end{pmatrix} \tag{4.33}$$

The Hamiltonian can be written as:

$$H = -|\mathbf{u}| + \lambda^T (A\mathbf{x} + B\mathbf{u}) \tag{4.34}$$

According to the Pontryagin's Maximum Principle, the optimal control maximize H. To this purpose, the thrust direction must be

$$\mathbf{u}(t) = u(t)\mathbf{p}(t) \tag{4.35}$$

where $u \in [0,1]$ and $\mathbf{p}$ is the primer vector defined as:

$$\mathbf{p}(t) = B^T \lambda \tag{4.36}$$

For this problem the switching function can be expressed as:

$$sf = |\lambda^T B| - 1 \tag{4.37}$$

whereas the costate differential equations take the following form:

$$\dot{\lambda} = -\frac{\partial H}{\partial \mathbf{x}} = -A^T \lambda \tag{4.38}$$

Considering Equation 4.32, it is possible to state that only $\lambda_{\delta a}$ has a non-null derivative, while the other costates stay constant throughout the trajectory.
The analysis is conducted for formation reconfiguration problems with the following characteristics: the chief orbit is the same throughout the cases and its absolute orbital elements are:

$$\{a, e, i, \Omega, \omega, M\}_c = \{7128.137\text{km}, 0.001, 80.19\,\text{deg}, 50\,\text{deg}, 0\,\text{deg}, 0\,\text{deg}\} \tag{4.39}$$

that is a low-Earth, near circular orbit, with $R_E = 6378.137$km Earth radius. Two study-case are considered: Case$_{\text{FR}}$ 1 presents variations of δa and $\delta\Lambda$ between

Table 4.9 Initial and final ROEs (meters) for Case $_{\text{FR}}$ 1 and 2.

	Case $_{\text{FR}}$ 1	Case $_{\text{FR}}$ 2
δa_0	-100	0
δa_f	0	0
$\delta \Lambda_0$	1000	10000
$\delta \Lambda_f$	9000	10000
δe_{x_0}	500	500
δe_{x_f}	500	200
δe_{y_0}	-10	-10
δe_{y_f}	-10	100

the initial and final configuration with a maximum acceleration $a_{max} = 3e^{-5}\text{m/s}^2$. Whereas, Case $_{\text{FR}}$ 2 considers variations of the eccentricity vector δe_x and δe_y with $a_{max} = 5e^{-5}\text{m/s}^2$. The initial and final relative orbital elements used in Case $_{\text{FR}}$ 1 and 2 are shown in Table 4.9 and the considered final time is $t_f = 2P$, where P defines the chief's orbital period.

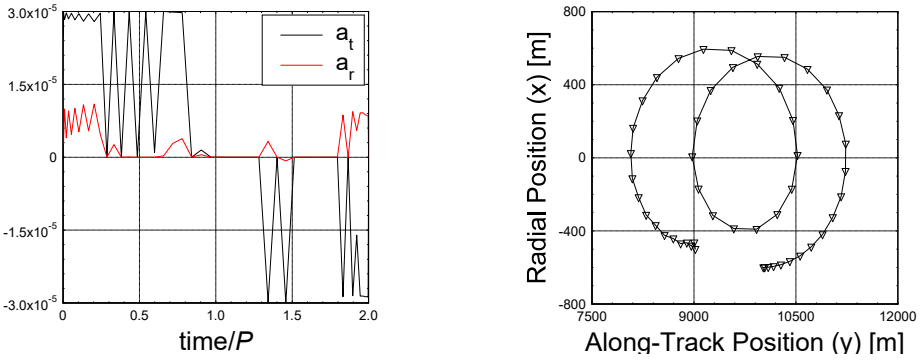
In the following subsections, the two cases are studied using both LGL and LGR PS and Ind.PS methods, then the Bellman method is applied to the same problems. Results are found using 50 nodes and compared, in terms of required ΔV, with the impulsive solutions obtained by Costigliola [110] $\Delta V_{\text{Case}_{\text{FR}}1} = 0.1200\text{m/s}$ and D'Amico [107] $\Delta V_{\text{Case}_{\text{FR}}2} = 0.1676\text{m/s}$.

4.2.1 PS Method

To address the problems using the PS method, it is necessary to have initial tentative solutions for both the states (ROEs) and controls (u, α). In the formation reconfiguration problem, the control angle α is related to the ratio of radial velocity u_r to tangential velocity u_t through the equation $\tan\alpha = u_r/u_t$. For the states, a standard tentative solution can be easily constructed by creating linearly spaced vectors that span from the initial values to the final values. On the other hand, for the control, a constant value of the thrust angle $\alpha = \pi/2$ is arbitrarily considered, along with a tentative u that includes three burning arcs equally spaced throughout the trajectory.

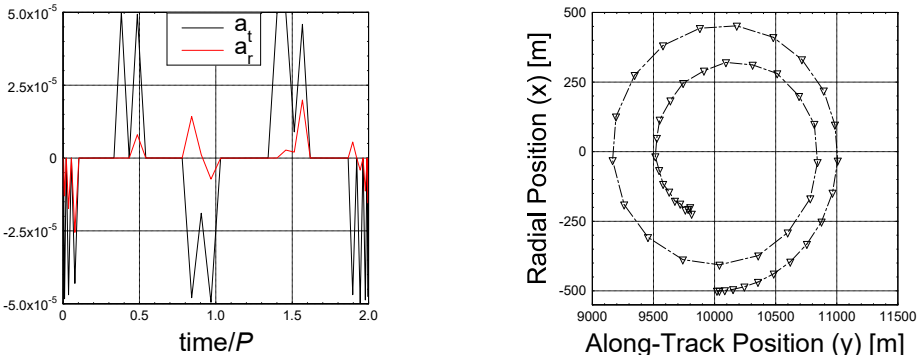
LGL

For Case $_{\text{FR}}$ 1 solved using LGL nodes, the acceleration profile and the corresponding trajectory are presented in Figure 4.34. Results do not show the net thrusting arcs we were expecting; the method tends to favor frequent on/off switches of the engine. Consequently, the final ΔV value is $\Delta V = 0.1557\text{m/s}$, which is 29% higher than the value reported by Costigliola [110] for an impulsive case.



(a) $\mathbf{a_r}$ and $\mathbf{a_t}$ trend (b) Trajectory

Fig. 4.34 Acceleration and trajectory for Case $_{\text{FR}}$ 1 solved with LGL PS method



(a) $\mathbf{a_r}$ and $\mathbf{a_t}$ trend (b) Trajectory

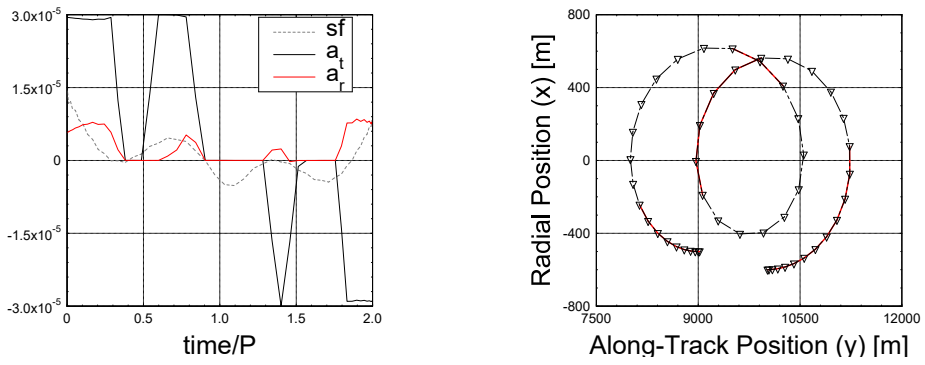
Fig. 4.35 Acceleration and trajectory for Case $_{\text{FR}}$ 2 solved with LGL PS method

While the method is capable of achieving a solution with a ΔV value that is reasonably close to the case involving impulsive thrust, the resulting acceleration profile is not suitable for real problems. Results obtained for Case $_{\text{FR}}$ 2 exhibit similar behavior to Case $_{\text{FR}}$ 1 (Figure 4.35) with the acceleration switching from maximum to zero instead of finding net burning arcs. However, it is possible to

identify five distinct thrusting regions, where the acceleration is primarily directed in the transverse direction. The final $\Delta V = 0.1641\text{m/s}^2$ is 2% lower than the one found by D'Amico [107] (note that the PS solution is an approximation so the obtained mass value may not correspond to the actual one). Also in this case, even though the ΔV estimation is good, the acceleration profiles are not applicable to a real mission concept.

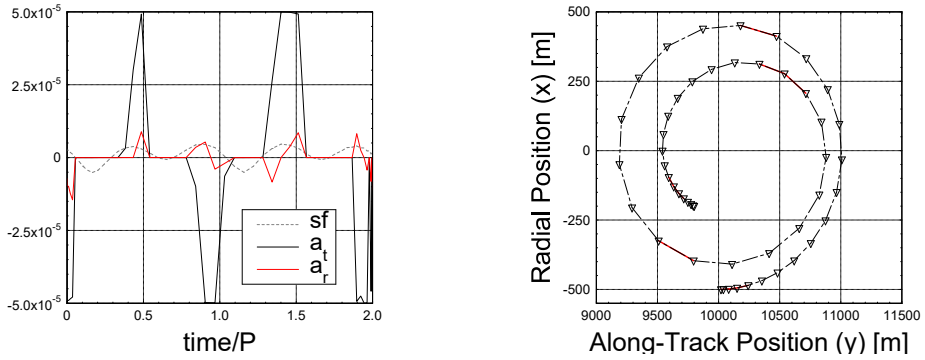
In this thesis, a basic algorithm such as *fmincon* was used to avoid the requirement of sophisticated software. However, when *IPOPT* [111], a software library designed for large-scale nonlinear optimization of continuous systems, is employed, better results are obtained as illustrated in Figure 4.36 and Figure 4.37. Here, it is possible to identify net thrusting arcs, primarily exerted in the tangential direction. Regarding the required ΔV for the reconfiguration, *IPOPT* determines a ΔV of 0.1711m/s for Case$_{\text{FR}}$ 1, which is 9.9% higher than the *fmicon* solution. Conversely, for Case$_{\text{FR}}$ 2, the ΔV is 0.1755m/s, representing a 7% increase compared to *fmincon* result.

The limits of using a less sophisticated optimization toolbox, such as *fmincon*, become clear in this case, as it fails to accurately locate net thrusting arcs. Nonetheless, the following section will show how this problem can also be solved by employing different collocational points, such as LGR.



(a) $\mathbf{a_r}$ and $\mathbf{a_t}$ trend (b) Trajectory

Fig. 4.36 Acceleration and trajectory for Case$_{\text{FR}}$ 1 solved with *IPOPT* LGL PS method

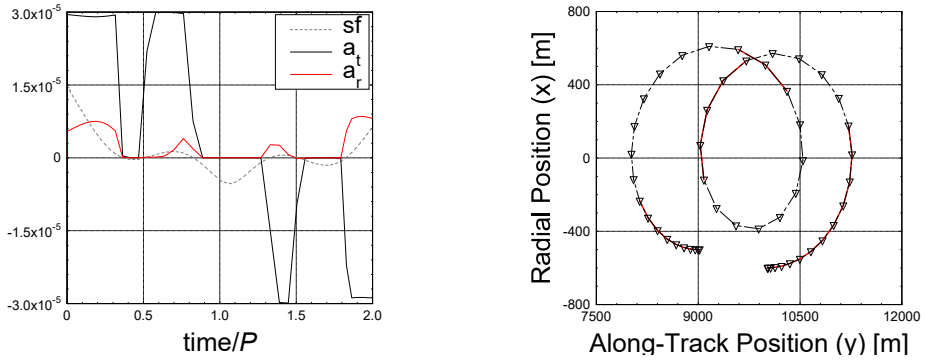


(a) $\mathbf{a_r}$ and $\mathbf{a_t}$ trend (b) Trajectory

Fig. 4.37 Acceleration and trajectory for Case $_{\text{FR}}$ 2 solved with *IPOPT* LGL PS method

LGR

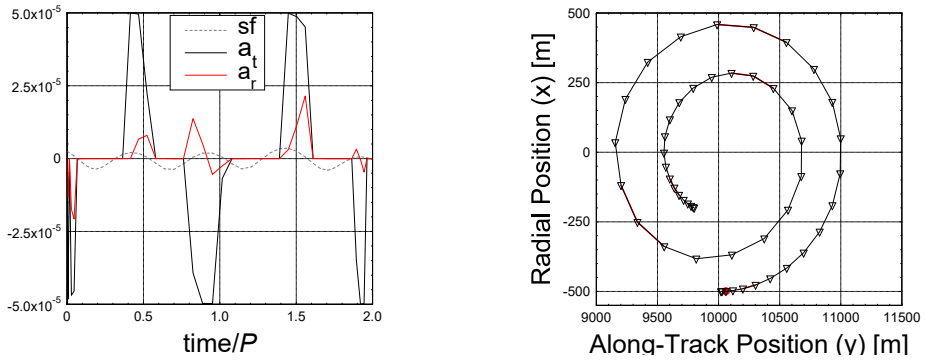
The utilization of the LGR discretization point leads to enhanced outcomes for both cases. In Figure 4.38, four propulsion arcs are depicted for Case $_{\text{FR}}$ 1, displaying an additional arc compared to the three-impulse solution found by Costigliola [110]. This leads to a required ΔV for executing the maneuver that is 48% higher compared to the impulsive scenario. This increase can be attributed to the transition from an impulsive case to a finite thrust scenario, where the impulse is spread across arcs. Consequently, the thrust is applied not only at the most favorable points but also in the surrounding region, leading to a degradation in the final performance.



(a) $\mathbf{a_r}$ and $\mathbf{a_t}$ trend (b) Trajectory

Fig. 4.38 Acceleration and trajectory for Case $_{\text{FR}}$ 1 solved with LGR PS method

In relation to Case $_{\text{FR}}$ 2, a total of five thrusting arcs are observed (Figure 4.39). Nevertheless, when examining the switching function generated exploiting the CMT, it becomes apparent that the fourth firing arc does not align with the region characterized by a positive switching function. This indicates a discrepancy between the Pseudospectral control and the switching function value. However, despite this inconsistency, applying the discrete control found by LGR results in a ΔV that is only 1.4% higher than that of the impulsive case. The significant agreement with the impulsive solution observed here can be attributed to the shorter duration of the thrusting arcs, highlighting the potential of utilizing LGR PS method to solve this class of problems.



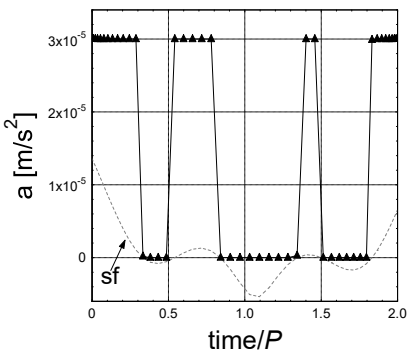
(a) $\mathbf{a_r}$ and $\mathbf{a_t}$ trend (b) Trajectory

Fig. 4.39 Acceleration and trajectory for Case $_{\text{FR}}$ 2 solved with LGR PS method

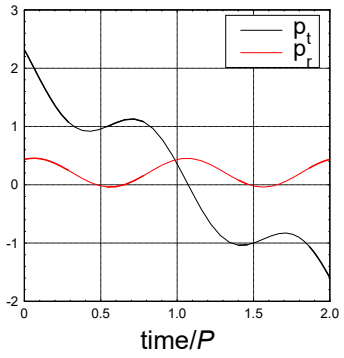
4.2.2 Ind.PS Method

In the Ind.PS method, it is necessary to provide an initial tentative solution for the costates as well. For this purpose, the values of $\lambda_{\delta\Lambda_0}$, $\lambda_{\delta e_{x_0}}$, and $\lambda_{\delta e_{y_0}}$ are set as constants with a value of 0.1. As for $\lambda_{\delta a}$, it is divided into five segments with alternating values of 1 and 0. Furthermore, the smoothing technique is initiated with an ε value of 1 and continues until the ε value reaches $1e^{-7}$, at which point the smoothing process is stopped . Results regarding Case $_{\text{FR}}$ 1 are presented in Figure 4.40. Also in this case, the optimal three-impulse solution in Costigliola [110] is converted in four thrusting arcs. However, it has been verified that when the value of the maximum acceleration is increased, one of the arcs vanishes with the remaining three short arcs in correspondence with the impulses. The acceleration is primarily applied in the tangential direction near the apsides, as depicted in Figure 4.40b, where

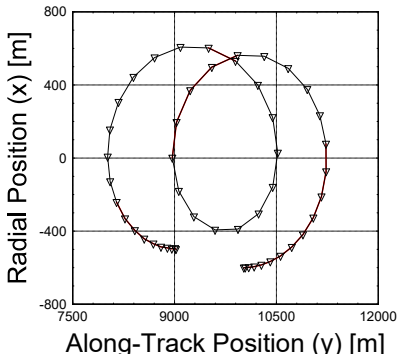
the highlighted regions represent the burning arcs. The ΔV required to perform the reconfiguration is equal to 0.1679m/s, 40% greater than the one obtained with an impulsive strategy. Also in this case, the discrepancy arises due to the distribution of thrust across relatively long arcs, rather than being solely applied in the most advantageous positions. As a result, the overall performance is diminished. However, when the acceleration is increased to $a = 6e^{-5}\text{m/s}^2$, the solution closely resembles the impulsive approach, leading to a decrease in the required ΔV to 0.1388m/s (only 15.7% greater than the optimal value). This demonstrates that by employing lower acceleration values, the ΔV increases due to the use of thrust in non-favorable positions.



(a) Acceleration trend



(b) $\mathbf{p_r}$ and $\mathbf{p_t}$ trend

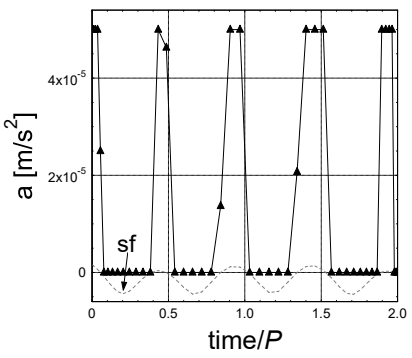


(c) Trajectory

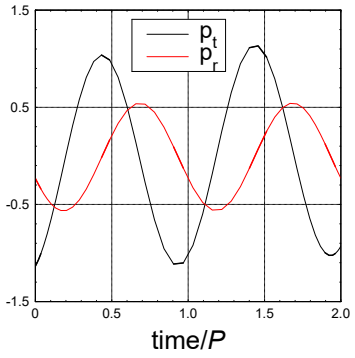
Fig. 4.40 Acceleration, primer vector and trajectory for Case $_{\text{FR}}$ 1 solved with Ind.PS method

On the other hand, Case $_{\text{FR}}$ 2 presents five thrusting arcs, aligned with region of positive sf and distributed as shown in Figure 4.41. Upon analyzing Equation 4.33, it becomes evident that tangential thrust at the apsides yields more favorable results for altering the eccentricity vector. This observation is supported by Figure 4.41b,

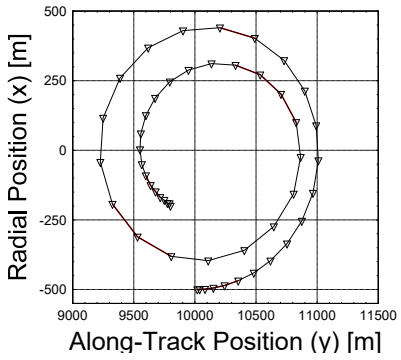
which demonstrates that the thrusting arcs align with the maximum value of $|p_t|$. Remarkably, the obtained results exhibit close agreement with the impulsive strategy. In this case the three-impulse solution is transformed to a five-arc solution. However the duration of these arcs is short (around 15 minutes), resulting in a ΔV that is just 4% greater than the one found by D'Amico [107]. This highlights the suitability of this approach for efficiently and effectively solving such problems.



(a) Acceleration trend



(b) $\mathbf{p_r}$ and $\mathbf{p_t}$ trend



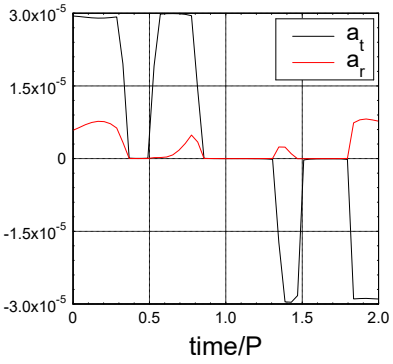
(c) Trajectory

Fig. 4.41 Acceleration, primer vector and trajectory for Case $_{\text{FR}}$ 2 solved with Ind.PS method

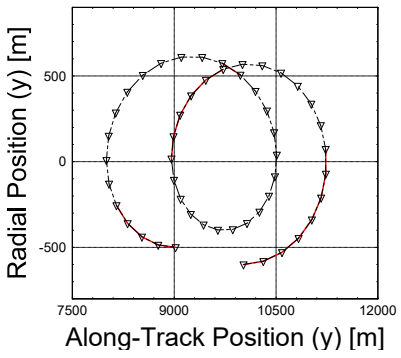
4.2.3 Trapezoidal collocational method

Similar to the approach taken for orbital transfer problems, we aim to assess the performance of the TC method in comparison to the PS methods for formation reconfiguration problems. The results depicted in Figure 4.42 and Figure 4.43, utilizing 50 nodes and the TC method, show excellent agreement with solutions obtained through LGR and Ind.PS. The placement of thrusting arcs is accurate, and the required ΔVs for reconfigurations align with those obtained from different

methods. Specifically, for Case $_{\text{FR}}$ 1 $\Delta V = 0.1721\text{m/s}$ and $\Delta V = 0.1835\text{m/s}$ for Case $_{\text{FR}}$ 2, requiring respectively, an additional 43.4% and 9.5% in change in velocity compared to the impulsive cases. Therefore, it can be affirmed that the TC method is well-suited for addressing these types of problems. Additionally, the computational time of TC is comparable to that of the LGR and Ind.PS methods, since the rate of convergence is slower for all these methods when solving formation reconfiguration problems.

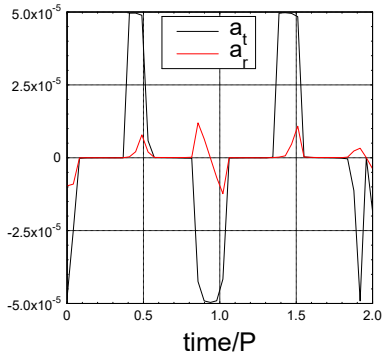


(a) $\mathbf{a_r}$ and $\mathbf{a_t}$ trend

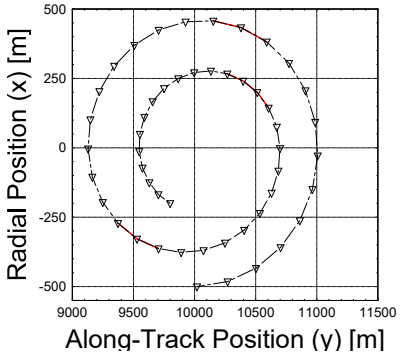


(b) Trajectory

Fig. 4.42 Acceleration and trajectory for Case $_{\text{FR}}$ 1 solved with TC method



(a) $\mathbf{a_r}$ and $\mathbf{a_t}$ trend



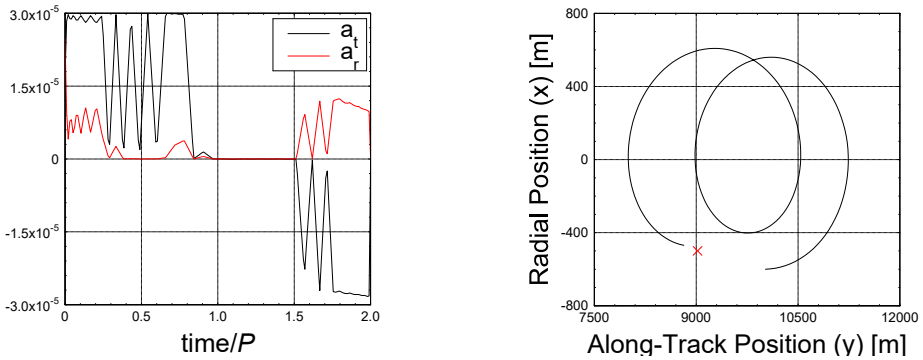
(b) Trajectory

Fig. 4.43 Acceleration and trajectory for Case $_{\text{FR}}$ 2 solved with TC method

4.2.4 Bellman Algorithm

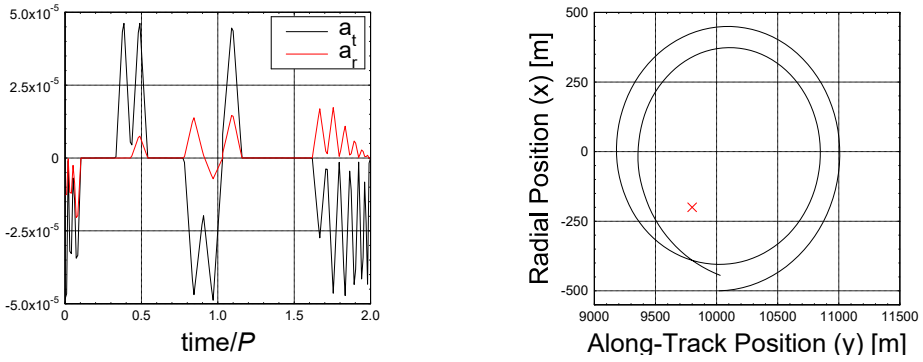
After obtaining results for both PS and Ind.PS methods, the next step involves applying the Bellman algorithm to map the discrete solutions into the continuous

time domain for the same problems. In this study, two and four Bellman segments (NB) are deemed sufficient to achieve satisfactory accuracy and are thus utilized.



(a) $\mathbf{a_r}$ and $\mathbf{a_t}$ trend (b) Trajectory

Fig. 4.44 Acceleration and trajectory for Case $_{\text{FR}}$ 1 solved using Bellman+LGL PS method and $N_B = 2$.



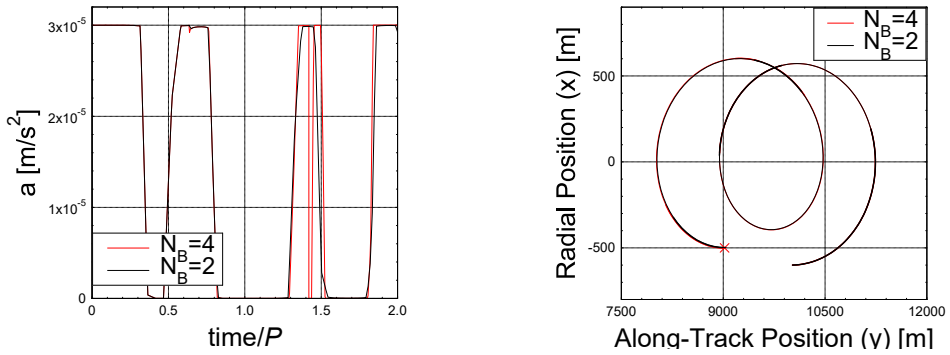
(a) $\mathbf{a_r}$ and $\mathbf{a_t}$ trend (b) Trajectory

Fig. 4.45 Acceleration and trajectory for Case $_{\text{FR}}$ 2 solved using Bellman+LGL PS method and $N_B = 2$.

During the application of the Bellman method in the LGL PS case, the occurrence of numerous engine switches is also observed, leading to significant errors in the final states (on the order of hundreds of meters), as depicted in Figure 4.44 and Figure 4.45. A clear distinction can be observed when comparing Figure 4.45b with Figure 4.35b, where the final position deviates entirely from the desired position (indicated by a red cross). Notably, the errors on the final δ_a and δ_Λ are as large as 115m and 305m, respectively. The LGL PS method appears inadequate for solving this type of bang-bang problem, as it fails to identify an appropriate acceleration profile. Consequently,

when the Bellman algorithm is applied, the resulting trajectories deviate significantly from the required end conditions.

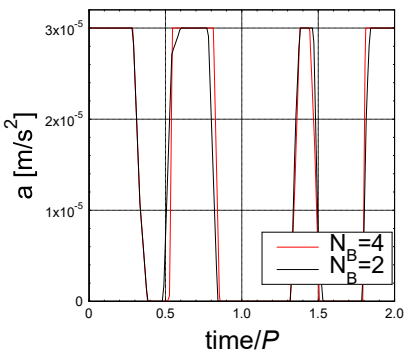
Figure 4.46 and Figure 4.47 showcase the acceleration trend and trajectory derived from the utilization of the Bellman algorithm in conjunction with the LGR PS and Ind.PS method, respectively, for Case$_{\text{FR}}$ 1. Notably, in this instance, the control employed during the integration process is interpolated from the discrete control solution. In Figure 4.46a, using LGR and $N_B = 2$ the thrust retains the same trend as in the discrete solution. However, when considering 4 segments the forth arc is divided in two separated arcs, with a brief coasting arc for $time/P \in [1.42, 1.44]$. On the other hand, when considering solution of the Bellman+Ind.PS (Figure 4.47), four thrusting arcs are still present.



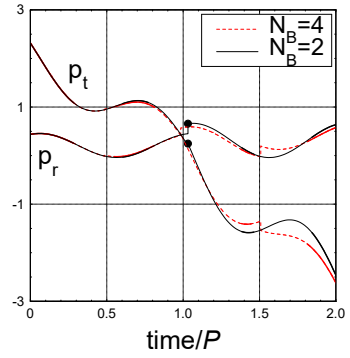
(a) Acceleration trend

(b) Trajectory

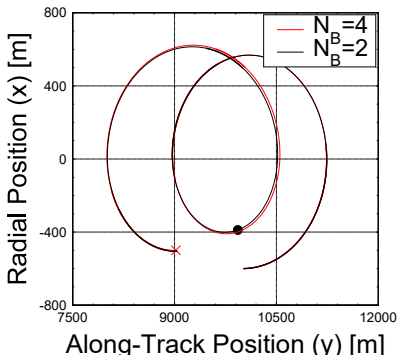
Fig. 4.46 Acceleration and trajectory for Case$_{\text{FR}}$ 1 solved with Bellman+LGR PS method, using discrete control interpolation



(a) Acceleration trend



(b) $\mathbf{p_r}$ and $\mathbf{p_t}$



(c) Trajectory

Fig. 4.47 Acceleration, primer vector and trajectory for Case $_{\text{FR}}$ 1 solved with Bellman+Ind.PS method, using discrete control interpolation

Due to the lack of continuity equations for the costates between the segments, Figure 4.47b shows the components of the primer vector $\mathbf{p}$ display sudden changes in their values. Considering both results using LGR and Ind.PS in the Bellman algorithm, the errors on the final states for Case $_{\text{FR}}$ 1 after integration are presented in Table 4.10. Using 2 segments already achieves a satisfactory level of accuracy in meeting the final condition. However, when 4 segments are employed, the errors are reduced to just a few meters, indicating a higher level of precision. These findings emphasize the effectiveness of interpolating discrete controls during integration, resulting in favorable outcomes within the continuous time domain.

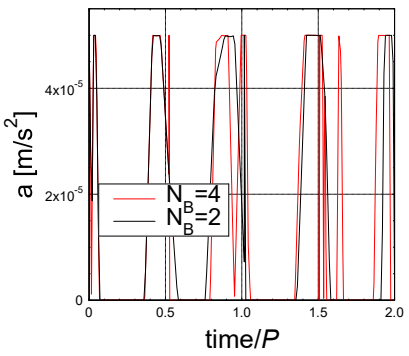
Table 4.10 Errors on the final ROEs after the application of Bellman algorithm for Case $_{\text{FR}}$ 1, using discrete control interpolation.

	Ind.PS		LGR	
Errors on final ROEs [m]	$N_B = 2$	$N_B = 4$	$N_B = 2$	$N_B = 4$
δa_f	4.9	0.5	0.6	0.1
$\delta \Lambda_f$	20.3	8.3	16.6	9.8
δe_{x_f}	0.01	1.5	2.7	0.9
δe_{y_f}	2.8	2.3	0.5	0.5

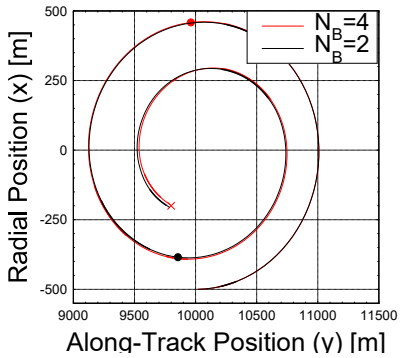
The same considerations can be done for Case $_{\text{FR}}$ 2. Figure 4.48a illustrates the results obtained using Bellman+LGR method. The thrust presents three additional short thrusting arcs when employing $N_B = 4$, loosing the net five-arc solution obtained by the Pseudospectral case (see Figure 4.39a). However, this assures final states well aligned with the desired problem final conditions (Figure 4.48b). Greater errors are instead found for $N_B = 2$, especially for the $\delta \Lambda_f$, where the error is $16.6\text{m}/\text{s}$, as shown in Table 4.11. Hence, also in this case, using 4 segments allows for better results. As regard results obtained with the Ind.PS method, the acceleration retains the five-burning arcs structure both for $N_B = 4$ and 2. The primer vector changes its value through the segments as result of the change in the costates (Figure 4.49b). Moreover, also in this case, $N_B = 2$ presents a $33.7\text{m}/\text{s}$ error on $\delta \Lambda_f$ (Table 4.11), resulting in an end point in the trajectory that goes beyond the required end conditions (red cross). Results improve when using $N_B = 4$, with errors below $3.5\text{m}/\text{s}$ for all the states.

Table 4.11 Errors on the final ROEs after the application of Bellman algorithm for Case $_{\text{FR}}$ 2, using discrete control interpolation.

	Ind.PS		LGR	
Errors on final ROEs [m]	$N_B = 2$	$N_B = 4$	$N_B = 2$	$N_B = 4$
δa_f	6.9	1.5	6	0.9
$\delta \Lambda_f$	33.7	3.4	13.4	4.8
δe_{x_f}	4.1	0.5	1.3	5.8
δe_{y_f}	6.2	0.1	0.8	2

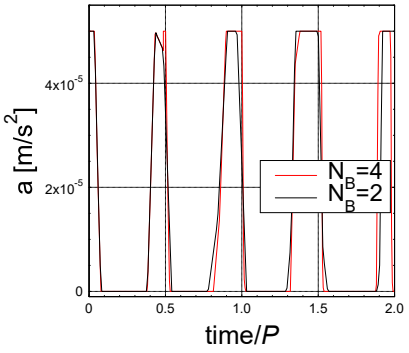


(a) Acceleration trend

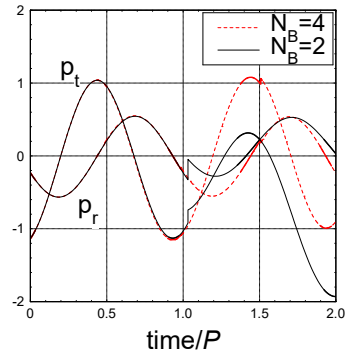


(b) Trajectory

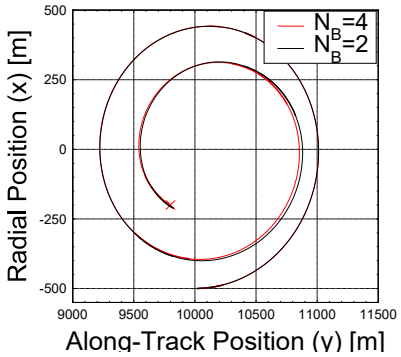
Fig. 4.48 Acceleration and trajectory for Case$_{\mathrm{FR}}$ 2 solved with Bellman+LGR PS method, using discrete control interpolation



(a) Acceleration trend



(b) $\mathbf{p_r}$ and $\mathbf{p_t}$



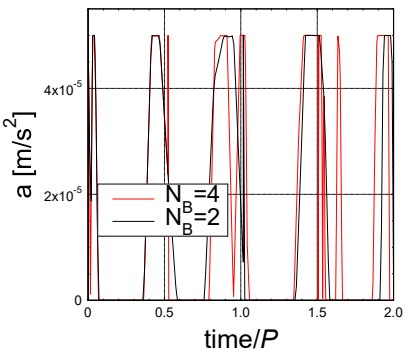
(c) Trajectory

Fig. 4.49 Acceleration, primer vector and trajectory for Case$_{\mathrm{FR}}$ 2 solved with Bellman+Ind.PS method, using discrete control interpolation

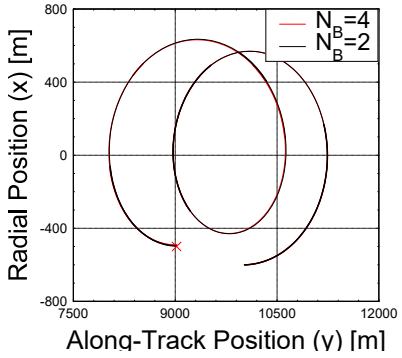
When a pure bang-bang control is sought, it is possible to exploit the switching function to evaluate the thrust magnitude, as already done for the orbital transfer problem. Figure 4.50 displays the results obtained with Bellman+LGR PS for Case$_{\text{FR}}$ 1, using sf. Case$_{\text{FR}}$ 1 presents an additional arc when considering 4 Bellman segments when $time/P \in [1.59, 1.62]$. Even if this does not lead to appreciable differences in the trajectory (Figure 4.50b), Table 4.12 shows smaller errors, below 5m, when considering 4 segments. The same behavior is also observed when using Bellman+Ind.PS, with the addition of a firing arc for $N_B = 4$ when $time/P \in [1.59, 1.62]$. Moreover, when $N_B = 2$, the acceleration is mostly exerted in the tangential direction in the second segment, as shown in Figure 4.51, where, in the highlighted region, $p_t = -1$ and $p_r = 1e^{-6}$. Furthermore, Table 4.12 shows decreasing errors when increasing the number of segments, with the error on $\delta\Lambda_f$ that goes from 81.9m when $N_B = 2$ to 12m when $N_B = 4$.

Table 4.12 Errors on the final ROEs after the application of Bellman algorithm for Case$_{\text{FR}}$ 1, using sf for control reconstruction.

	Ind.PS		LGR	
Errors on final ROEs [m]	$N_B = 2$	$N_B = 4$	$N_B = 2$	$N_B = 4$
δa_f	18.9	1.6	3.6	3.5
$\delta\Lambda_f$	81.9	12	23	1.8
δe_{x_f}	6.4	1.7	6.9	4.9
δe_{y_f}	3.3	2.1	15.7	3.1

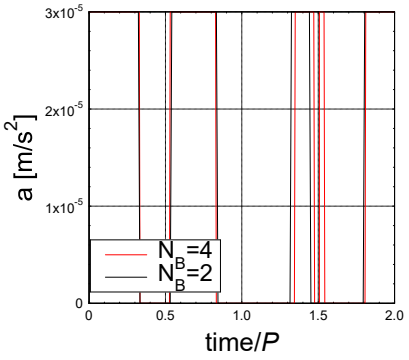


(a) Acceleration trend

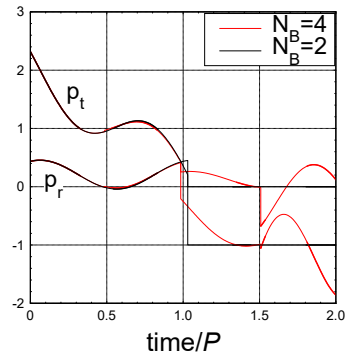


(b) Trajectory

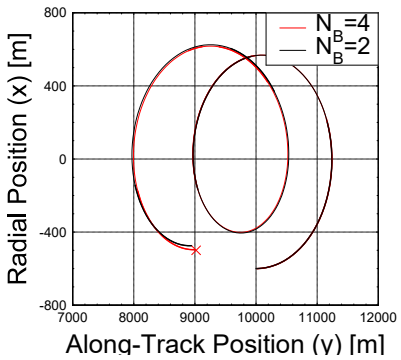
Fig. 4.50 Acceleration and trajectory for Case$_{\text{FR}}$ 1 solved with Bellman+LGR PS method, using sf for control reconstruction



(a) Acceleration trend



(b) $\mathbf{p_r}$ and $\mathbf{p_t}$



(c) Trajectory

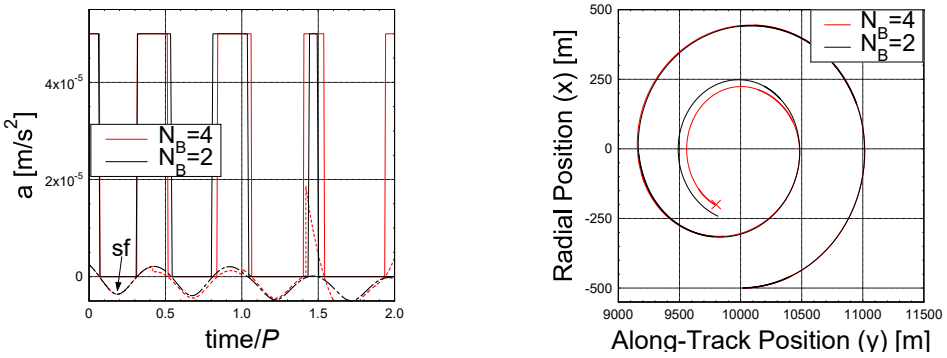
Fig. 4.51 Acceleration, primer vector and trajectory for Case$_{\text{FR}}$ 1 solved with Bellman+Ind.PS method, using sf for control reconstruction

Regarding the resolution of Case$_{\text{FR}}$ 2 using the Bellman algorithm in conjunction with LGR PS, we have observed in Figure 4.39a that the fourth region of positive switching function (sf) does not align with the thrusting arc obtained from the PS method. Additionally, the first arc does not exhibit a maximum thrust value. Consequently, when integrating the system by considering a maximum thrust based on the positive sf, the evolution of the relative orbital elements (ROEs) differs from the pure Pseudospectral solution. Moreover, in the final region of the problem sf does not present positive values, leading to the disappearance of the last thrust arc when employing 2 segments. This discrepancy also results in greater errors in the final states, as illustrated in Table 4.13. However, notable improvements are observed when 4 segments are utilized in the trajectory reconstruction, where the five-arc solution is still found. Moreover, a highly accurate solution is achieved when solving the problem using the Bellman algorithm in conjunction with Ind.PS, as depicted in Figure 4.53. Here an additional brief firing arc is applied when $time/P = [1.52, 1.54]$.

Table 4.13 Errors on the final ROEs after the application of Bellman algorithm for Case $_{\text{FR}}$ 2, using sf for control reconstruction.

	Ind.PS		LGR	
Errors on final ROEs [m]	$N_B = 2$	$N_B = 4$	$N_B = 2$	$N_B = 4$
δa_f	7.6	1	6.5	2.5
$\delta \Lambda_f$	27.2	3.1	10.2	2.2
δe_{x_f}	0.2	0.1	35.3	3.1
δe_{y_f}	0.6	0.2	2.3	5.6

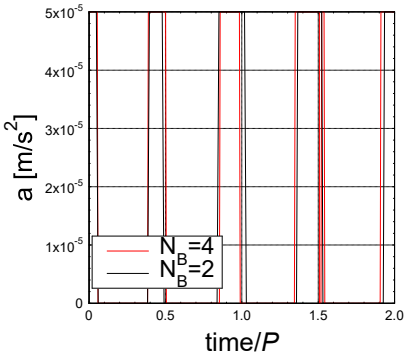
Figure 4.51c shows how the two trajectories deviates after the second burn, since when using $N_B = 4$ the arc lasts longer. Despite these differences both the cases are able to reach really accurate results (Table 4.13), with $N_B = 4$ having always errors below 3.1m.



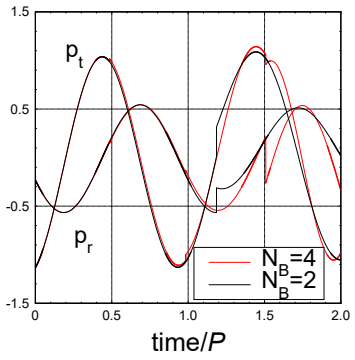
(a) Acceleration trend

(b) Trajectory

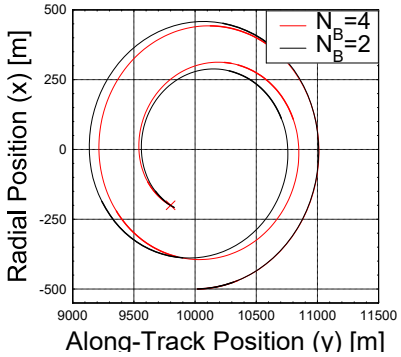
Fig. 4.52 Acceleration and trajectory for Case $_{\text{FR}}$ 2 solved with Bellman+LGR PS method, using sf for control reconstruction



(a) Acceleration trend



(b) $\mathbf{p_r}$ and $\mathbf{p_t}$



(c) Trajectory

Fig. 4.53 Acceleration, primer vector and trajectory for Case$_{\text{FR}}$ 2 solved with Bellman+Ind.PS method, using sf for control reconstruction

The resulting ΔV obtained after the integration are summarized in Table 4.14, when considering a discrete control interpolation and in Table 4.15, when considering sf. Case$_{\text{FR}}$ 1 exhibits larger errors compared to Case$_{\text{FR}}$ 2. As already, explained, this can be attributed to the longer arcs present in Case$_{\text{FR}}$ 1, which result in a higher ΔV compared to the impulsive strategy. The overestimation rapidly decreases as the acceleration is increased, for instance, decreasing from 40% to 15% when the acceleration grows from $3e^{-5}$m/s^2 to $6e^{-5}$m/s^2. Furthermore, using Bellman+LGR with $N_B = 4$ approach, leads to larger ΔV in both cases; this is due to the additional thrusting arcs found by the solution. Furthermore, when reconstructing the bang-bang control profile using the switching function (sf), a larger amount of ΔV is required to execute the reconfiguration. This is primarily due to the fact that, in some cases, the integration process results in the spacecraft deviating from the expected states determined by the PS method. As a consequence, additional thrust arcs are necessary to correct the trajectory and satisfy the final conditions.

Table 4.14 Resulting ΔV after the application of Bellman algorithm, using discrete control interpolation.

		Ind.PS		LGR	
	impulsive	$N_B = 2$	$N_B = 4$	$N_B = 2$	$N_B = 4$
Case $_{\text{FR}}$ 1	0.1200 m/s	0.1732 m/s	0.1660 m/s	0.1712 m/s	0.1849 m/s
Case $_{\text{FR}}$ 2	0.1676 m/s	0.1780 m/s	0.1720 m/s	0.1795 m/s	0.1861 m/s

Table 4.15 Resulting ΔV after the application of Bellman algorithm, using sf.

		Ind.PS		LGR	
	impulsive	$N_B = 2$	$N_B = 4$	$N_B = 2$	$N_B = 4$
Case $_{\text{FR}}$ 1	0.1200 m/s	0.1670 m/s	0.1941 m/s	0.1838 m/s	0.1710m/s
Case $_{\text{FR}}$ 2	0.1676 m/s	0.1871 m/s	0.1972 m/s	0.1689 m/s	0.1769 m/s

This algorithm enables the consideration of the spacecraft's position at a given moment and allows for the formulation of a new optimization problem starting from that point. Consequently, this can lead to a greater demand for ΔV, with potential increases of up to 58% (as observed in Bellman+Ind.PS Case $_{\text{FR}}$ 1 with $N_B = 4$ using sf). Nonetheless, both the methods Bellman+LGR and Bellman+Ind.PS proved to be viable approach to solve formation reconfiguration problem, reconstructing a thrust profile that enables to have small errors on the final mass, albeit at the expense of requiring larger ΔVs .

Chapter 5

Conclusion

The main focus of this thesis was to investigate and enhance the effectiveness and efficiency of Legendre Pseudospectral methods in the context of solving optimal control problems. Both direct and indirect formulations of these methods were thoroughly examined in order to improve their performance. The primary objective was to assess the impact of the number of nodes on the results, considering that reducing the number of nodes can significantly decrease computational time, which is particularly important for potential applications in autonomous guidance systems. Results have been presented for three orbital transfer problems solved using Legendre-Gauss-Lobatto (LGL), Legendre-Gauss-Radau (LGR), and Indirect LGL PS (Ind.PS) methods. The several tests conducted in this thesis show that Legendre PS can successfully obtain accurate solutions using a limited number of nodes in most of the cases. This holds, with some differences, for all the tested formulations. The number of nodes, however, has an impact on performance and accuracy for this kind of bang-bang problem as the placement of nodes is crucial in accurately capturing the behavior of the control and its switching points. For this reason, simply increasing the number of nodes did not always lead to improved outcomes, contrary to what one might intuitively expect. A key aspect to consider is the length of the firing/coasting arc. In general, it is essential to ensure that an adequate number of nodes are positioned within the critical region of the transfer to accurately capture the underlying control law. Careful attention should be given to this aspect to ensure proper detection and representation of the control behavior. However, this thesis showed that, with low number of nodes and without specifically adjusting the node positions, even though the resultant control law may not always perfectly align with

the optimal solution (such as when intermediate thrust values are included), the obtained final masses consistently demonstrate an impressive level of closeness to those of the reference solution. The average error is a mere 0.01%, with a maximum deviation of 0.3%. Among the methods that were evaluated, LGR stands out as the most effective in minimizing errors in the final mass calculation. When it comes to arc detection, Ind.PS excels in reducing the occurrence of intermediate thrust values within its arcs and more closely replicates the optimal control law, except when the sought solution has only very short thrusting arcs. On the other hand, LGR and LGL do not exhibit consistent differences.

By leveraging the Covector Mapping Theorem, we have successfully reconstructed the switching function associated with both the LGL and LGR Pseudospectral methods. Through this mapping, a control law based on the switching function can be used instead of the discrete controls provided by the PS solution. This approach is extremely accurate and precise when LGR formulation is used, as the switching function shows a smooth and regular behavior. In contrast, the approach is unsatisfactory in the case of LGL points, with the switching function showing an oscillatory trend at the boundary of the problem. This behavior arises due to the need to satisfy the double boundary conditions required for closure.

An additional approach that combines the indirect formulation with PS interpolation is proposed in this thesis, in order to tackle the challenges encountered by the direct method when dealing with bang-bang control problems. Results show that the indirect formulation (Ind.PS) overcomes the issues associated with the oscillating switching function observed in the direct method. This achievement is made possible by directly computing the switching function using the problem's adjoint variables, eliminating the reliance on KKT multipliers as mandated by the Covector Mapping Theorem. As a result, the indirect formulation yields a switching function that is not only smooth but also perfectly aligned with the thrust values obtained through the smoothing technique. Besides, already mentioned issues may arise when short thrusting arcs are present.

The computational efficiency and cost functional evaluation of three PS methods—LGL, LGR, and Ind.PS— is also compared with the Trapezoidal collocational method. The analysis revealed that, with an increasing number of nodes, LGR proves to be the most efficient method for problem resolution both in terms of computational time and cost function evaluation. In contrast, Ind.PS and TC display relatively higher computational time. While the slower convergence of Ind.PS can be attributed

to the utilization of a smoothing technique, TC's slower convergence seems just to be the result of a lower rate of convergence.

After deriving solutions using a limited number of discretization points through the direct and indirect Pseudospectral methods, it became essential to assess their practical applicability from an engineering standpoint. To enable the transition from discrete-time to continuous-time solutions, the integration of the Pseudospectral methods with the Bellman algorithm was employed. Through this comprehensive analysis, the practical feasibility of the obtained solutions was rigorously evaluated. This work explored two distinct approaches for reconstructing the control during integration. The first approach involved interpolating the discrete control obtained from the Pseudospectral methods. The second approach leveraged the knowledge of the switching function value to impose a purely bang-bang control. Both methods were thoroughly investigated to determine their effectiveness in accurately reconstructing the control profiles and the satisfaction of the target conditions after the integration. The interpolation of the discrete control led to results that aligned more closely with the desired final conditions. The errors between the adimensional final states and the target states fall within the range of $2e^{-4}$. Whereas, when employing the switching function to impose the thrust value, the average errors in the final states increase by an order of magnitude to $2e^{-3}$. These errors still remain within the acceptable range of 0.1%. This disparity arises from the fact that the optimization problem is solved using a Pseudospectral method, and through the Bellman method, we observe that the discrete control effectively guides the system toward the target conditions. On the contrary, when employing a bang-bang control based on the switching function, the control law is intrinsically different from the one adopted by the PS method to obtain the trajectory and therefore the evolution of the states might deviate from the discrete solution. However, it was insightful to analyze this alternative approach, which leveraged the knowledge of the switching function obtained through the covector mapping theorem. Although the results are one order of magnitude less accurate compared to interpolation, they are still fully acceptable. Moreover, this method provides an opportunity to explore the benefits of incorporating the switching function into the Pseudospectral control scheme, as this control law can be applied to cases where the thruster has only an on/off throttle, and intermediate thrust values are not allowed.

A very different problem was also analyzed in order to explore the flexibility and adaptability of the methods developed in this thesis. Pseudospectral methods and

Bellman algorithm were used to solve two low-thrust formation reconfiguration problems, utilizing the comprehensive knowledge we acquired throughout this research. The results revealed that both LGR and the Ind.PS proved to be very effective in solving these formation reconfigurations, while the LGL method was not suitable for this type of problem, as it generated acceleration profiles that were impractical for real scenarios and it required more sophisticated optimization toolbox to improve its results. Once the discrete solutions were obtained, the application of the Bellman algorithm to these case studies was also explored. By using four segments, it was possible to achieve small errors between the final and target conditions (never exceeding 12m) at the expense of higher required ΔV (up to a 58% of additional change in velocity). This increase in ΔV can be attributed to several factors. First of all, the consideration of firing arcs instead of impulsive maneuvers introduced discrepancies due to the distribution of thrust over relatively long durations, rather than applying it solely at the most advantageous points. An increase in the acceleration magnitude shows that this is the dominant effect in the growth of the ΔVs. Additionally, the implementation of the Bellman algorithm shows that additional thrusting arcs may be required to correct discrepancies in the orbital changes obtained in the previous segments with respect to the optimal ones. These adjustments are introduced by Bellman algorithm, as it considers a new optimization problem starting from the state at the end of each segment: they are fundamental to achieve the final state with sufficient accuracy, but penalize the propellant consumption.
Overall, the combination of Pseudospectral methods and the Bellman algorithm provided valuable insights into the feasibility and challenges associated with solving formation reconfiguration problems, highlighting the importance of careful method selection and the potential trade-off between solution accuracy and required ΔV.
The utilization of PS methods, both in direct and indirect formulation, has demonstrated that a reduced number of nodes (ranging from 20 to 50) is sufficient for approximating the optimal solutions in the considered orbital transfer and formation reconfiguration problems. These computations can be completed within a few seconds in most cases and always within a minute, making them highly compatible with real-time on-board applications for the examined maneuvers. Notably, these methods accurately identify both the cost function value and successfully derive trajectory control laws that align with the optimal solution. Moreover, the indirect formulation of the Pseudospectral method has demonstrated its effectiveness in overcoming the challenges encountered by the direct Pseudospectral methods when dealing with

bang-bang control problems. It effectively solved issues such as incorporating intermediate thrust values into the control law and it accurately evaluated the switching function. The Bellman algorithm further demonstrates the practical applicability of discrete solutions in a continuous time domain. Generally, the errors in achieving the final desired state are typically lower than 0.1%, and only a few exceptional cases show unsatisfactory results.

In future research, it would be intriguing to investigate problems with longer durations and a higher number of revolutions to determine if the current number of considered nodes remains adequate to accurately evaluate the cost function and the control law reconstruction. Additionally, while global interpolation methods were employed throughout the thesis, exploring the use of Pseudospectral methods on a local scale could be valuable to better detect the switching regions. The problem interval can be divided into segments, with each segment being spanned by a global interpolating polynomial. These segments will then be connected using linkage conditions for the continuity of the states.

Further enhancements can also be applied to the Bellman algorithm. Currently, the segments are evenly spaced during the algorithm, however, it would be interesting to explore solutions obtained by correlating the beginning and end points of a firing arc with the corresponding beginning and end of a Bellman segment. This approach would result in closely spaced nodes being concentrated in these critical regions, of great interest when solving bang-bang control problems. Alternatively, another intriguing concept to investigate is positioning the start of a segment in a region where the Pseudospectral solution has exhibited an intermediate thrust value. By increasing the number of nodes in these specific regions, a more comprehensive understanding of the control behavior in those areas could be obtained.

It would be highly beneficial to expand the application of the proposed methods to encompass rendezvous missions and formation reconfiguration problems in a three-dimensional setting. The inclusion of rendezvous missions would involve orbital transfers from a given orbit to a specific time-dependent position on a target orbit. On the other hand, addressing formation reconfiguration would require accounting for the influence of the J_2 effect, which introduces perturbations due to the Earth's oblateness, and considering circular as well as elliptic orbits. These extensions would significantly contribute to the development of realistic and robust approaches for practical scenarios. They would greatly enhance our ability to design efficient and

accurate maneuver plans, taking into account the complexities of operating within realistic orbital environments.

References

[1] J.C. Willems. 1696: the birth of optimal control. In *Proceedings of 35th IEEE Conference on Decision and Control*, volume 2, pages 1586–1587 vol.2, 1996.

[2] Leonhard Euler. Elementa calculi variationum. *Novi commentarii academiae scientiarum Petropolitanae*, pages 51–93, 1766.

[3] Donald E Kirk. *Optimal control theory: an introduction*. Courier Corporation, 2004.

[4] Lev Semenovich Pontryagin. *Mathematical theory of optimal processes*. CRC press, 1987.

[5] A.E. Bryson. Optimal control-1950 to 1985. *IEEE Control Systems Magazine*, 16(3):26–33, 1996.

[6] Derek F Lawden. *Optimal trajectories for space navigation*, volume 3. Butterworths, 1963.

[7] JR Foots. Optimization techniques (g. leitmann, 1963.

[8] Angelo Miele. Recent advances in gradient algorithms for optimal control problems. *Journal of Optimization Theory and Applications*, 17:361–430, 1975.

[9] John V. Breakwell. The optimization of trajectories. *Journal of the Society for Industrial and Applied Mathematics*, 7(2):215–247, 1959.

[10] Henry J. Kelley. Gradient theory of optimal flight paths. *ARS Journal*, 30:947–954, 1960.

[11] Arthur Earl Bryson and Walter F Denham. A steepest-ascent method for solving optimum programming problems. 1962.

[12] John T Betts. Survey of numerical methods for trajectory optimization. *Journal of guidance, control, and dynamics*, 21(2):193–207, 1998.

[13] John T Betts. *Practical methods for optimal control and estimation using nonlinear programming*. SIAM, 2010.

[14] S Vasantharajan and LT Biegler. Simultaneous strategies for optimization of differential-algebraic systems with enforcement of error criteria. *Computers & chemical engineering*, 14(10):1083–1100, 1990.

[15] AE Bryson Jr and Stanley E Ross. Optimum rocket trajectories with aerodynamic drag. *Journal of Jet Propulsion*, 28(7):465–469, 1958.

[16] Hans Georg Bock and Karl-Josef Plitt. A multiple shooting algorithm for direct solution of optimal control problems. *IFAC Proceedings Volumes*, 17(2):1603–1608, 1984.

[17] Paul J Enright and Bruce A Conway. Discrete approximations to optimal trajectories using direct transcription and nonlinear programming. *Journal of Guidance, Control, and Dynamics*, 15(4):994–1002, 1992.

[18] Stephen Paris, John Riehl, and Waldy Sjauw. Enhanced procedures for direct trajectory optimization using nonlinear programming and implicit integration. In *AIAA/AAS astrodynamics specialist conference and exhibit*, page 6309, 2006.

[19] Gamal Elnagar, Mohammad A Kazemi, and Mohsen Razzaghi. The pseudospectral legendre method for discretizing optimal control problems. *IEEE transactions on Automatic Control*, 40(10):1793–1796, 1995.

[20] Bengt Fornberg. *A practical guide to pseudospectral methods*. Number 1. Cambridge university press, 1998.

[21] C Canuto, MY Hussaini, TA Zang Jr, and A Quarteroni. Spectral methods in fluid dynamics. 2. rev. 1991.

[22] Daniel Henry Gottlieb, M. Yousuff Hussaini, and Steven A. Orszag. Theory and applications of spectral methods. 1984.

[23] John Villadsen and Michael L Michelsen. Solution of differential equation models by polynomial approximation(book). *Englewood Cliffs, N. J., Prentice-Hall, Inc., 1978. 460 p*, 1978.

[24] Nazareth Bedrossian, Sagar Bhatt, Michael Lammers, Louis Nguyen, and Yin Zhang. First ever flight demonstration of zero propellant maneuver (tm) attitute control concept. In *AIAA Guidance, Navigation and Control Conference and Exhibit*, page 6734, 2007.

[25] Nazareth S. Bedrossian, Sagar Bhatt, Wei Kang, and I. Michael Ross. Zero-propellant maneuver guidance. *IEEE Control Systems Magazine*, 29(5):53–73, 2009.

[26] Ping Lu, Hongsheng Sun, and Bruce Tsai. Closed-loop endoatmospheric ascent guidance. *Journal of Guidance, Control, and Dynamics*, 26(2):283–294, 2003.

[27] Da Zhang, Lei Liu, and Yongji Wang. On-line ascent phase trajectory optimal guidance algorithm based on pseudo-spectral method and sensitivity updates. *The Journal of Navigation*, 68(6):1056–1074, 2015.

[28] Robert Melton. Comparison of direct optimization methods applied to solar sail problems. In *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, page 4728, 2002.

[29] Jeannette Heiligers, Giorgio Mingotti, and Colin McInnes. Optimisation of solar sail interplanetary heteroclinic connections. In *2nd Conference on Dynamics and Control of Space Systems, DyCoSS2*, pages AIAA–AAS, 2014.

[30] John Riehl, Stephen Paris, and Waldy Sjauw. Comparison of implicit integration methods for solving aerospace trajectory optimization problems. In *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, page 6033, 2006.

[31] Kathryn F Graham and Anil V Rao. Minimum-time trajectory optimization of multiple revolution low-thrust earth-orbit transfers. *Journal of spacecraft and rockets*, 52(3):711–727, 2015.

[32] Tawfiqur Rahman, Hao Zhou, Liang Yang, and Wanchun Chen. Pseudospectral model predictive control for exo-atmospheric guidance. *International Journal of Aeronautical and Space Sciences*, 16(1):64–76, 2015.

[33] Xiuqiang Jiang, Shuang Li, and Ting Tao. Innovative hazard detection and avoidance guidance for safe lunar landing. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 230(11):2086–2103, 2016.

[34] Jeremy Rea. Launch vehicle trajectory optimization using a legendre pseudospectral method. In *AIAA guidance, navigation, and control conference and exhibit*, page 5640, 2003.

[35] Mathieu Balesdent, Loïc Brevault, Jorge-Luis Valderrama-Zapata, and Annafederica Urbano. All-at-once formulation integrating pseudo-spectral optimal control for launch vehicle design and uncertainty quantification. *Acta Astronautica*, 200:462–477, 2022.

[36] SI Infeld and W Murray. Optimization of stationkeeping for a libration point mission. In *AAS Spaceflight Mechanics Meeting*, pages 04–150, 2004.

[37] Ying-Jing Qian, Yue Liu, Wei Zhang, Xiao-Dong Yang, and Ming-Hui Yao. Stationkeeping strategy for quasi-periodic orbit around earth–moon l 2 point. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 230(4):760–775, 2016.

[38] Gamal N Elnagar and Mohammad A Kazemi. Pseudospectral chebyshev optimal control of constrained nonlinear dynamical systems. *Computational Optimization and Applications*, 11(2):195–217, 1998.

[39] Z Sabeh, M Shamsi, and Mehdi Dehghan. Distributed optimal control of the viscous burgers equation via a legendre pseudo-spectral approach. *Mathematical Methods in the Applied Sciences*, 39(12):3350–3360, 2016.

[40] Lianghui Tu, Yuhao Wang, Chao Yan, and Yang Yang. Optimal transfer orbit design of spacecraft with finite thrust based on legendre pseudospectral method. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, page 09544100221138164, 2022.

[41] Stuart Andres Stanton. *Optimal orbital transfer using a legendre pseudospectral method*. PhD thesis, Massachusetts Institute of Technology, 2003.

[42] Paul Williams. Jacobi pseudospectral method for solving optimal control problems. *Journal of Guidance, Control, and Dynamics*, 27(2):293–297, 2004.

[43] Fariba Fahroo and I Michael Ross. Direct trajectory optimization by a chebyshev pseudospectral method. *Journal of Guidance, Control, and Dynamics*, 25(1):160–166, 2002.

[44] Fariba Fahroo and I Michael Ross. Costate estimation by a legendre pseudospectral method. *Journal of Guidance, Control, and Dynamics*, 24(2):270–277, 2001.

[45] Fariba Fahroo and I Michael Ross. Pseudospectral methods for infinite-horizon nonlinear optimal control problems. *Journal of Guidance, Control, and Dynamics*, 31(4):927–936, 2008.

[46] S Karneswaran and Lorenz T Biegler. Convergence rates for direct transcription of optimal control problems with final-time equality constraints using collocation at radau points. In *2006 American Control Conference*, pages 7–pp. IEEE, 2006.

[47] Marco Sagliano, Stephan Theil, Michiel Bergsma, Vincenzo D'Onofrio, Lisa Whittle, and Giulia Viavattene. On the radau pseudospectral method: theoretical and implementation advances. *CEAS Space Journal*, 9:313–331, 2017.

[48] David A Benson, Geoffrey T Huntington, Tom P Thorvaldsen, and Anil V Rao. Direct trajectory optimization and costate estimation via an orthogonal collocation method. *Journal of Guidance, Control, and Dynamics*, 29(6):1435–1440, 2006.

[49] Geoffrey T Huntington and Anil V Rao. Optimal reconfiguration of spacecraft formations using the gauss pseudospectral method. *Journal of Guidance, Control, and Dynamics*, 31(3):689–698, 2008.

[50] Paul Williams. Hermite-legendre-gauss-lobatto direct transcription in trajectory optimization. *Journal of guidance, control, and dynamics*, 32(4):1392–1395, 2009.

[51] John T Betts and Sven O Erb. Optimal low thrust trajectories to the moon. *SIAM Journal on Applied Dynamical Systems*, 2(2):144–170, 2003.

[52] Philip E Gill, Walter Murray, and Michael A Saunders. Snopt: An sqp algorithm for large-scale constrained optimization. *SIAM review*, 47(1):99–131, 2005.

[53] Michael C Ferris and Todd S Munson. Interior-point methods for massive support vector machines. *SIAM Journal on Optimization*, 13(3):783–804, 2002.

[54] I Michael Ross, Qi Gong, and Pooya Sekhavat. Low-thrust, high-accuracy trajectory optimization. *Journal of Guidance, Control, and Dynamics*, 30(4):921–933, 2007.

[55] I Michael Ross, Qi Gong, and Pooya Sekhavat. Bellman pseudospectral method. In *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, page 6448, 2008.

[56] Sayyed Mohammad Hoseini. A composite pseudospectral method for solving multi-delay optimal control problems involving piecewise constant delay functions. *Journal of the Franklin Institute*, 355(16):7895–7923, 2018.

[57] M Khaksar-e Oshagh and M Shamsi. Direct pseudo-spectral method for optimal control of obstacle problem–an optimal control problem governed by elliptic variational inequality. *Mathematical Methods in the Applied Sciences*, 40(13):4993–5004, 2017.

[58] Mohammad Maleki and Ishak Hashim. Adaptive pseudospectral methods for solving constrained linear and nonlinear time-delay optimal control problems. *Journal of the Franklin Institute*, 351(2):811–839, 2014.

[59] Divya Garg, Michael Patterson, William Hager, Anil Rao, David R Benson, and Geoffrey T Huntington. An overview of three pseudospectral methods for the numerical solution of optimal control problems. 2017.

[60] I Michael Ross and Fariba Fahroo. Pseudospectral knotting methods for solving nonsmooth optimal control problems. *Journal of Guidance, Control, and Dynamics*, 27(3):397–405, 2004.

[61] Qi Gong, Fariba Fahroo, and I Michael Ross. Spectral algorithm for pseudospectral methods in optimal control. *Journal of Guidance, Control, and Dynamics*, 31(3):460–471, 2008.

[62] Christopher L Darby, William W Hager, and Anil V Rao. An hp-adaptive pseudospectral method for solving optimal control problems. *Optimal Control Applications and Methods*, 32(4):476–502, 2011.

[63] M Shamsi. A modified pseudospectral scheme for accurate solution of bang-bang optimal control problems. *Optimal Control Applications and Methods*, 32(6):668–680, 2011.

[64] Tieding Guo, Fanghua Jiang, and Junfeng Li. Homotopic approach and pseudospectral method applied jointly to low thrust trajectory optimization. *Acta Astronautica*, 71:38–50, 2012.

[65] Mohammad A Mehrpouya. A modified pseudospectral method for indirect solving a class of switching optimal control problems. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 234(9):1531–1542, 2020.

[66] Mohammad Ali Mehrpouya and Haijun Peng. A robust pseudospectral method for numerical solution of nonlinear optimal control problems. *International Journal of Computer Mathematics*, 98(6):1146–1165, 2021.

[67] W Hohmann. Die erreichbarkeit der himmelskörper (the accessibility of celestial bodies), 1925.

[68] John E Prussing. Simple proof of the global optimality of the hohmann transfer. *Journal of Guidance, Control, and Dynamics*, 15(4):1037–1038, 1992.

[69] RB Barrar. An analytic proof that the hohmann-type transfer is the true minimum two-impulse transfer. Technical report, SYSTEM DEVELOPMENT CORP SANTA MONICA CALIF, 1962.

[70] Denílson Paulo Souza dos Santos and Jorge Kennety da Silva Formiga. Application of a genetic algorithm in orbital maneuvers. *Computational and Applied Mathematics*, 34:437–450, 2015.

[71] Mauro Pontani and Bruce A Conway. Particle swarm optimization applied to impulsive orbital transfers. *Acta Astronautica*, 74:141–155, 2012.

[72] Ralph Kahle, Hartmut Runge, Jean-Sebastien Ardaens, Steffen Suchandt, and Roland Romeiser. Formation flying for along-track interferometric oceanography—first in-flight demonstration with tandem-x. *Acta Astronautica*, 99:130–142, 2014.

[73] Eberhard Gill, Simone D'Amico, and Oliver Montenbruck. Autonomous formation flying for the prisma mission. *Journal of Spacecraft and Rockets*, 44(3):671–681, 2007.

[74] Per Bodin, Ron Noteborn, Robin Larsson, Thomas Karlsson, Simone D'Amico, Jean Sebastien Ardaens, Michel Delpech, and Jean-Claude Berges. The prisma formation flying demonstrator: Overview and conclusions from the nominal mission. *Advances in the Astronautical Sciences*, 144(2012):441–460, 2012.

[75] Gabriella Gaias and Jean-Sébastien Ardaens. Flight demonstration of autonomous noncooperative rendezvous in low earth orbit. *Journal of Guidance, Control, and Dynamics*, 41(6):1337–1354, 2018.

[76] M Guelman and M Aleshin. Optimal bounded low-thrust rendezvous with fixed terminal-approach direction. *Journal of Guidance, Control, and Dynamics*, 24(2):378–385, 2001.

[77] Riccardo Bevilacqua, Jason S Hall, and Marcello Romano. Multiple spacecraft rendezvous maneuvers by differential drag and low thrust engines. *Celestial Mechanics and Dynamical Astronomy*, 106(1):69–88, 2010.

[78] Andrea Garulli, Antonio Giannitrapani, Mirko Leomanni, and Fabrizio Scortecci. Autonomous low-earth-orbit station-keeping with electric propulsion. *Journal of guidance, control, and dynamics*, 34(6):1683–1693, 2011.

[79] Arthur Earl Bryson. *Applied optimal control: optimization, estimation, and control*. Routledge, 2018.

[80] Frank L Lewis, Draguna Vrabie, and Vassilis L Syrmos. *Optimal control*. John Wiley & Sons, 2012.

[81] Philip J Davis. *Interpolation and approximation*. Courier Corporation, 1975.

[82] Carl Runge. Über empirische funktionen und die interpolation zwischen äquidistanten ordinaten. *Zeitschrift für Mathematik und Physik*, 46(224-243):20, 1901.

[83] John P Boyd. *Chebyshev and Fourier spectral methods*. Courier Corporation, 2001.

[84] Philip J Davis and Philip Rabinowitz. *Methods of numerical integration*. Courier Corporation, 2007.

[85] I Michael Ross and Fariba Fahroo. A pseudospectral transformation of the convectors of optimal control systems. *IFAC Proceedings Volumes*, 34(13):543–548, 2001.

[86] Fariba Fahroo and I Michael Ross. Advances in pseudospectral methods for optimal control. In *AIAA guidance, navigation and control conference and exhibit*, page 7309, 2008.

[87] I Michael Ross and Mark Karpenko. A review of pseudospectral optimal control: From theory to flight. *Annual Reviews in Control*, 36(2):182–197, 2012.

[88] I Michael Ross and Fariba Fahroo. Legendre pseudospectral approximations of optimal control problems. In *New trends in nonlinear dynamics and control and their applications*, pages 327–342. Springer, 2003.

[89] I Michael Ross. A historical introduction to the covector mapping principle. In *Proceedings of Astrodynamics Specialists Conference*. Citeseer, 2005.

[90] Qi Gong, I Michael Ross, Wei Kang, and Fariba Fahroo. Connections between the covector mapping theorem and convergence of pseudospectral methods for optimal control. *Computational Optimization and Applications*, 41:307–335, 2008.

[91] Richard F Hartl, Suresh P Sethi, and Raymond G Vickson. A survey of the maximum principles for optimal control problems with state constraints. *SIAM review*, 37(2):181–218, 1995.

[92] I Michael Ross and Fariba Fahroo. A perspective on methods for trajectory optimization. In *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, page 4727, 2002.

[93] Fariba Fahroo and I Ross. Trajectory optimization by indirect spectral collocation methods. In *Astrodynamics specialist conference*, page 4028, 2000.

[94] Régis Bertrand and Richard Epenoy. New smoothing techniques for solving bang–bang optimal control problems—numerical results and statistical interpretation. *Optimal Control Applications and Methods*, 23(4):171–197, 2002.

[95] Ehsan Taheri and John L Junkins. Generic smoothing for optimal bang-off-bang spacecraft maneuvers. *Journal of Guidance, Control, and Dynamics*, 41(11):2470–2475, 2018.

[96] Lamberto Cesari. *Optimization—theory and applications: problems with ordinary differential equations*, volume 17. Springer Science & Business Media, 2012.

[97] Ehsan Taheri and Ossama Abdelkhalik. Shape based approximation of constrained low-thrust space trajectories using fourier series. *Journal of Spacecraft and Rockets*, 49(3):535–546, 2012.

[98] Geoffrey Todd Huntington. *Advancement and analysis of a Gauss pseudospectral transcription for optimal control problems*. PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and . . . , 2007.

[99] H Gardner Moyer and Gordon Pinkham. Several trajectory optimization techniques: Part ii: Application. In *Computing Methods in Optimization Problems*, pages 91–105. Elsevier, 1964.

[100] William W Hager. Runge-kutta methods in optimal control and the transformed adjoint system. *Numerische Mathematik*, 87:247–282, 2000.

[101] Francesco Topputo, Chen Zhang, et al. Survey of direct transcription for low-thrust space trajectory optimization with applications. In *Abstract and Applied Analysis*, volume 2014. Hindawi, 2014.

[102] Lorenzo Casalino, Guido Colasurdo, and Matteo Rosa Sentinella. Indirect optimization method for low-thrust interplanetary trajectories. In *30th International Electric Propulsion Conference, Florence*, 2007.

[103] Matthew Kelly. An introduction to trajectory optimization: How to do your own direct collocation. *SIAM Review*, 59(4):849–904, 2017.

[104] Matthew Peter Kelly. OptimTraj: Trajectory Optimization for Matlab, 12 2022.

[105] Marco Sagliano, David Seelbinder, and Stephan Theil. Spartan: Rapid trajectory analysis via pseudospectral methods. 2021.

[106] Alessia De Iuliis, Davide Costigliola, and Lorenzo Casalino. Optimal control for formation reconfiguration problems using pseudospectral methods.

[107] Simone D'Amico and Oliver Montenbruck. Proximity operations of formation-flying spacecraft using an eccentricity/inclination vector separation. *Journal of Guidance, Control, and Dynamics*, 29(3):554–563, 2006.

[108] Simone D'Amico. *Autonomous formation flying in low earth orbit*. PhD thesis, TU Delft, 2010.

[109] Gabriella Gaias, Simone D'Amico, and Jean-Sébastien Ardaens. Generalised multi-impulsive manoeuvres for optimum spacecraft rendezvous in near-circular orbit. *International Journal of Space Science and Engineering*, 3(1):68–88, 2015.

[110] Davide Costigliola and Lorenzo Casalino. Autonomous phasing maneuvers in near circular earth orbits.

[111] Andreas Wächter and Lorenz T Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106:25–57, 2006.