A WEIGHTED RESIDUAL FRAMEWORK FOR FORMULATION AND ANALYSIS OF DIRECT TRANSCRIPTION METHODS FOR OPTIMAL CONTROL

A Dissertation

by

BALJEET SINGH

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

December 2010

Major Subject: Aerospace Engineering

A WEIGHTED RESIDUAL FRAMEWORK FOR FORMULATION AND ANALYSIS OF DIRECT TRANSCRIPTION METHODS FOR OPTIMAL CONTROL

A Dissertation

by

BALJEET SINGH

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Raktim Bhattacharya
Committee Members,	John L. Junkins
	Srinivas R. Vadali
	Suman Chakravorty
	D.V.A.H.G. Swaroop
Head of Department,	Dimitris C. Lagoudas

December 2010

Major Subject: Aerospace Engineering

ABSTRACT

A Weighted Residual Framework for Formulation and Analysis of Direct Transcription Methods for Optimal Control. (December 2010)

Baljeet Singh, B.Tech., Indian Institute of Technology, Bombay, India;
M.Tech., Indian Institute of Technology, Bombay, India

Chair of Advisory Committee: Dr. Raktim Bhattacharya

In the past three decades, numerous methods have been proposed to transcribe optimal control problems (OCP) into nonlinear programming problems (NLP). In this dissertation work, a unifying weighted residual framework is developed under which most of the existing transcription methods can be derived by judiciously choosing test and trial functions. This greatly simplifies the derivation of optimality conditions and costate estimation results for direct transcription methods.

Under the same framework, three new transcription methods are devised which are particularly suitable for implementation in an adaptive refinement setting. The method of Hilbert space projection, the least square method for optimal control and generalized moment method for optimal control are developed and their optimality conditions are derived. It is shown that under a set of equivalence conditions, costates can be estimated from the Lagrange multipliers of the associated NLP for all three methods. Numerical implementation of these methods is described using B-Splines and global interpolating polynomials as approximating functions.

It is shown that the existing pseudospectral methods for optimal control can be formulated and analyzed under the proposed weighted residual framework. Performance of Legendre, Gauss and Radau pseudospectral methods is compared with the methods proposed in this research.

Based on the variational analysis of first-order optimality conditions for the op-

timal control problem, an posteriori error estimation procedure is developed. Using these error estimates, an h-adaptive scheme is outlined for the implementation of least square method in an adaptive manner. A time-scaling technique is described to handle problems with discontinuous control or multiple phases. Several real-life examples were solved to show the efficacy of the h-adaptive and time-scaling algorithm.

To my parents, Amarjeet Kaur and Sukhdev Singh.

ACKNOWLEDGMENTS

First of all, I would like to thank my advisor Dr. Raktim Bhattacharya for motivating me to embark on doctoral studies and for making this endeavor a memorable learning experience. His enthusiastic demeanor and broad research interests engaged me in a variety of projects during my stay at TAMU which greatly expanded my technical horizons. I thank him for the freedom he gave me in choosing my research direction and for the open and cordial atmosphere he provided in all our interactions and discussions.

I also thank my committee members Dr. John Junkins, Dr. Srinivas Vadali, Dr. Suman Chakravorty and Dr. Swaroop Darbha for their support. Thanks to Dr. John Valasek for his guidance in the Model-Based Aerospace Challenge. Special thanks to Mrs. Knabe Karen and Lisa Willingham for all the administrative help rendered during my stay at TAMU.

I am grateful to my roommates Surinder Pal and Navdeep Singh for making my stay at TAMU much like home. I thank my colleagues and friends, Hrishikesh, Shalom, Luis, Avinash, Roshmik, Parikshit, Sandeep, Abhishek, Xiaoli, Monika and Mrinal for their contributions to my learning experience at TAMU. I also thank all "SSA members", Jaspreet, Gaurav and Chanpreet for their support.

I would like to express my sincere gratitude to my teachers, Mr. Surinderpal Singh Gill of Shalimar Model High School, Ludhiana, and Mr. Narinder Kumar of SCD Government College, Ludhiana, for realizing my potential and instilling in me the much needed confidence at a very critical juncture of my life.

Finally, I offer my deepest gratitude to my parents and family; and last but not the least, my wife Kamaljeet. This dissertation would have not been possible without their love, support and encouragement.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

FIGURE Page

FIGURE Page

CHAPTER I

INTRODUCTION

Optimal control problems arise in a wide variety of applications areas of engineering, economics and sciences. In the fields of aerospace and robotics, optimal control problems can be defined for atmospheric reentry guidance, orbital maneuvers, attitude control, missile guidance, robot motion planning and many other applications. An optimal control problem seeks to determine the control input which drives a given dynamical system in such a manner that a prescribed performance criterion is minimized and the associated terminal and path constraints are satisfied. Optimal control theory strives to address such problems, and we begin this chapter with a brief historical account of major developments that has taken place in this field.

A. History of Developments in Optimal Control Theory

The origin of optimal control theory dates back to the 17^{th} century with the emergence the calculus of variations. It is believed that the calculus of variations started in 1662 when Pierre de Fermat (1601-1665) postulated his principle that the light rays follow the minimum time path [1, 2]. In the late 17^{th} century, Johann Bernoulli (1667-1748) challenged [3] his colleagues to solve the famous "brachistochrone" problem originally posed by Galileo Galilei (1629-1695) in 1638. This gave rise to further studies by a number of outstanding mathematicians such as Newton, Bernoulli, Leibnitz, Euler, Lagrange and Hamilton, marking the beginnings of optimal control theory.

Euler (1707-1783) and Lagrange (1736-1813) further developed the calculus of variations and gave the first-order necessary conditions, known as Euler-Lagrange

---

The journal model is *IEEE Transactions on Automatic Control.*

equations, for minimizing or maximizing a functional. Next, Legendre (1752-1833) and later Clebsch (1833-1872) looked at the second variation and postulated another condition for optimality known as the Legendre-Clebsch condition. Later Bolza (1857-1942) and Bliss (1876-1951) gave the calculus of variations its present rigorous form.

Based on the Hamilton-Jacobi theory in analytical mechanics and Bellman's principle of optimality, a new approach known as dynamic programming was proposed by Richard Bellman and his colleagues in the 1950's leading to the Hamilton-Jacobi-Bellman (HJB) equation [4]. The HJB equation is a partial differential equation which relates the optimal feedback control to the optimal cost to go function defined over the entire state space.

In the middle of 20^{th} century, Pontryagin developed the maximum principle [5] to extend the calculus of variations for handling control variable inequality constraints. The maximum principle is inherent in dynamic programming since the HJB equation includes finding the controls that minimize the Hamiltonian at each point in the state space.

The advent of commercial computers in the 1950's vitally transformed the field by enabling efficient numerical solutions of the optimal control problems. In the present day, there exist a variety of numerical methods for solving optimal control problems varying greatly in their approach and complexity.

B. A Review of Numerical Optimal Control Methods

Available numerical methods to solve optimal control problems can largely be divided into two categories: indirect methods and direct methods [6]. In an indirect method, the optimal control problem (OCP) is dualized by adjoining with the costates, also known as dual variables. The optimality conditions, also known as Euler-Lagrange

(EL) equations, are derived using the calculus of variations and Pontryagin's minimum principle, leading to a two-point boundary-value problem (TPBVP) [7, 8, 9]. Various numerical techniques are then used to solve the TPBVP [10]. In a direct method, the optimal control problem is discretized by parameterizing the controls, and frequently states as well, to transcribe the continuous time OCP into a finite-dimensional nonlinear programming problem (NLP) [11]. The NLP is then solved using numerical optimizers such as 'fmincon' in MATLAB, SNOPT [12], or NPSOL [13].

Direct methods have gained wide popularity in the last decade for many reasons. The foremost being that direct methods do not require analytical derivation of the EL equations, making it easier to automate the direct transcription process inside a software program. This has led to the development of many software tools, such as OTIS [14], SOCS [15], DIRCOL [16], NTG [17], DIDO [18], DIRECT [19], OPTRAGEN [20] and GPOPS [21], for direct transcription. Furthermore, experience has shown that direct methods have much larger radii of convergence than indirect methods, thus are more robust with respect to the inaccurate initial guess. Also, incorporating state and control constraints is much easier in direct methods. While the indirect methods provide high accuracy solutions, these methods are not popular in industry as the TPBVP has to be derived analytically and requires in-depth knowledge of optimal control theory. Also, solving the TPBVP can be extremely difficult due to its sensitivity to the unknown boundary conditions and the initial guess.

The direct methods can be further categorized as: 1) differentiation based (DB) and 2) integration based (IB) methods [22]. The two types differ in the way state dynamics $\dot{x} = f(x, u)$ is imposed. The differentiation based methods approximate the tangent bundle $\dot{x}$, while the integration based methods rely on approximating the integral of the vector field $f(x, u)$. The most popular among the DB methods are the pseudospectral (PS) methods [23, 24, 25, 26, 27]. The PS methods are

based on expanding the state and control variables in terms of global polynomials, which interpolate these functions at some specially chosen nodes. These nodes are zeros of orthogonal polynomials (or their derivatives) such as Legendre polynomials (Legendre–Gauss points) or Chebyshev polynomials (Chebyshev points). To deal with the problems having switches and discontinuities, the pseudospectral methods use the concept of PS knots [28] where the entire time interval is divided into a number of subinterval and the standard PS methodology is employed in each interval. The discontinuities can be enforced by knotting constraints and the unknown knot locations can be treated as parameters of optimization. The IB methods employ Runge-Kutta integration schemes to approximate the state dynamics, and the concept of phases is used to deal with non-smooth problems [6]. The phase boundary conditions define the continuity properties across adjoining phases. Here too the phase lengths can be made a part of the optimization process.

With regard to the satisfaction of optimality conditions, there is a fundamental difference between direct and indirect methods. The indirect methods are based on solving the necessary conditions derived by using the Minimum Principle, thus naturally satisfying the conditions of optimality. Direct methods are based on satisfying the Karush-Kuhn-Tucker (KKT) conditions associated with the NLP, which do not necessarily correspond to the optimality conditions of the OCP. Therefore, a direct method needs to be analyzed for the conditions under which such correspondence exists. Further, in the absence of any direct information about the costates, a costate estimation procedure needs to be derived for a direct method. The estimated costates can then be used to numerically verify the optimality by using the Minimum Principle. Also, these costate estimates can serve as initial guess for more accurate indirect methods.

The optimality verification and costate estimation results for most the direct

methods are available in the literature. Hager [29] presented the convergence analysis for Runge-Kutta based direct methods. It was noted that additional conditions on the coefficients of the integration scheme were required for a complete commutation between dualization and discretization. This discrepancy led Hager to design new Runge-Kutta methods for control applications. Stryk and Blurisch [30] showed the equivalence between the costates and the KKT multipliers of the NLP for the Hermite-Simpson method. Later, Williams [31] generalized these results for HD methods. Ross and Fahroo [32, 33] presented a similar result for Legendre pseudospectral method, deriving a set of closure conditions under which the costates can be estimated from the KKT multipliers. Williams [26] generalized the same result for the Jacobi pseudospectral method. Benson et al. [27] have shown equivalence between the discrete costates and the KKT multipliers for the Gauss pseudospectral method.

Adaptive mesh refinement algorithms have been reported in the optimal control literature. Betts and Huffman [34] selects the new grid points by solving an integer programming problem that minimizes the maximum discretization error by subdividing the current grid. The work of Binder et al. [35, 36] is based on using wavelet spaces to discretize the OCP to an NLP. They use wavelet analysis of the control profile for adaptive refinement. Jain and Tsiotras [37] presented a multi-resolution technique for trajectory optimization where interpolative error coefficients were used for adaptive mesh refinement. In the pseudospectral knotting method, Ross and Fahroo [38] introduce adaptivity using the concept of *free knots*, and by making them a part of optimization process.

C. Research Motivation and Objectives

This research work is motivated by the following two key considerations related to the direct transcription methods:

1. To devise novel methods and techniques to solve optimal control problems in a fast, efficient and robust manner.

2. To put forth a generalized mathematical framework for formulation and analysis of direct transcription methods.

From the pervious discussion, we see that there is a large body of work on the development and analysis of direct transcription methods. However, less work has been done to make these algorithms efficient and robust with respect to the complexity of a given problem. Optimal control problems in the fields of aerospace and robotics often exhibit solutions with discontinuities or corners. Solving these problems becomes difficult as the location of discontinuity or corner is not known before hand. Therefore, to capture the solution well using a standard numerical method, one has to approximate the unknown variables on a very high resolution grid, leading to higher computational cost in terms of both CPU time and memory. A possible solution is to locally refine the approximation in the regions of irregularities, requiring the approximating functions to exhibit local support. Another desirable feature would be the use of some special functions, like exponentials or sinusoids, that characterize the known local behavior of the system as approximations. In the current body of literature, most of the direct transcription methods use global approximations like polynomials and harmonics, or B-Splines. Global approximating functions provide no scope for local refinement. While B-Splines have local approximation property and scope for h-refinement, there is no direct mechanism for local p-refinement or inclusion

of any special functions as approximation. This motivates for the development of a new class of direct methods which are based on a more generic framework using function spaces.

As stated earlier, a systematic analysis of direct methods is required to obtain the costate estimates. This analysis has been performed for most of the direct methods and the derivations are available in the literature. However, since most of these derivations are quite involved and there are a variety of methods available, the corresponding costate estimation results are very difficult to comprehend. This motivates for the development of a single mathematical framework to analyze most of the direct methods.

D. Dissertation Overview

- Chapter II introduces many mathematical concepts that are relevant in the context of this dissertation work. First, a basic understanding of optimization concepts is developed. Next, the optimal control problem under consideration is formulated, and the associated first-order necessary conditions are derived. The concept of direct transcription is introduced to the reader using Euler discretization. Since the methods and techniques introduced in this dissertation make use of a number of numerical approximation concepts, many such schemes are discussed at length in this chapter. These topics include interpolating and orthogonal polynomials, B-Splines, partition of unity (PU) based approximations and numerical quadrature.

- In Chapter III, a generic method based on the weighted residual formulation of the trajectory constraints is considered for the direct transcription of optimal control problems. All trajectory variables are approximated in a general basis

expansion form. Analysis of primal-dual consistency is carried out by deriving the KKT conditions associated with the nonlinear programming problem and comparing them with the approximated first-order optimality conditions of the optimal control problem. A set of conditions are derived under which the indirect and direct approaches are equivalent.

- In this Chapter IV, the method of Hilbert space projection (MHSP) is formulated and analyzed. This method is a special case of the weighted residual method (WRM) where the test functions are chosen to be same as the trial functions approximating the state variables. A nonlinear programming problem is formulated for the original optimal control problem and a set of equivalence conditions are derived for costate estimation. Further, the numerical implementation of MHSP is described using B-Splines as approximating functions. Numerical convergence of MHSP is demonstrated using example problems.

- A least square method (LSM_{oc}) for direct transcription of optimal control problems is presented in Chapter V. This method is based on the L^2-minimization of the residual in state dynamics. The equivalence conditions for costate mapping are derived and a relationship between the costates and the KKT multipliers of the nonlinear programming problem is established. Further, numerical implementation of LSM_{oc} using B-Splines as approximating functions is described in detail. It is also shown that the numerical quadrature does not change the optimality and costate estimation results. Further, a polynomial version of LSM_{oc}, s-LSM_{oc} is derived by using the global Lagrange polynomials as test and trial functions. Example problems are solved to numerically demonstrate the convergence properties of LSM_{oc} and s-LSM_{oc}.

- Another special case of weighted residual formulation, the generalized moment

method for optimal control (GMM_{oc}), is developed and analyzed in Chapter VI. A polynomial version of GMM_{oc}, s-GMM_{oc} is derived by using the global Lagrange polynomials as test and trial functions. Numerical convergence of both GMM_{oc} and s-GMM_{oc} is demonstrated using example problems.

- Chapter VII investigates the relationship between many of the existing orthogonal collocation based methods and the generic weighted residual method formulated in Chapter III. While the primal approximation is straight forward, dual approximation results for Legendre, Radau and Gauss pseudospectral methods are derived under the unifying framework of WRM.

- Chapter VIII presents the performance comparison results for the existing and the proposed direct transcription methods. Comparisons are made between local and global methods as well as between different methods within each category of local and global methods.

- In Chapter IX, an adaptive algorithm to solve optimal control problems is developed using partition of unity approximations. The least square method is used as the underlying direct transcription method. An *a posteriori* error estimation procedure based on the residual analysis of the EL equations is proposed to capture the regions of irregularities in the solution. Based on this error estimation, an h-adaptive scheme is outlined for mesh refinement and a number of example problems are solved to demonstrate its efficiency.

- In Chapter X, a direct optimization algorithm to solve problems with discontinuous control is developed based on LSM_{oc}. To accommodate the discontinuities and corners, time domain is divided into a number of subintervals, each defining a control phase. Depending upon the problem in hand, a control type is

assigned to each phase. To deal with the unknown switching times, control phases are mapped on a computational domain with equal intervals and the state dynamics is appropriately scaled in each interval. The scaled problem is then discretized using B-Splines and transcribed to a nonlinear programming problem (NLP) using the LSM_{oc}. The NLP is solved and the solution is mapped back to the original time domain. A number of numerical examples are solved at the end of this chapter.

- Finally, Chapter XI includes a summary of dissertation contribution and concluding remarks.

CHAPTER II

MATHEMATICAL BACKGROUND

This chapter introduces many mathematical concepts that are relevant in the context of this dissertation work. First, a basic understanding of optimization concepts is developed. Next, the optimal control problem under consideration is formulated, and the associated first-order necessary conditions are derived. The concept of direct transcription is introduced to the reader using Euler discretization. Since the methods and techniques introduced in this dissertation make use of a number of numerical approximation concepts, many such topics are discussed in this chapter including interpolating and orthogonal polynomials, B-Splines, partition of unity (PU) based approximations and numerical quadrature.

A. Parameter Optimization

Parameter optimization deals with the problem of minimizing a scalar function of several variables subject to a number of equality or inequality constraints. In this section, we derive optimality conditions for three types of parameter optimization problems.

First, consider a multi-dimensional unconstrained minimization problem to find $\mathbf{x}^* \in \mathbb{R}^n$ such that the value of cost function $\Psi(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}$ is minimized. Applying Taylor expansion we can write,

$$\Psi(\mathbf{x}^* + \delta\mathbf{x}) = \Psi(\mathbf{x}^*) + \left.\frac{\partial \Psi(\mathbf{x})}{\partial \mathbf{x}}\right|_{\mathbf{x}^*} \delta\mathbf{x} + \delta\mathbf{x}^T \left.\frac{\partial^2 \Psi(\mathbf{x})}{\partial \mathbf{x}^2}\right|_{\mathbf{x}^*} \delta\mathbf{x} + \ldots \text{ higher order terms.} \tag{2.1}$$

For the point $\mathbf{x}^*$ to be a minima, $\Psi(\mathbf{x}^* + \delta\mathbf{x})$ should be greater than $\Psi(\mathbf{x}^*)$ for all possible values of variation $\delta\mathbf{x}$. This observation along with Eqn. (2.1) lead to the

following conditions for minima:

$$\Psi_{\mathbf{x}}(\mathbf{x}^*) = 0, \qquad \Psi_{\mathbf{xx}}(\mathbf{x}^*) > 0, \tag{2.2}$$

where a subscript variable denotes the partial derivative with respect to it. The same notation is used in all the subsequent treatment, i.e.

$$\mathcal{F}_{\mathbf{x}} \triangleq \frac{\partial \mathcal{F}}{\partial \mathbf{x}}, \qquad \mathcal{F}_{\mathbf{xu}} \triangleq \frac{\partial^2 \mathcal{F}}{\partial \mathbf{x} \partial \mathbf{u}}, \tag{2.3}$$

where $\mathbf{x}$ and $\mathbf{u}$ are vector variables. First condition in Eqn. (2.2) provides n number of equations which can theoretically solve for n number of unknown components of $\mathbf{x}^*$.

Next, we look at a constrained optimization problem in the form,

$$\min_{\mathbf{x}} J = \Psi(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}, \text{ subject to } \psi(\mathbf{x}) = 0 \in \mathbb{R}^m; \quad m < n. \tag{2.4}$$

This problem is solved by adjoining the equality constraint in Eqn. (2.4) to the cost function with a set of Lagrange multipliers [39] $\Lambda \in \mathbb{R}^m$, so that the modified problem is,

$$\min_{\{\mathbf{x},\Lambda\}} J' = \Psi(\mathbf{x}) + \Lambda^T \psi(\mathbf{x}), \text{ subject to } \psi(\mathbf{x}) = 0. \tag{2.5}$$

In a manner similar to the unconstrained case, the conditions for minima are obtained by setting the partial derivatives of J' with respect to both $\mathbf{x}$ and Λ equal to zero. So that,

$$J'_{\mathbf{x}} = \Psi_{\mathbf{x}}(\mathbf{x}^*) + \psi^T_{\mathbf{x}}(\mathbf{x}^*)\Lambda^* = 0, \qquad J'_{\Lambda} = \psi(\mathbf{x}^*) = 0. \tag{2.6}$$

The conditions in Eqn. (2.6) represent $n + m$ equations in terms of $n + m$ number of unknowns, thus making this system solvable for $\{\mathbf{x}^*, \Lambda^*\}$. However, second-order

conditions for minimality need to be checked, i.e.

$$\delta\mathbf{x}^T J'_{\mathbf{xx}}\delta\mathbf{x} = \delta\mathbf{x}^T \left(\Psi_{\mathbf{xx}}(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i \psi_{\mathbf{xx}}(\mathbf{x}^*) \right) \delta\mathbf{x} > 0 \quad \forall \{\delta\mathbf{x} \mid \psi_{\mathbf{x}}(\mathbf{x}^*)\delta\mathbf{x} = 0\}. \tag{2.7}$$

We see that in Eqn. (2.7), positive-definiteness of $J'_{\mathbf{xx}}$ is only required in the directions tangent to the constraint surface $\psi(\mathbf{x})$ at point $\mathbf{x}^*$.

Finally, we consider an optimization problem with inequality constraints as following:

$$\min_{\mathbf{x}} J = \Psi(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}, \text{ subject to } \psi(\mathbf{x}) \leq 0 \in \mathbb{R}^m; \quad m < n. \tag{2.8}$$

For this problem, we derive the conditions for optimality using slack variables and Lagrange multipliers. Slack variables are used to convert the inequality constraints to equality constraints. With a set of slack variables $\mathbf{s}\in\mathbb{R}^m$ and Lagrange multipliers $\Lambda \in \mathbb{R}^m$, the modified problem is defined as,

$$\min_{\{\mathbf{x},\Lambda,\mathbf{s}\}} J' = \Psi(\mathbf{x}) + \Lambda^T \left(\psi(\mathbf{x}) + \mathbf{s}\circ\mathbf{s}\right), \text{ subject to } \psi(\mathbf{x}) = 0. \tag{2.9}$$

Here '$\circ$' denotes the Hadamard product of two vectors, i.e. for $\mathbf{s}=[s_1\ s_2\ ...\ s_m]^T$, $\mathbf{s}\circ\mathbf{s} = [s_1^2\ s_2^2\ ...\ s_m^2]^T$. Analogously to the previous cases, we set the partial derivatives of J' with respect to $\mathbf{x}$,Λ and $\mathbf{s}$ equal to zero. So that,

$$J'_{\mathbf{x}} = \Psi_{\mathbf{x}}(\mathbf{x}^*) + \psi_{\mathbf{x}}^T(\mathbf{x}^*)\Lambda^* = 0, \tag{2.10}$$

$$J'_{\Lambda} = \psi(\mathbf{x}^*) + \mathbf{s}^* \circ \mathbf{s}^* = 0, \tag{2.11}$$

$$J'_{\mathbf{s}} = 2\mathbf{s}^* \circ \Lambda^* = 0. \tag{2.12}$$

The second-order condition is obtained as,

$$\delta\mathbf{x}^T J'_{\mathbf{xx}}\delta\mathbf{x} + \delta\mathbf{s}^T J'_{\mathbf{ss}}\delta\mathbf{s} > 0 \quad \forall \{\delta\mathbf{x}, \delta\mathbf{s} \mid \psi_{\mathbf{x}}(\mathbf{x}^*)\delta\mathbf{x} + 2\mathbf{s}\circ\delta\mathbf{s} = 0\}, \tag{2.13}$$

which implies,

$$\Rightarrow \delta\mathbf{x}^T \left(\Psi_{\mathbf{xx}}(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i \psi_{\mathbf{xx}}(\mathbf{x}^*) \right) \delta\mathbf{x} + 2\delta\mathbf{s}^T \Lambda \delta\mathbf{s} > 0. \tag{2.14}$$

There are two interesting cases to be considered. First, when the minima lies inside the constraint boundary. In this case, using Eqn. (2.12),

$$\Lambda = 0 \Rightarrow \begin{cases} \Psi_{\mathbf{x}}(\mathbf{x}^*) = & 0 \\ \Psi_{\mathbf{xx}}(\mathbf{x}^*) > & 0 \end{cases}, \tag{2.15}$$

which is equivalent to an unconstrained problem. In the second case, minima lies at the boundary. Here,

$$\mathbf{s} = 0 \Rightarrow \begin{cases} \Psi_{\mathbf{x}}(\mathbf{x}^*) + \psi_{\mathbf{x}}^T(\mathbf{x}^*)\Lambda^* = & 0 \\ \psi(\mathbf{x}^*) = & 0 \\ \delta\mathbf{x}^T[\Psi_{\mathbf{xx}}(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i \psi_{\mathbf{xx}}(\mathbf{x}^*)]\delta\mathbf{x} > & 0 \ \forall \ \{\delta\mathbf{x} \mid \psi_{\mathbf{x}}(\mathbf{x}^*)\delta\mathbf{x} = 0\} \\ \Lambda > & 0 \end{cases} \tag{2.16}$$

The conditions in Eqn. (2.16) are equivalent to the conditions for an optimization problem with an equality constraint as given in Eqn. (2.6) and Eqn. (2.7). The additional requirement on the value of Lagrange multiplier, $\Lambda > 0$, signifies that leaving the constraint boundary would increase the value of cost function.

B. Optimal Control Theory

Optimal control problems can be viewed as infinite-dimensional extension of parameter optimization problems. While the optimization variables in parameter optimization problems are points, optimal control problems deal with continuous functions. A standard optimal control problem consists of a dynamical system represented by a set of differential equations, associated terminal and path constraints, and a scalar

performance index. In the differential equations, the differentiated variables are called states and the undifferentiated variables are called controls. The objective is to find the control histories that drive the system from its initial state to the final state while minimizing the performance index and satisfying the path constraints. Thus, the unknowns for optimal control problems are curves and points.

Let us consider a fairly general optimal control problem stated in Bolza form as following: Determine the state-control pair $\{\mathbf{X}(\tau) \in \mathbb{R}^n, \mathbf{U}(\tau) \in \mathbb{R}^m; \tau \in [\tau_0, \tau_f]\}$ and time instances τ_0 and τ_f, that minimize the cost,

$$J = \Psi(\mathbf{X}(\tau_0), \mathbf{X}(\tau_f), \tau_0, \tau_f) + \int_{\tau_0}^{\tau_f} L(\mathbf{X}(\tau), \mathbf{U}(\tau), \tau) d\tau, \tag{2.17}$$

subject to the state dynamics,

$$\frac{d\mathbf{X}(\tau)}{d\tau} = \mathbf{f}(\mathbf{X}(\tau), \mathbf{U}(\tau), \tau) \in \mathbb{R}^n, \tag{2.18}$$

end-point state equality constraints,

$$K_0\mathbf{X}(\tau_0) = \mathbf{x}_0 \in \mathbb{R}^a, \quad K_1\mathbf{X}(\tau_f) = \mathbf{x}_f \in \mathbb{R}^{n-a}, \tag{2.19}$$

$$\psi(\mathbf{X}(\tau_0), \mathbf{X}(\tau_f), \tau_0, \tau_f) = 0 \in \mathbb{R}^p, \tag{2.20}$$

and path inequality constraints,

$$\mathbf{h}(\mathbf{X}(\tau), \mathbf{U}(\tau), \tau) \leq 0 \in \mathbb{R}^q. \tag{2.21}$$

Here K_0 and K_1 are constant matrices which split the boundary conditions between initial and final times.

For notational simplicity, most work in this dissertation considers a modified form of the above problem. We make the following observations for simplifications. First, The integral argument $L(\mathbf{X}(\tau), \mathbf{U}(\tau), \tau)$ in the cost function can be treated

a dynamic equation by appending the state vector $\mathbf{X}(\tau)$ with an additional state $\mathbf{Z}(\tau)$ and setting $\dot{\mathbf{Z}}(\tau) = L(\mathbf{X}(\tau), \mathbf{U}(\tau), \tau)$. By doing so, the integral term in the cost function is reduced to $\mathbf{Z}(\tau_f) - \mathbf{Z}(\tau_0)$ which gets assimilated into the terminal cost $\Psi(\mathbf{X}(\tau_0), \mathbf{X}(\tau_f), \tau_0, \tau_f)$. Second, the time dependence in $\mathbf{f}(\mathbf{X}(\tau), \mathbf{U}(\tau), \tau)$ and $L(\mathbf{X}(\tau), \mathbf{U}(\tau), \tau)$ can be removed by further taking τ as an additional state and setting $\dot{\tau} = 1$. Lastly, the time domain $\tau \in [\tau_0, \tau_f]$ can be mapped to a computational domain $t \in [0, 1]$ by using the transformation,

$$\tau(t) = (\tau_f - \tau_0)t + \tau_0. \tag{2.22}$$

Using Eqn. (2.22), we can write,

$$\frac{d\mathbf{X}(\tau(t))}{d\tau} = \frac{1}{(\tau_f - \tau_0)}\dot{\mathbf{x}}(t), \tag{2.23}$$

where an overdot denotes the derivative with respect to t. The computational interval is chosen to be $[0, 1]$ because methods presented in the subsequent chapters are based on approximations defined on the interval $[0, 1]$. Following these observations, we define the problem statement for this dissertation.

1. Optimal Control Problem Statement: $\mathcal{M}$

Consider the following optimal control problem in Mayer form and denote it as Problem $\mathcal{M}$: Find the state-control pair $\{\mathbf{x}(t) \in \mathbb{R}^n, \mathbf{u}(t) \in \mathbb{R}^m; t \in [0, 1]\}$, slack variable functions $\mathbf{s}(t) \in \mathbb{R}^q$ and time instances τ_0 and τ_f, that minimize the cost,

$$J = \Psi(\mathbf{x}(0), \mathbf{x}(1), \tau_0, \tau_f), \tag{2.24}$$

subject to the state dynamics,

$$\dot{\mathbf{x}}(t) = (\tau_f - \tau_0)\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \tag{2.25}$$

end-point state equality constraints,

$$K_0\mathbf{x}(0) = \mathbf{x}_0 \in \mathbb{R}^a, \quad K_1\mathbf{x}(1) = \mathbf{x}_f \in \mathbb{R}^{n-a}, \quad \psi(\mathbf{x}(0), \mathbf{x}(1), \tau_0, \tau_f) = 0 \in \mathbb{R}^p, \tag{2.26}$$

and mixed path constraints,

$$\mathbf{h}(\mathbf{x}(t), \mathbf{u}(t)) + \mathbf{s}(t) \circ \mathbf{s}(t) = 0 \in \mathbb{R}^q, \tag{2.27}$$

where,

$$\Psi : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \times \mathbb{R} \to \mathbb{R}, \qquad \mathbf{f} : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n,$$
$$\psi : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \times \mathbb{R} \to \mathbb{R}^p, \qquad \mathbf{h} : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^q.$$

are continuously differentiable with respect to their arguments. It is assumed that the optimal solution to the above problem exists, and at any time $t \in [0, 1]$, $\frac{\partial \overline{\mathbf{h}}}{\partial \mathbf{u}}$ has full rank, where $\overline{\mathbf{h}}$ is the active constraint set at time τ. Thus, the constraint qualifications required to apply the first-order optimality conditions are implicitly assumed.

2. Indirect Solution Approach

A classical approach to solve Problem $\mathcal{M}$ is to apply the principles of calculus of variations. In this setting, the minimization of a cost functional $J(\mathbf{x}(t))$ is achieved by requiring that the variation in cost vanishes for all possible first order variations in $\mathbf{x}(t)$, so that,

$$\delta J(\mathbf{x}^*(t), \delta\mathbf{x}(t)) = 0 \quad \forall \delta\mathbf{x}(t), \tag{2.28}$$

where $\mathbf{x}^*(t)$ is the minimizing solution. Using this principle and the theory of Lagrange multipliers for constrained optimization, a set of necessary first-order optimality conditions are derived for problem $\mathcal{M}$. This lead to a two-point boundary value problem

derived by using the augmented Hamiltonian $\mathcal{H}$ and the terminal cost $\mathcal{C}$ defined as,

$$\begin{aligned}\mathcal{H}(\mathbf{x},\mathbf{u},\lambda,\xi,\mathbf{s}) &= \lambda^T(t)(\tau_f-\tau_0)\mathbf{f}(\mathbf{x},\mathbf{u}) + \xi^T(t)[\mathbf{h}(\mathbf{x},\mathbf{u}) + \mathbf{s}(t)\circ\mathbf{s}(t)],\\ \mathcal{C}(\mathbf{x}(\tau_0),\mathbf{x}(\tau_f),\tau_0,\tau_f,\upsilon,\kappa_0,\kappa_1) &= \Psi(\mathbf{x}(0),\mathbf{x}(1),\tau_0,\tau_f) + \upsilon^T\psi(\mathbf{x}(0),\mathbf{x}(1),\tau_0,\tau_f)\\ &\quad + \kappa_0^T[K_0\mathbf{x}(0)-\mathbf{x}_0] + \kappa_1^T[K_1\mathbf{x}(1)-\mathbf{x}_f], \end{aligned} \tag{2.29}$$

where $\lambda(t) \in \mathbb{R}^n$ is the costate, and $\xi(t) \in \mathbb{R}^q$, $\upsilon \in \mathbb{R}^p$ and $\{\kappa_0 \in \mathbb{R}^a, \kappa_1 \in \mathbb{R}^{n-a}\}$ are the lagrange multipliers. The time dependence of state and control trajectories has been dropped for brevity. Problem $\mathcal{M}$ seeks to find the functions $\{\mathbf{x}(t),\mathbf{u}(t),\lambda(t),\xi(t),\mathbf{s}(t); t \in [0,1]\}$, vectors $\upsilon,\kappa_0,\kappa_1$ and time instances τ_0 and τ_f that satisfy the following conditions,

$$\begin{aligned} \dot{\mathbf{x}} - (\tau_f-\tau_0)\mathbf{f}(\mathbf{x},\mathbf{u}) = 0, \qquad & \mathcal{H}_{\mathbf{u}} = (\tau_f-\tau_0)\mathbf{f}_{\mathbf{u}}^T\lambda + \mathbf{h}_{\mathbf{u}}^T\xi = 0,\\ K_0\mathbf{x}(0)-\mathbf{x}_0 = 0, \qquad & \mathbf{h}(\mathbf{x},\mathbf{u}) + \mathbf{s}(t)\circ\mathbf{s}(t) = 0,\\ K_1\mathbf{x}(1)-\mathbf{x}_f = 0, \qquad & \mathcal{H}_{\mathbf{s}} = 2\xi(t)\circ\mathbf{s}(t) = 0,\\ \dot{\lambda} + \mathcal{H}_{\mathbf{x}} = \dot{\lambda} + (\tau_f-\tau_0)\mathbf{f}_{\mathbf{x}}^T\lambda + \mathbf{h}_{\mathbf{x}}^T\xi = 0, \qquad & \psi(\mathbf{x}(0),\mathbf{x}(1),\tau_0,\tau_f) = 0,\\ \{\lambda(t_0),\lambda(t_f)\} = \{-\mathcal{C}_{\mathbf{x}(\tau_0)},\mathcal{C}_{\mathbf{x}(\tau_f)}\}, \qquad & \{\mathcal{H}|_{t=0},\mathcal{H}|_{t=1}\} = \{\mathcal{C}_{\tau_0},-\mathcal{C}_{\tau_f}\}. \end{aligned} \tag{2.30}$$

The conditions in Eqn. (2.30), also known as Euler-Lagrange (EL) equations, constitute problem $\mathcal{M}_\lambda$.

For some problems, the control solution can not be obtained form EL equations alone. For such problems, Pontryagin's minimum principle is used which is based on minimizing the Hamiltonian $\mathcal{H}(t)$ globally by proper selection of admissible control $\mathbf{u}(t)$. Thus, if $\mathcal{U}$ is the set of admissible controls, then Pontryagin's minimum principle states that the optimal control $\mathbf{u}^*(t) \in \mathcal{U}$ is such that,

$$\mathcal{H}(\mathbf{x}^*,\mathbf{u}^*,\lambda^*,\xi^*,\mathbf{s}^*) \leq \mathcal{H}(\mathbf{x}^*,\mathbf{u},\lambda^*,\xi^*,\mathbf{s}^*), \quad \forall \mathbf{u}(t) \in \mathcal{U}. \tag{2.31}$$

3. Direct Transcription Approach

Another approach to solve problem $\mathcal{M}$ is to directly derive its finite-dimensional approximation in the form of an associated nonlinear programming problem $\mathcal{M}_\phi$. This is achieved by first approximating the state and control variables is a finite dimensional function space. Next, the integral constraints in $\mathcal{M}$ are approximated using numerical quadrature and the state dynamics is approximated by using some differentiation or integration based scheme.

In this section, an example of Euler discretization of problem $\mathcal{M}$ is presented to enforce the understanding of direct transcription process. In Euler transcription, the time domain $t \in [0, 1]$ is divided into a set of N intervals defined by the node points $0 = t_1 < t_2 < .. < t_N = 1$. The interval lengths are $\{h_i := t_{i+1} - t_i\}_{i=1}^{N-1}$. All trajectory variables are discretized at the node points so that,

$$\mathbf{x}(t_i) = \mathbf{x}_i, \quad \mathbf{u}(t_i) = \mathbf{u}_i, \quad \mathbf{s}(t_i) = \mathbf{s}_i; \quad i = 1, .., N-1. \tag{2.32}$$

The state dynamics is approximated using the Euler formula,

$$\dot{\mathbf{x}}(t_i) \approx \frac{\mathbf{x}_{i+1} - \mathbf{x}_i}{h_i} = (\tau_f - \tau_0)\mathbf{f}(\mathbf{x}_i, \mathbf{u}_i); \quad i = 1, .., N-1. \tag{2.33}$$

The boundary and path constraints in problem $\mathcal{M}$ are approximated as,

$$K_0\mathbf{x}_1 = \mathbf{x}_0, \quad K_1\mathbf{x}_N = \mathbf{x}_f, \quad \psi(\mathbf{x}_1, \mathbf{x}_N, \tau_0, \tau_f) = 0, \tag{2.34}$$

$$\mathbf{h}(\mathbf{x}_i, \mathbf{u}_i) + \mathbf{s}_i \circ \mathbf{s}_i = 0; \quad i = 1, .., N-1. \tag{2.35}$$

Similarly, the approximate cost is defined as,

$$J \approx \Psi(\mathbf{x}_1, \mathbf{x}_N, \tau_0, \tau_f). \tag{2.36}$$

Eqns. (2.32)-(2.36) transcribe problem $\mathcal{M}$ to a nonlinear programming problem $\mathcal{M}_\phi$

defined as following: Determine $\{\mathbf{x}_k \in \mathbb{R}^n, \mathbf{u}_k \in \mathbb{R}^m, \mathbf{s}_k \in \mathbb{R}^q\}_{k=1}^{N-1}$, $\nu_0 \in \mathbb{R}^a$, $\nu_1 \in \mathbb{R}^{n-a}$ and time instances τ_0 and τ_f, that minimize the cost,

$$\widehat{J} = \Psi(\mathbf{x}_1, \mathbf{x}_N, \tau_0, \tau_f), \tag{2.37}$$

subject to the constraints,

$$\frac{\mathbf{x}_{i+1} - \mathbf{x}_i}{h_i} = (\tau_f - \tau_0)\mathbf{f}(\mathbf{x}_i, \mathbf{u}_i), \tag{2.38}$$

$$K_0\mathbf{x}_1 = \mathbf{x}_0, \quad K_1\mathbf{x}_N = \mathbf{x}_f, \quad \psi(\mathbf{x}_1, \mathbf{x}_N, \tau_0, \tau_f) = 0, \tag{2.39}$$

$$\mathbf{h}(\mathbf{x}_i, \mathbf{u}_i) + \mathbf{s}_i \circ \mathbf{s}_i = 0, \tag{2.40}$$

where $i = 1, .., N-1$. Eqns. (2.37)-(2.40) define the direct approach to solve problem $\mathcal{M}$.

4. Mapping between Indirect and Direct Approaches

As seen in the previous sections, the indirect and direct approaches differ in the order in which approximation and dualization of the optimal control problem is carried out. In the indirect approach, the optimal control problem is dualized first and then the EL equations are approximated to find a numerical solution. In the direct approach the optimal control problem is approximated first and then the dualization is done in the process of solving the nonlinear programming problem (see Figure (1)). Further, in the direct approach, the costates are not the explicit variables of the problem. However, lagrange multipliers from the nonlinear programming problem are available in direct approach. Therefore, it is natural to investigate any possible relationship between the Lagrange multipliers of the nonlinear programming problem and the costates of the original optimal control problem. This relationship can be derived by analyzing the KKT conditions associated with the nonlinear programming problem

in the direct approach. We shall see in Chapter III that under certain conditions, complete mapping exists between direct and indirect solutions and costates can be estimated from the Lagrange multipliers of the nonlinear programming problem.

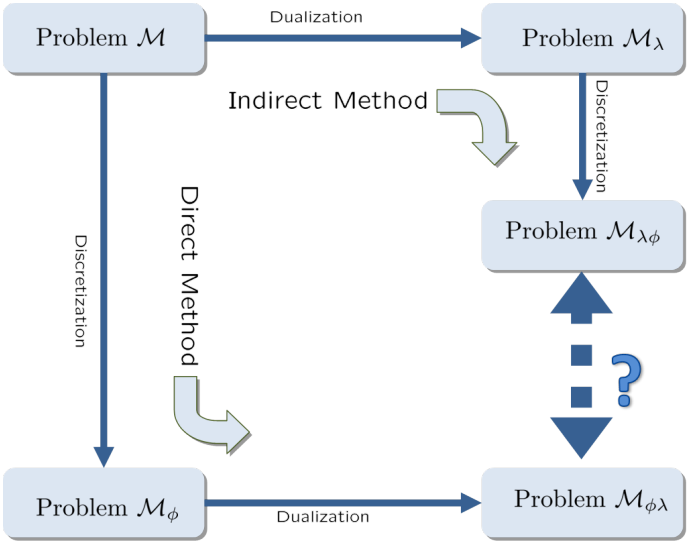


Fig. 1. Direct vs. indirect approach.

C. Numerical Approximation

1. Lagrange Interpolating Polynomials

A polynomial is a mathematical expression representing a weighted sum of powers in one or more variables. A polynomial in one variable with constant coefficients (weights) is given by,

$$P(t) = a_n t^n + a_{n-1} t^{n-1} + a_{n-2} t^{n-2} ... + a_0. \tag{2.41}$$

The individual powers of t, $\{t^n, t^{n-1}, ..., 1\}$, are called monomials. The highest power in a polynomial is called its order, or sometimes its degree.

Polynomials can be used for parametric approximation of more complicated curves and functions. Of particular interest is data interpolation, where polynomials are used to find interpolated values between discrete data points. Polynomial interpolation also forms the basis for algorithms in numerical quadrature, numerical ordinary differential equations and numerical optimal control methods.

Interpolation polynomials are based on the fact that given a set of N distinct data-points $\{(t_i, x_i)\}_{i=1}^{N}$, there exists a unique polynomial $P_N(t)$ of order $N-1$ such that,

$$P_N(t_i) = x_i; \quad i = 1, 2, .., N. \tag{2.42}$$

The polynomial $P_N(t)$ can be written in the basis form,

$$P_N(t) = \sum_{i=1}^{N} x_i \mathcal{L}_i(t), \tag{2.43}$$

where $\mathcal{L}_i$ are known as the Lagrange interpolation polynomials [40]. These polynomials can be expressed as,

$$\mathcal{L}_i(t) = \prod_{j=1, j\neq i}^{N} \frac{(t - t_j)}{(t_i - t_j)} \tag{2.44}$$

From Eqn. (2.44) we see that,

$$\mathcal{L}_i(t_j) = \delta_{ij} = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases} \tag{2.45}$$

Figure (2) shows Lagrange interpolating polynomials for a uniform grid of five points.

In the context of function approximation, polynomials are convenient to work with because they can be readily differentiated and integrated. Further, they ap-

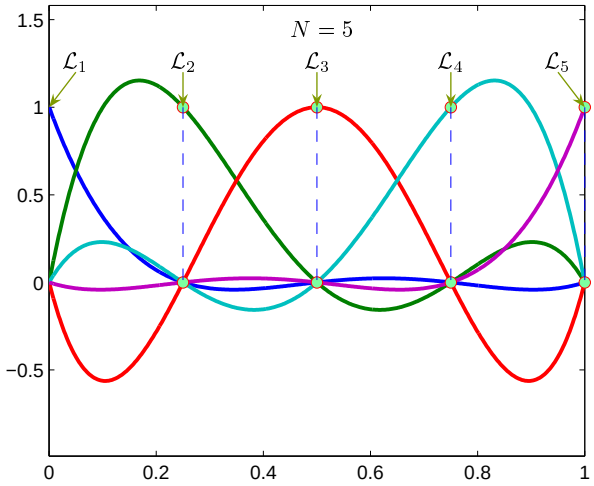


Fig. 2. Lagrange interpolating polynomials.

proximate continuous functions to any desired accuracy. That is, for any continuous function $f(t) : [t_i, t_f] \rightarrow \mathbb{R}$, there exists a polynomial $P_n(t)$ of sufficiently high order n such that $\|f(t) - P_n(t)\| < \epsilon \ \forall \ t \in \mathbb{R}$. Here ϵ is a pre-selected error bound. The proof comes from the famous Weierstrass approximation theorem [41]. In function approximation setting, a set of distinct node points are selected as $[t_i = t_0, t_1, t_2, ..., t_n = t_f]$, and $P_n(t)$ defines a polynomial which passes through the corresponding points $\{f(t_0), f(t_1), f(t_2), ..., f(t_n)\}$.

One might expect the quality of function approximation to increase with increasing order n of the polynomials used. However, it is observed that the relative distribution of the node points effects the approximation quality to a great extent. In particular, when equidistant node points are selected and the order n increases, the quality of polynomial approximation drops significantly at the boundaries resulting from large oscillations in those regions. This phenomenon is also known as Runge phenomenon. Fortunately, there exist sets of non-uniform node points that eliminate

the Runge phenomenon and can guarantee that the polynomial approximation error monotonically decreases as their number is increased. These node points are based on the roots of Legendre and Chebyshev polynomials and have the characteristic that the spacing between the support points is denser at the boundaries. Figure (3) demonstrates the Runge phenomenon and the effect of node distribution for polynomial approximations to the function $f(t) = \frac{1}{(1+15t^2)}$ using 20 node points. It clearly shows that the approximation using equidistant node points is quite poor near the boundaries, while the use of Chebyshev nodes mitigates this problem.

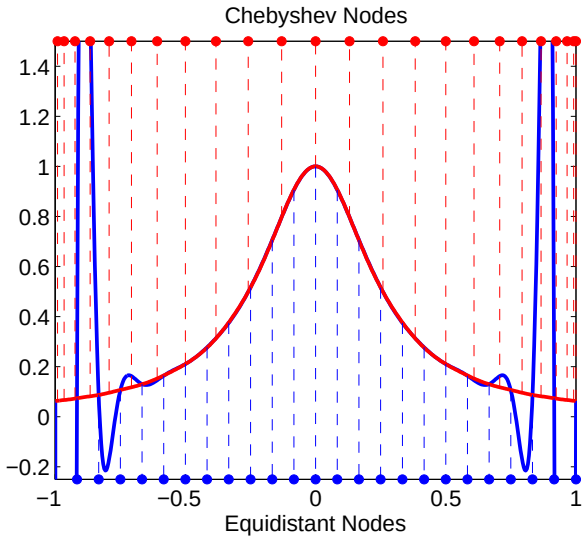


Fig. 3. Demonstration of Runge's phenomenon for 20 nodes.

2. Orthogonal Polynomials

The construction of an orthogonal family of polynomials first requires the definition of an inner product. Given a real interval $[t_1, t_2]$, functions $f(t), g(t) : [t_1, t_2] \to \mathbb{R}$ and a weight function $W(t) : [t_1, t_2] \to \mathbb{R}; W(t) \geq 0$, let an inner product $\langle \cdot, \cdot \rangle$ is defined

as,

$$\langle f(t), g(t) \rangle = \int_{t_1}^{t_2} f(t) W(t) g(t) dt. \tag{2.46}$$

Then, a sequence $\{P_i(t)\}_{i=0}^{N}$ is a sequence of orthogonal polynomials if,

1. $P_i(t)$ is a polynomial of degree i.

2. $\langle P_i(t), P_j(t) \rangle = \delta_{ij}$ (Kronecker Delta).

In numerical approximation, most commonly used orthogonal polynomial families are defined on interval $[-1, 1]$ and they differ in choice of weight function $W(t)$. Legendre polynomials , for example, use the standard L^2 inner product,

$$\langle f(t), g(t) \rangle = \int_{-1}^{1} f(t) g(t) dt; \qquad W(t) = 1. \tag{2.47}$$

Chebyshev polynomials of first kind are based on the inner product,

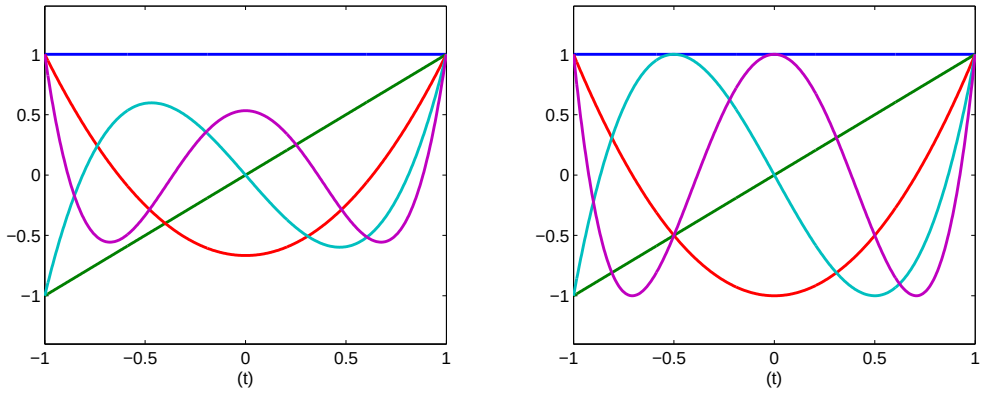
$$\langle f(t), g(t) \rangle = \int_{-1}^{1} \frac{f(t) g(t)}{\sqrt{1 - t^2}} dt; \qquad W(t) = \frac{1}{\sqrt{1 - t^2}}. \tag{2.48}$$

All orthogonal polynomial sequences have a number of elegant and fascinating properties. One of them is the existence of a recurrence formula relating any three consecutive polynomials in the sequence. This facilitates the construction of the whole family from its first two members. For Legendre polynomials ($L_j(t)$) and Chebyshev polynomials of first kind ($T_j(t)$), the recurrence relationships are,

$$(n+1) L_{n+1}(t) = (2n+1) \cdot t \cdot L_n(t) - n \cdot L_{n-1}(t); \qquad L_0(t) = 1, L_1(t) = t. \tag{2.49}$$

$$T_{n+1}(t) = 2 \cdot t \cdot T_n(t) - T_{n-1}; \qquad T_0(t) = 1, T_1(t) = t. \tag{2.50}$$

Figures 4(a) and 4(b) show first five members each from the family of Legendre and Chebyshev polynomials.



(a) (b)

Fig. 4. (a) Legendre polynomials, (b) Chebyshev polynomials of first kind.

3. Numerical Quadrature

In numerical analysis, a quadrature rule is an approximation of the definite integral of a function, usually stated as a weighted sum of function values at specified points within the domain of integration. The basic problem considered by numerical integration is to compute an approximate solution to a definite integral in the form,

$$\int_{t_i}^{t_f} f(t)dt \approx \sum_{k=1}^{N} w_k f(t_k); \quad t_i \leq t_1 < t_2 < .. < t_N \leq t_f, \tag{2.51}$$

where w_k's are called quadrature weights. If $f(t)$ is a smooth well-behaved function and the limits of integration are bounded, there are many methods of approximating the integral with arbitrary precision.

For an arbitrary set of unique points $\{t_k\}_{k=1}^{N}$, the exact quadrature approximation for polynomials of degree $N-1$ or less can be obtained by integrating Eqn. (2.43)

and using Eqn. (2.42),

$$\int_{t_i}^{t_f} P_N(t) = \sum_{k=1}^{N} P_N(t_k) \int_{t_i}^{t_f} \mathcal{L}_k(t), \tag{2.52}$$

Eqn. (2.52) implies,

$$w_k = \int_{t_i}^{t_f} \mathcal{L}_k(t), \tag{2.53}$$

The accuracy of the quadrature can be improved greatly by appropriately selecting the quadrature points. In interval $(-1, 1)$, one such selection is Gauss quadrature where N quadrature points are the Legendre-Gauss (LG) points, defined as the roots of the N^{th}-degree Legendre polynomial, $L_N(t)$ where,

$$L_N(t) = \frac{1}{2^N N!} \frac{d^N}{dt^N}[(t^2 - 1)^N]. \tag{2.54}$$

The corresponding Gauss quadrature weights are then found by the formula,

$$w_k|_{k=1}^{N} = \frac{2}{(1 - t_k^2)[\dot{L}_N(t_k)]^2} \tag{2.55}$$

where $\dot{L}_N$ is the derivative of the N^{th}-degree Legendre polynomial. The Gauss quadrature is exact for all polynomials of degree $2N - 1$ or less.

Another set of points are Legendre-Gauss-Radau (LGR) points, which lie on the interval $[-1, 1)$. LGR quadrature is accurate upto $2N - 2$ degree polynomials, one less than the LG points as one of the points is forced to lie at the boundary. The N LGR points are defined as the roots of $L_N(t) + L_{N-1}(t)$. The corresponding weights are,

$$w_1 = \frac{2}{N^2}, \quad w_k|_{k=2}^{N} = \frac{1}{(1 - t_k^2)[\dot{L}_{N-1}(t_k)]^2} \tag{2.56}$$

A third set of points are Legendre-Gauss-Lobatto (LGL) points, which lie on the

interval $[-1, 1]$. As two of the points are forced to lie at both boundaries, the LGL quadrature scheme is accurate upto $2N - 3$ degree polynomials. The N LGL points are the roots of $(1 - t^2)\dot{L}_{N-1}(t)$. The corresponding weights for the LGL points are,

$$w_k|_{k=1}^{N} = \frac{2}{N(N-1)[L_{N-1}(t_k)]^2} \tag{2.57}$$

An example of various quadrature nodes and corresponding weights for $N = 10$ is shown in Figure (5).

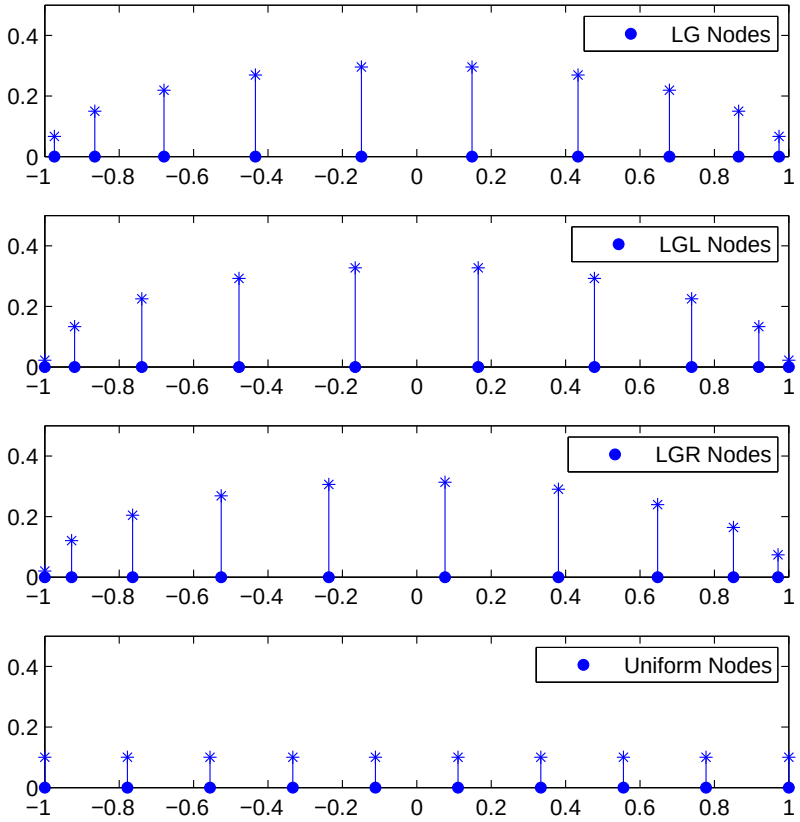


Fig. 5. Distribution of nodes and corresponding weights for LG, LGL, LGR and uniform nodes.

4. B-Splines

A B-Spline is a function defined on the interval $[t_0, t_f]$ of the real line, composed of segments of polynomials that are stitched at predefined break points, satisfying a given degree of smoothness. The break points are a strictly increasing set of points $\{t_i\}_{i=0}^{N}$ such that $t_0 = t_0 < t_1 < ... < t_N = t_f$. The number of continuous derivatives across the breakpoints defines the order of smoothness. An order of smoothness s_i at a breakpoint t_i implies that the curve is C^{s_i-1} continuously differentiable at t_i. Given the number of subintervals (N), the order of each polynomial segment (r) and the order of smoothness (s) at the breakpoints, a B-Spline curve $y(t)$ is represented in the basis form as,

$$y(t) = \sum_{k=1}^{N_c} \alpha_k B^{k,r}(t),$$

where α_k are the free parameters, $N_c = N(r-s)+s$ is the number of free parameters or the degrees of freedom of $y(t)$. To construct the basis functions $B^{k,r}(t)$, we define a knot vector. A knot vector Γ is a non-decreasing sequence containing breakpoints with a multiplicity of $(r-s)$ at the interior breakpoints. The multiplicity of end points $\{t_0, t_N\}$ is r,

$$\Gamma = [c_1, .., c_{(N_c+r)}] = [\overbrace{t_0, .., t_0}^{r-times} ... \overbrace{t_i, .., t_i}^{(r-s)-times} ... \overbrace{t_N, .., t_N}^{r-times}].$$

The basis functions $B^{k,r}(t)$ are defined by a recurrence relationship [42],

$$B^{k,0}(t) = \begin{cases} 1, & \text{if } c_k \leq t < c_{k+1} \\ 0, & \text{otherwise} \end{cases}$$

$$B^{k,r}(t) = \frac{t - c_k}{c_{k+r+1} - c_k} B^{k,r-1}(t) + \frac{c_{k+r} - t}{c_{k+r} - c_{k+1}} B^{k+1,r-1}(t). \tag{2.58}$$

The B-Spline basis functions defined by Eqn. (2.58) are continuous to a specified degree, have local support, and are linearly independent. A comprehensive list of B-Spline properties can be found in Ref. [42]. Figure (6) provides an example of a B-Spline and its basis functions with $N = 4, r = 4, s = 3$.

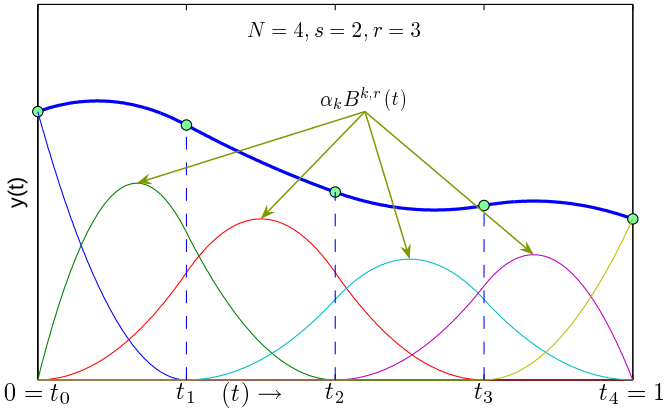


Fig. 6. Spline curve as a combination of B-Splines.

5. The Partition of Unity Paradigm

Partition of unity is a paradigm in which local approximations can be blended together to form a global approximation with a specified order or continuity. The global domain $[0, 1]$ is partitioned into overlapping subdomains $\{\Omega_i\}_{i=1}^n$, each having a compactly supported weight function $W_i(t)$ which is strictly zero outside Ω_i and has the property,

$$\sum_{i=1}^{n} W_i(t) = 1, \quad t \in [0, 1]. \tag{2.59}$$

Each subdomain Ω_i centers a corresponding node i and a local approximating function $f_i(t)$. The global approximation is obtained as,

$$\widehat{\mathbf{z}}(t) = \sum_{i=1}^{n} f_i(t) W_i(t). \tag{2.60}$$

When the local approximation $f_i(t)$ is constructed from a set of basis functions, known as *extrinsic* basis, we get,

$$f_i(t) = \sum_{j=1}^{n_i} \psi_{ij}(t) a_{ij}, \tag{2.61}$$

$$\widehat{\mathbf{z}}(t) = \sum_{i=1}^{n} \sum_{j=1}^{n_i} \psi_{ij}(t) W_i(t) a_{ij}. \tag{2.62}$$

PU spaces can be constructed using any of the following three major frameworks, (i) kernel approximation, (ii) moving least square approximation (MLS) and (iii) GLO-MAP (global local orthogonal polynomial mapping). The kernel approximation and the MLS approximation methods are based on mesh-free construction while the GLO-MAP is constructed using a mesh.

The first kernel approximation appeared in the smooth particle hydrodynamics (SPH) method [43, 44], and through improvements lead to the development of reproducing kernel particle method (RKPM) [45]. The idea of moving least squares (MLS)[46] was first presented by Shepard [46]. Duarte and Oden [47], and later Babuška and Melenek [48] introduced the partition of unity (PU) based approximations.

a. Reproducing Kernel Particle Method

Kernel approximation methods are based on convolution of a window function with the original function. So that on a domain $t \in [0, 1]$,

$$\widehat{\mathbf{z}}(t) = \int_0^1 \mathbf{w}(\tau - t)\mathbf{z}(\tau)d\tau \tag{2.63}$$

The window function $\mathbf{w}(t)$ is essentially an approximation of the dirac-delta function. When the continuous time convolution is replaced with a numerical quadrature, we obtain a particle method. This idea becomes apparent in the following development of reproducing kernel particle method from the continuous time reproducing kernel method.

The RKPM is derived based on Eqn. (2.63) by introducing a correction function to impose reproducibility conditions on the approximation. By reproducibility we mean that the approximation can exactly represent a set of predefined functions called the *intrinsic* basis. Let $\mathbf{p}(t)$ be the *intrinsic* basis so that,

$$\widehat{\mathbf{z}}(t) = \mathbf{z}(t) = \mathbf{p}(t)^T\mathbf{a}; \quad \forall \mathbf{z}(t) \in \mathbf{p}(t), \tag{2.64}$$

$$\mathbf{p}(t) = \left[\begin{array}{cccccccc} 1, & t, & t^2, & ... & sin(t), & cos(t), & ... & , \text{etc.} \end{array} \right]^T. \tag{2.65}$$

The coefficient vector $\mathbf{a}$ is determined form the reproducibility conditions. From Eqn. (2.64) we can write,

$$\int_0^1 \mathbf{p}(\tau)\mathbf{w}(\tau - t)\mathbf{z}(\tau)d\tau = \left[\int_0^1 \mathbf{p}(\tau)\mathbf{w}(\tau - t)\mathbf{p}^T(\tau)d\tau\right]\mathbf{a}. \tag{2.66}$$

Substituting the value of $\mathbf{a}$ form Eqn. (2.66) into $\widehat{\mathbf{z}}(t) = \mathbf{p}(t)^T\mathbf{a}$, we get,

$$\widehat{\mathbf{z}}(t) = \int_0^1 C(t, \tau)\mathbf{w}(\tau - t)\mathbf{z}(\tau)d\tau, \tag{2.67}$$

where,

$$C(t,\tau) = \mathbf{p}^T(t)\left[\int_0^1 \mathbf{p}(\tau)\mathbf{w}(\tau - t)\mathbf{p}^T(\tau)d\tau\right]^{-1}\mathbf{p}(\tau). \tag{2.68}$$

Using numerical quadrature to evaluate integrals in Eqn. (2.67) yields the particle form,

$$\widehat{\mathbf{z}}(t) = \sum_{i=1}^{n} C(t,t_i)\mathbf{w}(t_i - t)\mathbf{z}(t_i)\omega_i, \tag{2.69}$$

where ω_i are the quadrature weights, and $C(t,t_i)$ is computed as,

$$C(t,t_i) = \mathbf{p}^T(t)\left[\sum_{i=1}^{n} \mathbf{p}(t_i)\mathbf{w}(t_i - t)\mathbf{p}^T(t_i)\omega_i\right]^{-1}\mathbf{p}(t_i). \tag{2.70}$$

Form Eqn. (2.69), $\widehat{\mathbf{z}}(t)$ can be written in a basis form with Kronecker-delta property as following,

$$\widehat{\mathbf{z}}(t) = \sum_{i=1}^{n} \phi_i(t)\mathbf{z}(t_i), \quad \phi_i(t) = C(t,t_i)\mathbf{w}(t_i - t)\omega_i. \tag{2.71}$$

The approximation defined by Eqn. (2.71) can exactly reproduce the members of the *intrinsic* basis $\mathbf{p}(t)$.

b. Moving Least Square Approximations

The MLS approximation is based on constructing an approximation from a distribution of node points. Like RKPM, MLS also has an *intrinsic* basis $\mathbf{p}(t)$. The local approximation around a point $\tau \in [0,1]$ evaluated at a point $t \in [0,1]$ is given by,

$$\widehat{\mathbf{z}}(\tau,t) = \mathbf{p}^T(t)\mathbf{a}(\tau), \tag{2.72}$$

where $\mathbf{p}(t)$ has the same form as in Eqn. (2.65). The coefficient vector $\mathbf{a}(\tau)$ is a function of the "moving time" τ and is obtained by a locally weighted least square fit

on the nodal values $\{\mathbf{z}_i = \mathbf{z}(t_i), t_i \in [0,1]\}_{i=1}^n$. Given a weight function $\mathbf{w}(t-\tau)$ and the inner-product defined as,

$$\langle f(t), g(t)\rangle_\tau = \sum_{i=1}^{n} f(t_i)g(t_i)\mathbf{w}(t_i-\tau), \tag{2.73}$$

the minimum norm solution for $\mathbf{a}(\tau)$ based on Eqn. (2.73) is obtained as,

$$\mathbf{a}(\tau) = M^{-1}(\tau)\sum_{i=1}^{n}\mathbf{z}_i\mathbf{p}(t_i)\mathbf{w}(t_i-\tau), \tag{2.74}$$

where

$$M(\tau) = \sum_{i=1}^{n}\mathbf{p}^T(t_i)\mathbf{p}(t_i)\mathbf{w}(t_i-\tau). \tag{2.75}$$

Using Eqn. (2.72),

$$\widehat{\mathbf{z}}(t) = \mathbf{p}^T(t)\mathbf{a}(t) = \mathbf{p}^T(t)M^{-1}(t)\sum_{i=1}^{n}\mathbf{z}_i\mathbf{p}(t_i)\mathbf{w}(t_i-t), \tag{2.76}$$

$$= \sum_{i=1}^{n}\mathbf{p}^T(t)M^{-1}(t)\mathbf{p}(t_i)\mathbf{w}(t_i-t)\mathbf{z}_i. \tag{2.77}$$

Using Eqn. (2.77), $\widehat{\mathbf{z}}(t)$ can be written as,

$$\widehat{\mathbf{z}}(t) = \sum_{i=1}^{n}\phi_i(t)\mathbf{z}(t_i), \tag{2.78}$$

$$\phi_i(t) = \mathbf{p}^T(t)M^{-1}(t)\mathbf{p}(t_i)\mathbf{w}(t_i-t). \tag{2.79}$$

Interestingly, from Eqn. (2.68), Eqn. (2.71), Eqn. (2.75) and Eqn. (2.79) we see that MLS approximation is equivalent to the RKPM if the quadrature weights in the RKPM are unity.

The condition of partition of unity in Eqn. (2.59) is equivalent to the reproducibility of constant function by MLS approximation. It can been seen from Eqn.

(2.79),

$$\mathbf{p}(t) = 1, \quad \sum_{i=1}^{n} \phi_i(t) = \sum_{i=1}^{n} \frac{\mathbf{w}(t_i - t)}{\sum_{i=1}^{n} \mathbf{w}(t_i - \tau)} = 1. \tag{2.80}$$

Thus, we see that both RKPM and MLS approximations form a PU space. The order of continuity of the global approximation in Eqn. (2.60) depends upon the smoothness order of $W_i(t)$, which in turn is defined by the smoothness order of $\mathbf{w}(t)$ in Eqn. (2.79). The reproducing property of global approximation depends upon both *intrinsic* and *extrinsic* basis chosen for Eqn. (2.79) and Eqn. (2.60) respectively.

c. Global Local Orthogonal Polynomial Mapping

A PU based global approximation can also be constructed from the ideas of GLOMAP (global local orthogonal polynomial mapping) presented in Refs. [49, 50]. Let us assume,

A1 $\mathcal{T} = \{t_0, \cdots, t_n\}$ is a uniform grid with spacing h and $0 < t_i < t_{i+1} < 1$. Uniform grid is assumed for simplicity. It is not a limitation of the proposed approach. Let us define interval $\mathcal{I}_i = [t_i, t_{i+1}]$, $i = 0, 1, \cdots, n-1$.

A2 $\mathcal{F} = \{f_0(t), \cdots, f_n(t)\}$ is a set of continuous functions that approximates the global function $\mathbf{z}(t)$ at points $t_i \in \mathcal{T}$.

A3 $\mathcal{W}$ is a set of continuous functions in the interval $[-1, 1]$.

We define a non-dimensional local coordinate $\tau_i \in [-1, 1]$ as $\tau_i \stackrel{\Delta}{=} (t - t_i)/h$, centered on the i^{th} vertex $t = t_i$. Given $\mathcal{F}$ and weighting function $W(\tau_i) \in \mathcal{W}$, the weighted average approximation is defined as,

$$\bar{\mathbf{z}}_i(t) = W(\tau_i) f_i(t) + W(\tau_{i+1}) f_{i+1}(t), \tag{2.81}$$

for $0 \leq \tau_i, \tau_{i+1} < 1$ and $t \in [t_i, t_{i+1}]$. The weighting function $W(\tau_i)$ is used to blend or average the two adjacent preliminary local approximations $f_i(t)$ and $f_{i+1}(t)$. The global function is given by the expression,

$$\widehat{\mathbf{z}}(t) = \sum_{i=0}^{n} W(\tau_i) f_i(t); \;\; t \in [0,1], \;\; \tau_i \in [-1,1]. \tag{2.82}$$

The preliminary approximations $f_i(t) \in \mathcal{F}$ are completely arbitrary, as long as they are smooth and represents the local behavior of $\mathbf{z}(t)$ well. There exists a choice of weighting function that will guarantee piecewise global continuity while leaving the freedom to fit local data by any desired local functions. In Refs. [49, 50], it is shown that if the weighting functions of Eqn. (2.81) satisfy the following boundary value problem, then the weighted average approximation in Eqn. (2.81) form an m^{th}-order continuous globally valid model with complete freedom in the choice of the local approximations in $\mathcal{F}$. These conditions characterize the set $\mathcal{W}$. That is,

$$\mathcal{W} = \left\{ \begin{array}{ll} & W(0) = 1, \; W(1) = 0, \\ W(\tau): & W^{(k)}(0) = 0, \; W^{(k)}(1) = 0, \; k = 0, \cdots, m \\ & W(\tau) + W(\tau - 1) = 1, \; \forall \; \tau, \; -1 \leq \tau \leq 1 \end{array} \right\}$$

where $W^{(k)} \triangleq \frac{d^k W}{d\tau^k}$. The conditions can be summarized as follows.

1. The first derivative of the weighting function must have an m^{th}-order osculation with $W(0) = 1$ at the centroid of its respective local approximation.

2. The weighting function must have an $(m+1)^{th}$-order zero at the centroid of its neighboring local approximation.

3. The sum of two neighboring weighting functions must be unity over the entire closed interval between their corresponding adjacent local functional approximations.

Table I. GLOMAP weight functions for a given order of continuity

Continuity	Weight Function: $W(\tau), \forall\tau \in \{-1 \leq \tau \leq 1\},\ \eta \stackrel{\Delta}{=} \lvert\tau\rvert$
0	$1-\eta$
1	$1-\eta^2(3-2\eta)$
2	$1-\eta^3(10-15\eta+6\eta^2)$
3	$1-\eta^4(35-84\eta+70\eta^2-20\eta^3)$
$\vdots$	$\vdots$
m	$1-\eta^{m+1}\left\{\frac{(2m+1)!(-1)^m}{(m!)^2}\sum_{k=0}^{m}\frac{(-1)^k}{2m-k+1}\binom{m}{k}\eta^{m-k}\right\}$

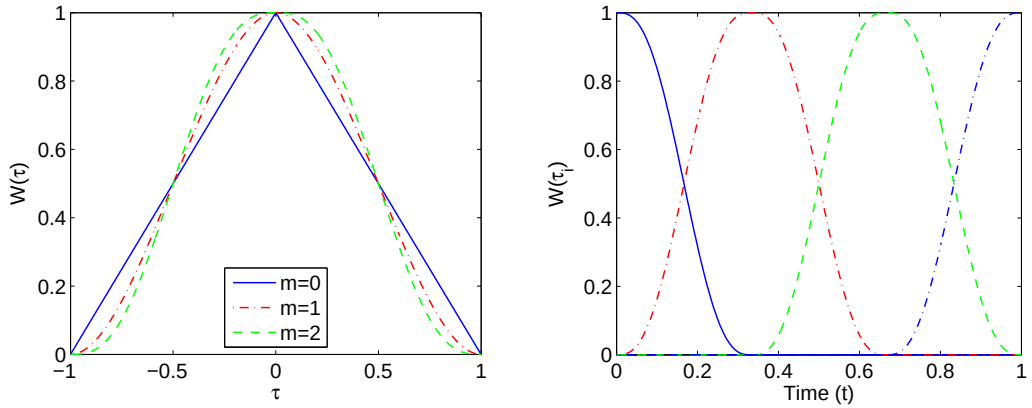
If the weighting function is assumed to be a polynomial in the independent variable τ, then adopting the procedure listed in Ref. [50], the lowest order weight function (for $m = 1$) can be shown to be

$$\begin{aligned} W(\tau) &= \left\{ \begin{array}{l} 1-\tau^2(3+2\tau),\ \ -1 \leq \tau < 0 \\ 1-\tau^2(3-2\tau),\ 0 \leq \tau \leq 1 \end{array} \right\} \\ &= 1-\tau^2(3-2|\tau|) \end{aligned} \tag{2.83}$$

The weighting functions obtained by solving the boundary value problem for increasing value of m are listed in Table I, and are shown in Figure (7(a)). Figure (7(b)) shows the GLOMAP weighting functions for second-order continuity ($m = 2$) on a uniform grid of four points.

D. The Method of Weighted Residuals

The method of weighted residuals (MWR) is a general mathematical framework to solve a large class of ordinary and partial differential equations. The basic idea is to



(a) $W(\tau)$ for the first three orders of continuity.

(b) Weighting functions for second-order continuity.

Fig. 7. GLOMAP weighting functions $w(\tau_i)$

first approximate the unknown solution as a linear combination of a priory selected basis functions, also known as trial functions. Then a weighted integral formulation is used to transform the differential constraints into algebraic constraints, the solution to which determine the unknown coefficients.

In a general setting, we look at the following problem:

$$\dot{\mathbf{x}}(t) - \mathbf{f}(\mathbf{x}(t), t) = 0;\ \mathbf{x}(t) \in \mathbb{R}^n, \qquad \mathbf{b}(\mathbf{x}(0), \mathbf{x}(t_f)) = 0, \qquad t \in [0, t_f]. \tag{2.84}$$

Here $\mathbf{f}(\cdot)$ can be nonlinear. In MWR, a set of basis function $\{\phi_i\}_{i=1}^N$ is chosen to define an approximating function space $\mathbb{V} := span\{\phi_1, .., \phi_N\}$, and $\mathbf{x}(t)$ is approximated as,

$$\widehat{\mathbf{x}}(t) = \sum_{k=1}^{N} \alpha_k \phi_k(t) \in \mathbb{V};\ \alpha_k \in \mathbb{R}^n. \tag{2.85}$$

Because $\widehat{\mathbf{x}}(t)$ is an approximation, residuals $\mathbf{R_i}(t)$ and $\mathbf{R_b}$ will be generated when

$\widehat{\mathbf{x}}(t)$ is substituted for $\mathbf{x}(t)$ in Eqn. (2.84),

$$\mathbf{R_i}(\alpha_k, \phi_k(t)) = \dot{\widehat{\mathbf{x}}}(t) - \mathbf{f}(\widehat{\mathbf{x}}(t), t);\ t \in [0, t_f], \qquad \mathbf{R_b} = \mathbf{b}(\widehat{\mathbf{x}}(0), \widehat{\mathbf{x}}(t_f)). \tag{2.86}$$

The interior residual $\mathbf{R_i}$ is forced to zero in a weighted average sense over an entire domain. Mathematically, this condition can be expressed with the inner product,

$$\langle W_j(t), \mathbf{R_i}(\alpha_k, \phi_k(t), t)\rangle = \int_0^{t_f} W_j^T(t)\mathbf{R_i}(\alpha_k, \phi_k(t))dt = 0; \quad j = 1, 2, .., N-1. \tag{2.87}$$

Here $W_j(t)$ are a pre-selected set of mutually independent weighting, or testing functions constituting a function space $\mathbb{P} := span\{W_1, .., W_N\}$. From the functional analysis perspective, Eqn. (2.87) represents the setting of the projection of residual $\mathbf{R_i}$ on space $\mathbb{V}$ to zero. The boundary residual is set to zero as,

$$\mathbf{R_b} = \mathbf{b}(\widehat{\mathbf{x}}(0), \widehat{\mathbf{x}}(t_f)) = 0. \tag{2.88}$$

Eqns. (2.86) and (2.88) constitute N number of equations in N number of unknown α_k's, hence the system is theoretically solvable.

The choice of trial functions differentiates finite-element and finite-difference methods from spectral methods. Finite-element and finite-difference methods use trial functions as local polynomials of low order, while spectral methods employ global polynomials of high order. Similarly, choice of test functions and the way boundary conditions are implemented define a number of special cases of MWR which are known as separate methods in their own terms. We consider following four such methods and describe the choice of test and trial functions for each method.

1. Tau method

   Tau method uses the test functions same as trial functions. The approximation

$\widehat{\mathbf{x}}(t)$ does not satisfy the boundary condition explicitly, so that,

$$\widehat{\mathbf{x}}(t) = \sum_{k=1}^{N} \alpha_k \phi_k(t), \qquad \mathbf{b}(\widehat{\mathbf{x}}(0), \widehat{\mathbf{x}}(t_f)) = 0, \tag{2.89}$$

$$\int_0^{t_f} \phi_j(t) \mathbf{R_i}(\alpha_k, \phi_k(t)) dt = 0; \quad j = 1, 2, .., N-1. \tag{2.90}$$

2. Galerkin method

   In Galerkin method, test functions are chosen to be same as trial functions. Further, trial functions are chosen in such a manner that the boundary conditions are explicitly satisfied. So that,

$$\widehat{\mathbf{x}}(t) = \phi_0(t) + \sum_{k=1}^{N} \alpha_k \phi_k(t), \qquad \mathbf{b}(\phi_0(0), \phi_0(t_f)) = 0, \tag{2.91}$$

$$\int_0^{t_f} \phi_j(t) \mathbf{R_i}(\alpha_k, \phi_k(t)) dt = 0; \quad j = 1, 2, .., N. \tag{2.92}$$

3. Least-square method

   In this method, test function are chosen as,

$$W_j = \frac{\partial \mathbf{R_i}}{\partial \alpha_j}, \tag{2.93}$$

   and trial functions are chosen in such that the boundary conditions are explicitly satisfied. So that,

$$\widehat{\mathbf{x}}(t) = \phi_0(t) + \sum_{k=1}^{N} \alpha_k \phi_k(t), \qquad \mathbf{b}(\phi_0(0), \phi_0(t_f)) = 0, \tag{2.94}$$

$$\int_0^{t_f} \frac{\partial \mathbf{R_i}}{\partial \alpha_j}^T \mathbf{R_i}(\alpha_k, \phi_k(t)) dt = 0; \quad j = 1, 2, .., N. \tag{2.95}$$

4. Generalized moment method

In this method, test function are chosen as the derivatives of trial functions [51],

$$\widehat{\mathbf{x}}(t) = \sum_{k=1}^{N} \alpha_k \phi_k(t), \qquad \mathbf{b}(\widehat{\mathbf{x}}(0), \widehat{\mathbf{x}}(t_f)) = 0, \tag{2.96}$$

$$\int_0^{t_f} \dot{\phi}_j(t) \mathbf{R_i}(\alpha_k, \phi_k(t)) dt = 0; \quad j = 2, .., N. \tag{2.97}$$

5. Subdomain method

   In this method, the domain $[0, t_f]$ is split into N subdomains (intervals) $\{\Omega_j\}_{j=1}^N$ and the test functions are selected as,

$$W_j(t) = \begin{cases} 1 & t \in \Omega_j \\ 0 & \text{otherwise} \end{cases} \tag{2.98}$$

   The solution is obtained using,

$$\int_{\Omega_j} \mathbf{R_i}(\alpha_k, \phi_k(t)) dt = 0; \quad j = 1, 2, .., N. \tag{2.99}$$

6. Collocation method

   In collocation methods, the residual $\mathbf{R_i}$ is set to zero at a pre-defined set of nodes $\{t_j\}_{j=1}^N$. The test functions in this case are,

$$W_j(t) = \delta(t - t_j), \qquad \mathbf{R_i}(\alpha_k, \phi_k(t_j)) = 0. \tag{2.100}$$

CHAPTER III

WEIGHTED RESIDUAL FORMULATION FOR DIRECT OPTIMAL CONTROL

Advancement in the numerical algorithms to solve complex nonlinear programming problems and the availability of ever increasing computational power of modern-day computers has made direct transcription technique extremely popular for solving optimal control problems. In the past three decades, numerous methods have been proposed to transcribe optimal control problems to nonlinear programming problems. While the variety in these methods provide abundant techniques to cater for specific classes of problems, it also poses difficulties in understanding the underlying mechanism which differentiates these methods from one another and is responsible for their unique properties. Therefore, it is highly desired to have a unifying mathematical framework for the formulation and analysis of direct transcription methods.

In this direction, work has been done to generalize Hermite differentiation based methods [31]. For a class of pseudospectral methods, a framework based on generic Jacobi polynomials is proposed [52], with a further generalization using nonclassical orthogonal and weighted polynomial interpolation [26].

In this chapter, a generic method for the direct transcription of optimal control problems is considered. This method, denoted as WRM, is based on the weighted residual formulation of the state dynamics and path constraints associated with an optimal control problem. All trajectory variables are approximated in a general basis expansion form. A higher level analysis of primal-dual consistency is carried out without using any numerical quadrature scheme in the nonlinear programming problem formulation. This amounts to the derivation of KKT conditions associated with the nonlinear programming problem and their comparison with the approximated first-order optimality conditions of the optimal control problem. Application of numerical

quadrature changes the structure of the nonlinear programming problem, which is taken into account in the subsequent chapters where special cases are considered for the implementation of WRM.

We begin the formulation of WRM with a recap of the optimal control problem defined in Chapter II Section B.1: Find the state-control pair $\{\mathbf{x}(t) \in \mathbb{R}^n, \mathbf{u}(t) \in \mathbb{R}^m; t \in [0,1]\}$, slack variable functions $\mathbf{s}(t) \in \mathbb{R}^q$ and time instances τ_0 and τ_f, that minimize the cost,

$$J = \Psi(\mathbf{x}(0), \mathbf{x}(1), \tau_0, \tau_f), \tag{3.1}$$

subject to the state dynamics,

$$\dot{\mathbf{x}}(t) = (\tau_f - \tau_0)\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \tag{3.2}$$

end-point state equality constraints,

$$K_0\mathbf{x}(0) = \mathbf{x}_0 \in \mathbb{R}^a, \quad K_1\mathbf{x}(1) = \mathbf{x}_f \in \mathbb{R}^{n-a}, \quad \psi(\mathbf{x}(0), \mathbf{x}(1), \tau_0, \tau_f) = 0 \in \mathbb{R}^p, \tag{3.3}$$

and mixed path constraints,

$$\mathbf{h}(\mathbf{x}(t), \mathbf{u}(t)) + \mathbf{s}(t) \circ \mathbf{s}(t) = 0 \in \mathbb{R}^q, \tag{3.4}$$

where, $\Psi : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \times \mathbb{R} \to \mathbb{R}$, $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$, $\psi : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \times \mathbb{R} \to \mathbb{R}^p$, $\mathbf{h} : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^q$.

A. Direct Transcription Formulation

The first step in any direct transcription method is to define a finite dimensional approximation for all the unknown trajectory variables. In WRM, state, control and slack variable trajectories are approximated in finite-dimensional function spaces

spanned by possibly different sets of a priory chosen basis functions. So that, for $t \in [0,1]$,

$$\mathbb{V}_{\mathbf{x}} := \text{span}\{\phi_j^{\mathbf{x}}(t)\}_{j=1}^{N_{\mathbf{x}}}, \quad \mathbb{V}_{\mathbf{u}} := \text{span}\{\phi_r^{\mathbf{u}}(t)\}_{r=1}^{N_{\mathbf{u}}}, \quad \mathbb{V}_{\mathbf{s}} := \text{span}\{\phi_p^{\mathbf{s}}(t)\}_{p=1}^{N_{\mathbf{s}}}. \tag{3.5}$$

$\mathbb{V}_{\mathbf{x}}, \mathbb{V}_{\mathbf{u}}$ and $\mathbb{V}_{\mathbf{s}}$ are equipped with an inner product,

$$\langle \mathbf{p}, \mathbf{q} \rangle = \int_0^1 \mathbf{p}^T(t)\mathbf{q}(t)dt; \quad \mathbf{p}(t), \mathbf{q}(t) \in \mathbb{V}_{\mathbf{x}} \cup \mathbb{V}_{\mathbf{u}} \cup \mathbb{V}_{\mathbf{s}}, \tag{3.6}$$

and the state, control and slack variable trajectories are approximated as,

$$\mathbf{x}(t) \approx \widehat{\mathbf{x}}(t) = \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \phi_k^{\mathbf{x}}(t) \in \mathbb{V}_{\mathbf{x}}, \tag{3.7}$$

$$\dot{\widehat{\mathbf{x}}}(t) = \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \dot{\phi}_k^{\mathbf{x}}(t), \tag{3.8}$$

$$\mathbf{u}(t) \approx \widehat{\mathbf{u}}(t) = \sum_{k=1}^{N_{\mathbf{u}}} \beta_k \phi_k^{\mathbf{u}}(t) \in \mathbb{V}_{\mathbf{u}}, \tag{3.9}$$

$$\mathbf{s}(t) \approx \widehat{\mathbf{s}}(t) = \sum_{k=1}^{N_{\mathbf{s}}} \varsigma_k \phi_k^{\mathbf{s}}(t) \in \mathbb{V}_{\mathbf{s}}, \tag{3.10}$$

Having defined the approximations for the unknown variables, the next step is to approximate the state dynamics and path constraints. Motivated by the formulation presented in Chapter II Section D, the state dynamics is imposed as a weighted residual form,

$$\int_0^1 W_j^T(\widehat{\mathbf{x}}(t), \widehat{\mathbf{u}}(t))[(\tau_f - \tau_0)\mathbf{f}(\widehat{\mathbf{x}}(t), \widehat{\mathbf{u}}(t)) - \dot{\widehat{\mathbf{x}}}(t)]dt + \frac{1}{2}K_0^T \nu_0 \phi_j^{\mathbf{x}}(0) + \frac{1}{2}K_1^T \nu_1 \phi_j^{\mathbf{x}}(1) = 0;$$
$$j = 1, 2, .., N_{\mathbf{x}}, \tag{3.11}$$

with the boundary constraints,

$$K_0\widehat{\mathbf{x}}(0) - \mathbf{x}_0 = 0, \quad K_1\widehat{\mathbf{x}}(1) - \mathbf{x}_f = 0, \tag{3.12}$$

$$\psi(\widehat{\mathbf{x}}(0), \widehat{\mathbf{x}}(1), \tau_0, \tau_f) = 0. \tag{3.13}$$

Here W_j are the test functions spanning a projection space,

$$\mathbb{P}_x := span\{W_1, .., W_{N_\mathbf{x}}\}. \tag{3.14}$$

Notice that the state approximation as defined in Eqn. (3.7) does not satisfy the boundary conditions explicitly. Therefore, Eqn. (3.12) represents a set of n equality constraints in terms of the components of α_k's.

Approximation defined in Eqn. (3.11) is different form the one in Eqn. (2.87) in two ways. First, two extra terms are introduced containing unknown variables ν_0 and ν_1. Second, the number of weighted residual constraints are $N_\mathbf{x}$ while boundary conditions are not explicitly satisfied. At a first glance, it appears that this would result in a over constrained system of equations. However, formulation presented here is specifically for optimal control problems which are inherently under constrained. Therefore, the degrees of freedom from the unknown control variables alow us to impose $N_\mathbf{x}$ weighted integral constraints. The purpose of adding terms containing ν_0 and ν_1 will be clarified in the context of least-square method for optimal control. At this stage, these terms are simply carried forward in all the analysis presented in this chapter.

Path constraints in Eqn. (3.4) are also imposed in a weighted residual from as,

$$\int_0^1 V_p^T[\mathbf{h}(\widehat{\mathbf{x}}(t), \widehat{\mathbf{u}}(t)) + \widehat{\mathbf{s}}(t) \circ \widehat{\mathbf{s}}(t)] = 0, \quad p = 1, 2, .., N_\mathbf{s}, \tag{3.15}$$

where, V_p's are the associated test functions constituting a projection space,

$$\mathbb{P}_s := span\{V_1, .., V_{N_\mathbf{s}}\}. \tag{3.16}$$

Next, the approximations defined in section are used to transcribe problem $\mathcal{M}$ into a nonlinear programming problem $\mathcal{M}_\phi$.

B. Nonlinear Programming Problem: $\mathcal{M}_\phi$

Function approximation of trajectory variables using Eqn. (3.7), Eqn. (3.9) and Eqn. (3.10), combined with the weighted residual formulation as in Eqn. (3.11), Eqn. (3.12), Eqn. (3.13) and Eqn. (3.15), transcribe problem $\mathcal{M}$ into a finite dimensional nonlinear programming problem, denoted as Problem $\mathcal{M}_\phi$. For the subsequent treatment, we denote the approximate state dynamics as,

$$\widehat{\mathbf{f}}(\alpha_k, \beta_k, \phi_k^\mathbf{x}(t), \phi_k^\mathbf{u}(t)) = \mathbf{f}(\widehat{\mathbf{x}}(t), \widehat{\mathbf{u}}(t)).$$

Using similar notation for all other function expressions, Problem $\mathcal{M}_\phi$ is to determine $\{\alpha_k \in \mathbb{R}^n\}_{k=1}^{N_\mathbf{x}}, \{\beta_k \in \mathbb{R}^m\}_{k=1}^{N_\mathbf{u}}, \{\varsigma_k \in \mathbb{R}^q\}_{k=1}^{N_\mathbf{s}}$, $\nu_0 \in \mathbb{R}^a$, $\nu_1 \in \mathbb{R}^{n-a}$ and time instances τ_0 and τ_f, that minimize the cost,

$$\widehat{J} = \widehat{\Psi}(\alpha_k, \phi_k^\mathbf{x}(0), \phi_k^\mathbf{x}(1), \tau_0, \tau_f), \tag{3.17}$$

subject to the constraints,

$$\int_0^1 [W_j^T(\alpha_k, \beta_k, \phi_k^\mathbf{x}, \phi_k^\mathbf{u}, \tau_0, \tau_f)][(\tau_f - \tau_0)\widehat{\mathbf{f}}(\alpha_k, \beta_k, \phi_k^\mathbf{x}, \phi_k^\mathbf{u}) - \sum_{k=1}^{N_\mathbf{x}} \alpha_k \dot{\phi}_k^\mathbf{x}]dt$$
$$+\frac{1}{2}K_0^T\nu_0\phi_j^\mathbf{x}(0) + \frac{1}{2}K_1^T\nu_1\phi_j^\mathbf{x}(1) = 0, \tag{3.18}$$

$$K_0 \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \phi_k^{\mathbf{x}}(0) - \mathbf{x}_0 = 0, \quad K_1 \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \phi_k^{\mathbf{x}}(1) - \mathbf{x}_f = 0, \tag{3.19}$$

$$\widehat{\psi}(\alpha_k, \phi_k^{\mathbf{x}}(0), \phi_k^{\mathbf{x}}(1), \tau_0, \tau_f) = 0, \tag{3.20}$$

$$\int_0^1 V_p^T [\widehat{\mathbf{h}}(\alpha_k, \beta_k, \phi_k^{\mathbf{x}}, \phi_k^{\mathbf{u}}) + \sum_{k=1}^{N_{\mathbf{s}}} \varsigma_k \phi_k^{\mathbf{s}} \circ \sum_{l=1}^{N_{\mathbf{s}}} \varsigma_l \phi_l^{\mathbf{s}}] = 0, \tag{3.21}$$

where $j = 1, .., N_x$ and $p = 1, .., N_s$. Problem $\mathcal{M}_\phi$ constituting Eqns. (3.17) to (3.21) needs to be further discretized for numerical implementation. All the integral expressions in $\mathcal{M}_\phi$ need to be approximated using sone numerical quadrature scheme. However, at this stage we keep problem $\mathcal{M}_\phi$ in the integral form and follow through the derivation of the associated KKT first-order necessary conditions.

C. Derivation of KKT Conditions: $\mathcal{M}_{\phi\lambda}$

The solution to a nonlinear program satisfies a set of first-order optimality conditions called the Karush-Kuhn-Tucker (KKT) conditions. The KKT conditions corresponding to the $\mathcal{M}_\phi$ are obtained from the Lagrangian formed by adjoining the cost function with the constraint equations. For brevity, we use $\widehat{\mathbf{f}}$ to denote $\widehat{\mathbf{f}}(\alpha_k, \beta_k, \phi_k^{\mathbf{x}}, \phi_k^{\mathbf{u}})$. Using similar notation for all other variables, we write,

$$\begin{aligned} J' = & \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j^T \left[\int_0^1 W_j^T [(\tau_f - \tau_0)\widehat{\mathbf{f}} - \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \dot{\phi}_k^{\mathbf{x}}] dt + \frac{1}{2} K_0^T \nu_0 \phi_j^{\mathbf{x}}(0) + \frac{1}{2} K_1^T \nu_1 \phi_j^{\mathbf{x}}(1) \right] \\ & + \sum_{p=1}^{N_{\mathbf{s}}} \zeta_p^T \int_0^1 V_p^T [\widehat{\mathbf{h}} + \sum_{k=1}^{N_{\mathbf{s}}} \varsigma_k \phi_k^{\mathbf{s}} \circ \sum_{l=1}^{N_{\mathbf{s}}} \varsigma_l \phi_l^{\mathbf{s}}] dt \\ & + \mu_0^T (K_0 \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \phi_k^{\mathbf{x}}(0) - \mathbf{x}_0) + \mu_1^T (K_1 \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \phi_k^{\mathbf{x}}(0) - \mathbf{x}_0) + \widehat{\Psi} + \eta^T \widehat{\psi}, \end{aligned} \tag{3.22}$$

where $\gamma_j \in \mathbb{R}^n$, $\mu_0 \in \mathbb{R}^a$, $\mu_1 \in \mathbb{R}^{n-a}$, $\eta \in \mathbb{R}^p$ and $\zeta_j \in \mathbb{R}^q$ are the KKT multipliers associated with the constraints given by Eqn. (3.18), Eqn. (3.19) and Eqn. (3.21) respectively. The KKT first-order necessary conditions are obtained by setting the

derivatives of J' with respect to the unknowns $\{\alpha_i, \beta_i, \varsigma_i, \nu_0, \nu_1, \gamma_i, \mu_0, \mu_1, \eta, \zeta_i, \tau_0, \tau_f\}$ equal to zero. We have for $i = 1, .., N_x$,

$$\begin{aligned}\frac{\partial J'}{\partial \alpha_i} = &\int_0^1 \sum_{j=1}^{N_{\mathbf{x}}} [(\tau_f - \tau_0)\widehat{\mathbf{f}}_{\mathbf{x}}^T W_j \phi_i^{\mathbf{x}} - W_j \dot{\phi}_i^{\mathbf{x}}]\gamma_j dt \\ &+ \mu_0 K_0 \phi_i^{\mathbf{x}}(0) + \mu_1 K_1 \phi_i^{\mathbf{x}}(1) + \int_0^1 \sum_{p=1}^{N_{\mathbf{s}}} \widehat{\mathbf{h}}_{\mathbf{x}}^T V_p \zeta_p \phi_i^{\mathbf{x}} dt + \int_0^1 \sum_{j=1}^{N_{\mathbf{x}}} \mathbf{r}^T W_{\mathbf{x}j} \phi_i^{\mathbf{x}} \gamma_j dt \\ &+ [\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}_{\mathbf{x}(0)}^T \eta]\phi_i^{\mathbf{x}}(0) + [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)}^T \eta]\phi_i^{\mathbf{x}}(1) = 0. \end{aligned} \tag{3.23}$$

Using integration by parts, we write,

$$\int_0^1 W_j(t)\dot{\phi}_i^{\mathbf{x}}(t)dt = W_j(1)\phi_i^{\mathbf{x}}(1) - W_j(0)\phi_i^{\mathbf{x}}(0) - \int_0^1 \dot{W}_j(t)\phi_i^{\mathbf{x}}(t)dt, \tag{3.24}$$

Using Eqn. (3.23), Eqn. (3.24) and re-arranging, we get,

$$\begin{aligned}\frac{\partial J'}{\partial \alpha_i} = &\int_0^1 \sum_{j=1}^{N_{\mathbf{x}}} [(\tau_f - \tau_0)\widehat{\mathbf{f}}_{\mathbf{x}}^T W_j \phi_i^{\mathbf{x}} + \dot{W}_j \phi_i^{\mathbf{x}}]\gamma_j dt - \sum_{j=1}^{N_{\mathbf{x}}} [W_j(1)\phi_i^{\mathbf{x}}(1) - W_j(0)\phi_i^{\mathbf{x}}(0)]\gamma_j \\ &+ \mu_0 K_0 \phi_i^{\mathbf{x}}(0) + \mu_1 K_1 \phi_i^{\mathbf{x}}(1) + \int_0^1 \sum_{p=1}^{N_{\mathbf{s}}} \widehat{\mathbf{h}}_{\mathbf{x}}^T V_p \zeta_p \phi_i^{\mathbf{x}} dt + \int_0^1 \sum_{j=1}^{N_{\mathbf{x}}} \mathbf{r}^T W_{\mathbf{x}j} \phi_i^{\mathbf{x}} \gamma_j dt \\ &+ [\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}_{\mathbf{x}(0)}^T \eta]\phi_i^{\mathbf{x}}(0) + [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)}^T \eta]\phi_i^{\mathbf{x}}(1) = 0. \end{aligned} \tag{3.25}$$

Re-arranging further,

$$\begin{aligned}\frac{\partial J'}{\partial \alpha_i} = &\int_0^1 [(\tau_f - \tau_0)\widehat{\mathbf{f}}_{\mathbf{x}}^T \sum_{j=1}^{N_{\mathbf{x}}} W_j \gamma_j + \sum_{j=1}^{N_{\mathbf{x}}} \dot{W}_j \gamma_j + \sum_{p=1}^{N_{\mathbf{s}}} \widehat{\mathbf{h}}_{\mathbf{x}}^T V_p \zeta_p]\phi_i^{\mathbf{x}} dt + \int_0^1 \sum_{j=1}^{N_{\mathbf{x}}} \mathbf{r}^T W_{\mathbf{x}j} \phi_i^{\mathbf{x}} \gamma_j dt \\ &+ [\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}_{\mathbf{x}(0)}^T \eta + \mu_0 K_0 + \sum_{j=1}^{N_{\mathbf{x}}} W_j(0)\gamma_j]\phi_i^{\mathbf{x}}(0) \\ &+ [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)}^T \eta + \mu_1 K_1 - \sum_{j=1}^{N_{\mathbf{x}}} W_j(1)\gamma_j]\phi_i^{\mathbf{x}}(1) = 0. \end{aligned} \tag{3.26}$$

also,

$$\frac{\partial J'}{\partial \mu_0} = K_0 \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \phi_k^{\mathbf{x}}(0) - \mathbf{x}_0 = 0, \tag{3.27}$$

$$\frac{\partial J'}{\partial \mu_1} = K_1 \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \phi_k^{\mathbf{x}}(1) - \mathbf{x}_f = 0, \tag{3.28}$$

$$\frac{\partial J'}{\partial \beta_r} = \int_0^1 [(\tau_f - \tau_0)\widehat{\mathbf{f}}_{\mathbf{u}}^T \sum_{j=1}^{N_x} W_j \gamma_j + \widehat{\mathbf{h}}_{\mathbf{u}}^T \sum_{p=1}^{N_s} V_p \zeta_p] \phi_r^{\mathbf{u}} dt + \int_0^1 \sum_{j=1}^{N_{\mathbf{x}}} \mathbf{r}^T W_{\mathbf{u}j} \phi_r^{\mathbf{u}} \gamma_j dt = 0, \tag{3.29}$$

$$\frac{\partial J'}{\partial \gamma_i} = \int_0^1 W_i^T [(\tau_f - \tau_0)\widehat{\mathbf{f}} - \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \dot{\phi}_k^{\mathbf{x}}] dt + \frac{1}{2} K_0^T \nu_0 \phi_i^{\mathbf{x}}(0) + \frac{1}{2} K_1^T \nu_1 \phi_i^{\mathbf{x}}(1) = 0, \tag{3.30}$$

$$\frac{\partial J'}{\partial \varsigma_q} = 2 \int_0^1 [\sum_{k=1}^{N_{\mathbf{s}}} \varsigma_k \phi_k^{\mathbf{s}} \circ \sum_{p=1}^{N_{\mathbf{s}}} V_p \zeta_p] \phi_q^{\mathbf{s}} dt = 0 \tag{3.31}$$

$$\frac{\partial J'}{\partial \nu_0} = K_0 \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j \phi_j(0) = 0, \tag{3.32}$$

$$\frac{\partial J'}{\partial \nu_1} = K_1 \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j \phi_j(1) = 0, \tag{3.33}$$

$$\frac{\partial J'}{\partial \eta} = \widehat{\psi} = 0, \tag{3.34}$$

$$\frac{\partial J'}{\partial \zeta_q} = \int_0^1 V_q^T [\widehat{\mathbf{h}} + \sum_{k=1}^{N_{\mathbf{s}}} \varsigma_k \phi_k^{\mathbf{s}} \circ \sum_{l=1}^{N_{\mathbf{s}}} \varsigma_l \phi_l^{\mathbf{s}}] = 0, \tag{3.35}$$

$$\frac{\partial J'}{\partial \tau_0} = -\int_0^1 \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j^T W_j^T \widehat{\mathbf{f}} dt + \int_0^1 \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j^T \frac{\partial W_j}{\partial \tau_0} \mathbf{r} dt + [\widehat{\Psi}_{\tau_0} + \eta^T \widehat{\psi}_{\tau_0}] = 0, \tag{3.36}$$

$$\frac{\partial J'}{\partial \tau_f} = \int_0^1 \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j^T W_j^T \widehat{\mathbf{f}} dt + \int_0^1 \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j^T \frac{\partial W_j}{\partial \tau_f} \mathbf{r} dt + [\widehat{\Psi}_{\tau_f} + \eta^T \widehat{\psi}_{\tau_f}] = 0, \tag{3.37}$$

where $i = 1, .., N_{\mathbf{x}}$, $q = 1, .., N_s$ and $r = 1, .., N_u$. Eqn. (3.26) through Eqn. (3.37) constitute the KKT conditions for Problem $\mathcal{M}_\phi$. Next, we derive the approximated first-order optimality conditions for problem $\mathcal{M}$, denoted as $\mathcal{M}_{\lambda\phi}$.

D. Weighted Residual Approximation of First-Order Optimality Conditions: $\mathcal{M}_{\lambda\phi}$

This section describes in the weighted residual approximation of the first-order necessary conditions $\mathcal{M}_\lambda$ derived in Chapter II Section B.2. This step is not necessary for the implementation of a direct method, but provides a mean to compare the direct and indirect solution of an optimal control problem. This comparison would results in the derivation of costate estimation results.

Conditions in problem $\mathcal{M}_\lambda$ are reproduced here for a quick reference. For,

$$\begin{aligned}\mathcal{C} =&\Psi(\mathbf{x}(0),\mathbf{x}(1),\tau_0,\tau_f)+\upsilon^T\psi(\mathbf{x}(0),\mathbf{x}(1),\tau_0,\tau_f)\\&+\kappa_0^T[K_0\mathbf{x}(0)-\mathbf{x}_0]+\kappa_1^T[K_1\mathbf{x}(1)-\mathbf{x}_f],\end{aligned}\tag{3.38}$$

the first-order optimality conditions are,

$$\begin{aligned}\dot{\mathbf{x}}-(\tau_f-\tau_0)\mathbf{f}(\mathbf{x},\mathbf{u})=0,&\qquad (\tau_{\{}-\tau_{\prime})\mathbf{f}_{\mathbf{u}}^T\lambda+\mathbf{h}_{\mathbf{u}}^T\xi=0,\\ K_0\mathbf{x}(0)-\mathbf{x}_0=0,&\qquad \mathbf{h}(\mathbf{x},\mathbf{u})+\mathbf{s}(t)\circ\mathbf{s}(t)=0,\\ K_1\mathbf{x}(1)-\mathbf{x}_f=0,&\qquad 2\xi(t)\circ\mathbf{s}(t)=0,\\ \dot{\lambda}+(\tau_f-\tau_0)\mathbf{f}_{\mathbf{x}}^T\lambda+\mathbf{h}_{\mathbf{x}}^T\xi=0,&\qquad \psi(\mathbf{x}(0),\mathbf{x}(1),\tau_0,\tau_f)=0,\\ \{\lambda(t_0),\lambda(t_f)\}=\{-\mathcal{C}_{\mathbf{x}(\tau_0)},\mathcal{C}_{\mathbf{x}(\tau_f)}\},&\qquad \{\mathcal{H}|_{t=0},\mathcal{H}|_{t=1}\}=\{\mathcal{C}_{\tau_0},-\mathcal{C}_{\tau_f}\},\end{aligned}\tag{3.39}$$

Approximations for $\widehat{\mathbf{x}}(t),\widehat{\mathbf{u}}(t)$ and $\widehat{\mathbf{s}}(t)$ are defined in Eqns. (3.7), (3.9) and (3.10) respectively. The costates $\lambda(t)$ and Lagrange multiplier functions $\xi(t)$ in $\mathcal{M}_\lambda$ are approximated as,

$$\widehat{\lambda}(t)=\sum_{k=1}^{N_{\mathbf{x}}}W_k(t)\widetilde{\gamma}_k\in\mathbb{P}_x,\quad \widehat{\xi}(t)=\sum_{k=1}^{N_{\mathbf{s}}}V_k(t)\widetilde{\zeta}_k\in\mathbb{P}_s,\tag{3.40}$$

where $\widetilde{\gamma}_k\in\mathbb{R}^n$ and $\widetilde{\zeta}_k\in\mathbb{R}^q$.

As in Eqn. (3.11), the approximate state dynamics is written as,

$$\int_0^1 W_i^T \left[(\tau_f - \tau_0)\widehat{\mathbf{f}} - \sum_{k=1}^{N_\mathbf{x}} \alpha_k \dot{\phi}_k^\mathbf{x} \right] dt + \frac{1}{2} K_0^T \pi_0 \phi_i^\mathbf{x}(0) + \frac{1}{2} K_1^T \pi_1 \phi_i^\mathbf{x}(1) = 0, \tag{3.41}$$

$$K_0 \sum_{k=1}^{N_\mathbf{x}} \alpha_k \phi_k^\mathbf{x}(0) - \mathbf{x}_0 = 0, \qquad K_1 \sum_{k=1}^{N_\mathbf{x}} \alpha_k \phi_k^\mathbf{x}(1) - \mathbf{x}_f = 0, \tag{3.42}$$

where $\pi_0 \in \mathbb{R}^a$ and $\pi_1 \in \mathbb{R}^{n-a}$. For the rest of the trajectory conditions is Eqn. (3.39), weighted residual approximations are defined by taking the test functions same as the corresponding trial functions. This is done to establish mapping between the KKT conditions and the approximated first-order optimality conditions. Using approximations defined in Eqn. (3.40) we write,

$$\int_0^1 \left[(\tau_f - \tau_0) \sum_{k=1}^{N_\mathbf{x}} \dot{W}_k \widetilde{\gamma}_k + \widehat{\mathbf{f}}_\mathbf{x}^T \sum_{k=1}^{N_\mathbf{x}} W_k \widetilde{\gamma}_k + \widehat{\mathbf{h}}_\mathbf{x}^T \sum_{k=1}^{N_\mathbf{s}} V_k \widetilde{\zeta}_k \right] \phi_i^\mathbf{x} dt = 0, \tag{3.43}$$

$$\sum_{k=1}^{N_\mathbf{x}} W_k(0)\widetilde{\gamma}_k + [\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}_{\mathbf{x}(0)}^T \upsilon + \kappa_0 K_0] = 0, \tag{3.44}$$

$$\sum_{k=1}^{N_\mathbf{x}} W_k(1)\widetilde{\gamma}_k - [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)}^T \upsilon + \kappa_1 K_1] = 0, \tag{3.45}$$

$$\int_0^1 [\widehat{\mathbf{f}}_\mathbf{u}^T \sum_{k=1}^{N_\mathbf{x}} W_k \widetilde{\gamma}_k + \widehat{\mathbf{h}}_\mathbf{u}^T \sum_{k=1}^{N_\mathbf{s}} V_k \widetilde{\zeta}_k] \phi_r^\mathbf{u} dt = 0, \tag{3.46}$$

$$\int_0^1 [\widehat{\mathbf{h}} + \sum_{k=1}^{N_\mathbf{s}} \varsigma_k \phi_k^\mathbf{s} \circ \sum_{l=1}^{N_\mathbf{s}} \varsigma_l \phi_l^\mathbf{s}] \phi_q^\mathbf{s} dt = 0, \tag{3.47}$$

$$2\int_0^1 [\sum_{k=1}^{N_\mathbf{s}} \varsigma_k \phi_k^\mathbf{s} \circ \sum_{k=1}^{N_\mathbf{s}} V_k \widetilde{\zeta}_k] \phi_q^\mathbf{s} dt = 0, \tag{3.48}$$

$$\widehat{\psi}(\alpha_k, \phi_k^\mathbf{x}(0), \phi_k^\mathbf{x}(1), \tau_0, \tau_f) = 0, \tag{3.49}$$

$$\sum_{k=1}^{N_\mathbf{x}} \widetilde{\gamma}_k^T W_k^T \widehat{\mathbf{f}}|_{t=o} + \widehat{\Psi}_{\tau_0} + \upsilon^T \widehat{\psi}_{\tau_0} = 0, \tag{3.50}$$

$$\sum_{k=1}^{N_\mathbf{x}} \widetilde{\gamma}_k^T W_k^T \widehat{\mathbf{f}}|_{t=1} + \widehat{\Psi}_{\tau_f} + \upsilon^T \widehat{\psi}_{\tau_f} = 0. \tag{3.51}$$

where $i = 1, .., N_\mathbf{x}$, $q = 1, .., N_\mathbf{s}$, $r = 1, .., N_\mathbf{u}$. Eqns. (3.41)-(3.51) represent the

indirect method solution to problem $\mathcal{M}$ and is an approximation of the true optimal solution in a weighted residual sense.

E. Primal-Dual Mapping Discrepancies

Having derived the KKT conditions and the approximated first-order optimality conditions, the next step is to look for any discrepancies between the two and, if possible, derive meaningful costate estimates form the KKT multipliers associated with $\mathcal{M}_{\phi\lambda}$. By comparing $\mathcal{M}_{\phi\lambda}$ and $\mathcal{M}_{\lambda\phi}$, we find that a one-to-one mapping would exist between $\left\{\gamma_j|_{j=1}^{N_{\mathbf{x}}}, \zeta_p|_{p=1}^{N_{\mathbf{s}}}, \mu_0, \mu_1, \eta\right\}$ and $\left\{\widetilde{\gamma}_k|_{k=1}^{N_{\mathbf{x}}}, \widetilde{\zeta}_p|_{p=1}^{N_{\mathbf{s}}}, \kappa_0, \kappa_1, \upsilon\right\}$ if the following conditions are satisfied,

$$[\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}^T_{\mathbf{x}(0)}\eta + \mu_0 K_0 + \sum_{j=1}^{N_{\mathbf{x}}} W_j(0)\gamma_j]\phi_i^{\mathbf{x}}(0) = 0; \qquad i = 1, 2, ..., N_{\mathbf{x}}, \tag{3.52}$$

$$[\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}^T_{\mathbf{x}(1)}\eta + \mu_1 K_1 - \sum_{j=1}^{N_{\mathbf{x}}} W_j(1)\gamma_j]\phi_i^{\mathbf{x}}(1) = 0; \qquad i = 1, 2, ..., N_{\mathbf{x}}, \tag{3.53}$$

and,

$$\int_0^1 \sum_{j=1}^{N_{\mathbf{x}}} \mathbf{r}^T W_{\mathbf{x}j}\phi_i^{\mathbf{x}}\gamma_j dt = 0; \qquad j = 1, .., N_{\mathbf{x}} \tag{3.54}$$

$$\int_0^1 \sum_{j=1}^{N_{\mathbf{x}}} \mathbf{r}^T W_{\mathbf{u}j}\phi_r^{\mathbf{u}}\gamma_j dt = 0; \qquad r = 1, .., N_{\mathbf{u}} \tag{3.55}$$

$$\int_0^1 \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j^T W_{\tau_0 j}\mathbf{r} dt = 0 \tag{3.56}$$

$$\int_0^1 \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j^T W_{\tau_f j}\mathbf{r} dt = 0. \tag{3.57}$$

Further, the KKT system imposes two extra conditions as,

$$K_0 \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j\phi_j(0) = 0, \qquad K_1 \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j\phi_j(1) = 0, \tag{3.58}$$

which are not applicable for the case when $\nu_0 = \nu_1 = 0$, however, as we shall see in Chapters V and VI, these conditions have an interesting implication in the case of least-square method and the generalized moment method for optimal control.

Conditions given by Eqns. (3.52)-(3.57) are referred as equivalence conditions because their satisfaction amounts to the complete equivalence of direct and indirect solution of an optimal control problem (see Figure (8)).

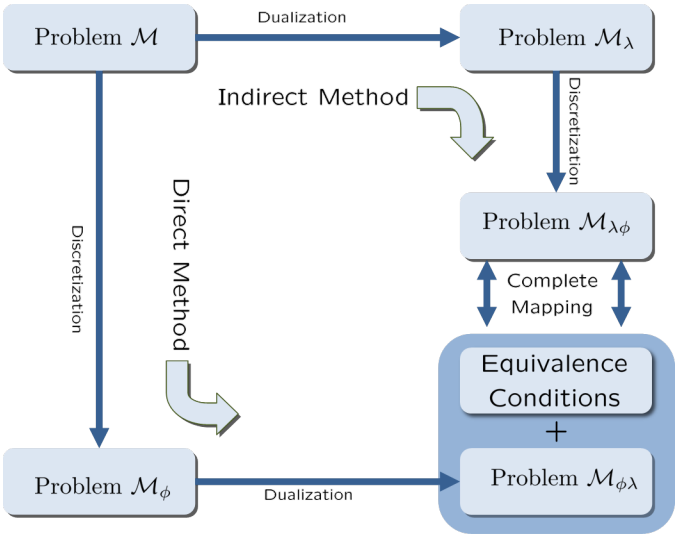


Fig. 8. For a complete mapping to exist between $\mathcal{M}_{\phi\lambda}$(Direct Method) and $\mathcal{M}_{\lambda\phi}$(Indirect Method), a set of equivalence conditions must be satisfied along with the KKT conditions associated with $\mathcal{M}_\phi$.

F. Costate Approximation

Assuming that Eqns. (3.52)-(3.57) are satisfied and Eqn. (3.58) is appropriately taken care of, the costates can be estimated from the KKT multipliers as,

$$\widehat{\lambda}(t) = \sum_{k=1}^{N_{\mathbf{x}}} W_k(t)\gamma_k, \tag{3.59}$$

and the Lagrange multiplier functions can be constructed as,

$$\widehat{\xi}(t) = \sum_{k=1}^{N_{\mathbf{s}}} V_k(t)\zeta_k. \tag{3.60}$$

Also,

$$\kappa_0 = \mu_0, \quad \kappa_1 = \mu_1, \quad \upsilon = \eta. \tag{3.61}$$

We make an interesting observation about the structure of the approximated costate dynamics:

"If $\widehat{\mathbf{x}}(t) \in \mathbb{V}_{\mathbf{x}}$, and the residual in state dynamics is set orthogonal to the space of test functions $\mathbb{P}_x$, then under the equivalence conditions, the costate estimates $\widehat{\lambda}(t) \in \mathbb{P}_x$ and the residual in costate dynamics is orthogonal to the space $\mathbb{V}_{\mathbf{x}}$."

This mapping relationship is depicted in Figure (9).

G. Conclusions

This chapter presented the formulation and optimality analysis of the weighted residual method for optimal control. The formulation was done in a generic manner without making any particular choice for relevant approximation spaces or numerical quadrature scheme. Comparison of KKT system with the approximated first-order optimality conditions resulted in a set of equivalence conditions necessary for a mapping to exist between direct and indirect solutions of the optimal control problem. When these equivalence conditions are satisfied, costates can be estimated form the KKT multipliers associated with the state dynamics constraints in the nonlinear programming problem.

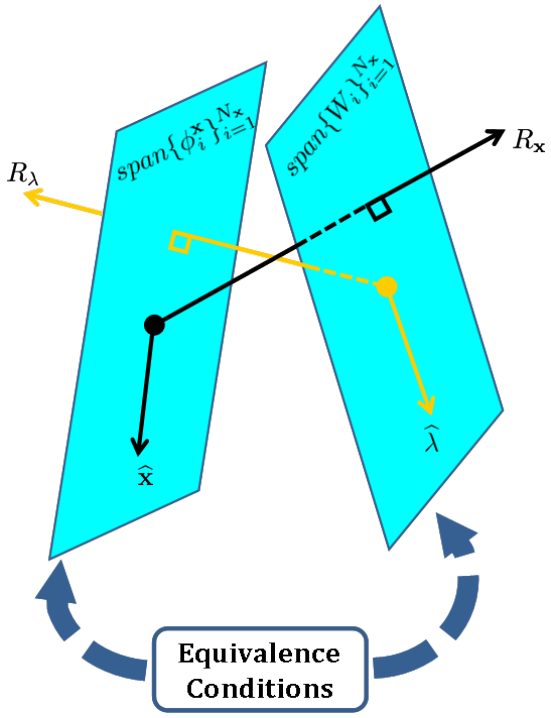


Fig. 9. Under the equivalence conditions, costates can be estimated in the projection space $\mathbb{P}_x$ and the residual in costate dynamics is orthogonal to the approximating space $\mathbb{V}_{\mathbf{x}}$.

CHAPTER IV

THE METHOD OF HILBERT SPACE PROJECTION

In this chapter, we consider the first special case of WRM where the test functions are chosen to be same as the trial functions approximating the trajectory variables. The method of Hilbert space projection (MHSP) closely resembles tau-methods discussed in Chapter II. In the optimal control literature tau-methods have been used with global orthogonal approximating functions, such as Legendre[53], Chebyshev[54, 55], or Fourier[56, 57] series expansions. The MHSP differs from the these methods as the basis functions need only be linearly independent but not necessarily orthogonal. This allows the use of more general class of basis functions such as B-Splines or partition of unity based approximations for direct transcription of optimal control problems. Depending upon the problem in hand, it can be advantageous to select such basis functions to construct the approximation space. Further, the MHSP allows the approximation of state and control variables in two different function spaces.

The formulation and analysis of MHSP is carried out in the framework of WRM as presented in Chapter III. The first step is the formulation of a nonlinear programming problem for the original optimal control problem $\mathcal{M}$. Next, equivalence conditions are derived and costate estimation results are presented for the MHSP. Further, the implementation of MHSP is described using B-Splines as approximating functions and a numerical quadrature scheme is outlined. Finally, a number of numerical examples are solved using this method and numerical convergence is demonstrated through simulations.

A. Direct Transcription Formulation

For WRM, take the projection space to be same as the approximating space for both state dynamics and path constraints, so that,

$$W_k(t) = \phi_k^{\mathbf{x}}(t); \qquad k = 1, .., N_{\mathbf{x}}, \tag{4.1}$$

$$V_k(t) = \phi_k^{\mathbf{s}}(t); \quad k = 1, .., N_{\mathbf{s}}. \tag{4.2}$$

Also take,

$$\nu_0 = \nu_1 = 0. \tag{4.3}$$

Further assume that $\phi_j^{\mathbf{x}}$'s are continuous functions on $[0, 1]$ so that the integration by parts relationship in Eqn. (3.24) is valid.

B. Nonlinear Programming Problem

Using Eqns. (4.1) and (4.3), the nonlinear programming problem formulated in Chapter III Section B takes the following form: Determine $\{\alpha_k \in \mathbb{R}^n\}_{k=1}^{N_{\mathbf{x}}}, \{\beta_k \in \mathbb{R}^m\}_{k=1}^{N_{\mathbf{u}}}, \{\varsigma_k \in \mathbb{R}^q\}_{k=1}^{N_{\mathbf{s}}}$, and time instances τ_0 and τ_f, that minimize the cost,

$$\widehat{J} = \widehat{\Psi}(\alpha_k, \phi_k^{\mathbf{x}}(0), \phi_k^{\mathbf{x}}(1), \tau_0, \tau_f), \tag{4.4}$$

subject to the constraints,

$$\int_0^1 \left[(\tau_f - \tau_0)\widehat{\mathbf{f}}(\alpha_k, \beta_k, \phi_k^{\mathbf{x}}, \phi_k^{\mathbf{u}}) - \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \dot{\phi}_k^{\mathbf{x}} \right] \phi_j^{\mathbf{x}}(t) dt = 0, \tag{4.5}$$

$$K_0 \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \phi_k^{\mathbf{x}}(0) - \mathbf{x}_0 = 0, \quad K_1 \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \phi_k^{\mathbf{x}}(1) - \mathbf{x}_f = 0, \tag{4.6}$$

$$\widehat{\psi}(\alpha_k, \phi_k^{\mathbf{x}}(0), \phi_k^{\mathbf{x}}(1), \tau_0, \tau_f) = 0, \tag{4.7}$$

$$\int_0^1 \left[\widehat{\mathbf{h}}(\alpha_k, \beta_k, \phi_k^{\mathbf{x}}, \phi_k^{\mathbf{u}}) + \sum_{k=1}^{N_{\mathbf{s}}} \varsigma_k \phi_k^{\mathbf{s}} \circ \sum_{l=1}^{N_{\mathbf{s}}} \varsigma_l \phi_l^{\mathbf{s}} \right] \phi_p^{\mathbf{s}}(t) dt = 0, \tag{4.8}$$

where $j = 1, .., N_{\mathbf{x}}$ and $p = 1, .., N_{\mathbf{s}}$. This approximation scheme as defined by Eqns. (4.4)-(4.8) represents the method of Hilbert space projection.

Next, we derive the equivalence conditions and costate estimation results for this method.

C. Equivalence Conditions

For MHSP, the conditions defined by Eqns. (3.52)-(3.57) in Chapter III Section E are trivially satisfied because,

$$\frac{\partial \phi_j^{\mathbf{x}}}{\partial \mathbf{u}} = \frac{\partial \phi_j^{\mathbf{x}}}{\partial \tau_0} = \frac{\partial \phi_j^{\mathbf{x}}}{\partial \tau_f} = 0. \tag{4.9}$$

Further, since not all basis functions $\phi_k^{\mathbf{x}}|_{k=1}^{N_{\mathbf{x}}}$ are zero at the boundaries, the conditions in Eqns. (3.52) and (3.53) are not explicitly satisfied. Therefore, the equivalence conditions for MHSP take the form,

$$\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}_{\mathbf{x}(0)}^T \eta + \mu_0 K_0 + \sum_{j=1}^{N_{\mathbf{x}}} \phi_j^{\mathbf{x}}(0) \gamma_j = 0, \tag{4.10}$$

$$\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)}^T \eta + \mu_1 K_1 - \sum_{j=1}^{N_{\mathbf{x}}} \phi_j^{\mathbf{x}}(1) \gamma_j = 0. \tag{4.11}$$

The extra conditions in Eqn. (3.58) are not applicable for MHSP as $\nu_0 = \nu_1 = 0$ in this method.

Eqns. (4.10) and (4.11) when added to the KKT conditions, fill the "gap" between the direct and indirect method solutions of problem $\mathcal{M}$. We see that the direct method discretization of problem $\mathcal{M}$ does not explicitly impose the boundary condi-

tions on the discrete costates. This loss of information is restored by the equivalence conditions in Eqns. (4.10) and (4.11).

D. Costate Estimates

The equivalence conditions defined in the previous section establish the relationship between KKT multipliers $\{\gamma_k\}_{k=1}^{N_{\mathbf{x}}}$ associated with the constraints in Eqn. (4.5), and the costate approximation $\widehat{\lambda}(t)$. Similarly, the Lagrange multiplier functions and variables $\{\xi(t), \mu_0, \mu_1, \eta\}$ in $\mathcal{M}_\lambda$ can be obtained from the corresponding KKT multipliers. The costate estimation results for MHSP can be summarized via the following theorem:

Theorem D.1 (Costate Mapping Theorem for the MHSP) *Assume that an optimal control problem is solved using the method of Hilbert space projection with the state approximation $\widehat{\mathbf{x}}(t) \in \mathbb{V}_{\mathbf{x}}$, and the equivalence conditions hold. Then, the estimates of the costates $\widehat{\lambda}(t) \in \mathbb{V}_{\mathbf{x}}$, Lagrange multiplier functions $\xi(t)$ associated with the path constraints and the terminal covector (υ) can be obtained using the KKT multipliers $(\gamma_k, \zeta_k, \eta)$ of the associated NLP as,*

$$\widehat{\lambda}(t) = \sum_{k=1}^{N_{\mathbf{x}}} \gamma_k \phi_k^{\mathbf{x}}(t) \in \mathbb{V}_{\mathbf{x}}, \quad \widehat{\xi}(t) = \sum_{k=1}^{N_{\mathbf{s}}} \widehat{\zeta}_k \phi_k^{\mathbf{s}}(t) \in \mathbb{V}_{\mathbf{s}}, \quad \eta = \upsilon. \tag{4.12}$$

Proof The solution to Problem $\mathcal{M}_{\lambda\phi}$ exists by assumption. Since equivalence conditions hold, the results in Chapter III Section F are valid. ∎

E. MHSP Using B-Spline Approximation

There are a number of ways to select the basis functions for state and control approximations with MHSP. The basis functions for state approximations should be

linearly independent and continuous on the computational domain of the problem in hand. This section describes the implementation of MHSP using B-Splines as approximating functions. implementation of the MHSP, which requires that the integrals in Eqns. (4.5) and (4.8) be evaluated numerically. This can be accomplished by using a numerical quadrature scheme. Using numerical quadrature changes the structure of the resulting NLP and in turn the KKT conditions. However, the results on costate estimation and equivalence conditions can still be derived if the quadrature scheme is chosen so that the integration by parts formula in Eqn. (3.24) holds. Here we take a simple example to substantiate this claim in the case of B-Spline approximation.

A B-Spline is a piecewise polynomial function with a specified level of global smoothness. Also, the B-Spline basis functions have local support, which means that each basis function only influences a local region of the global trajectory. Local support is a desirable property of basis functions for numerically stable algorithms. A brief introduction to the construction and properties of B-Splines is given in Chapter II Section C.4. For brevity, we consider a simplified form of problem $\mathcal{M}$ for mathematical brevity. Problem $\mathcal{A}$ is to determine the state-control pair $\{\mathbf{x}(t) \in \mathbb{R}^n, \mathbf{u}(t) \in \mathbb{R}^m; t \in [0,1]\}$, that minimizes,

$$J = \Psi(\mathbf{x}(1)), \tag{4.13}$$

subject to,

$$\dot{\mathbf{x}}(t) = F(\mathbf{x}(t), \mathbf{u}(t)), \tag{4.14}$$

$$\psi(\mathbf{x}(0), \mathbf{x}(1)) = 0, \tag{4.15}$$

where, $\Psi : \mathbb{R}^n \to \mathbb{R}, F : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n, \psi : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^p$. To solve this problem using the MHSP, we approximate the state and control trajectories as B-Splines. So

that,

$$\mathbf{x}(t) \approx \widehat{\mathbf{x}}(t) = \sum_{k=1}^{N} \alpha_k B_{k,r}(t), \qquad \mathbf{u}(t) \approx \widehat{\mathbf{u}}(t) = \sum_{k=1}^{N} \beta_k B_{k,r}(t), \tag{4.16}$$

for some N, r, s and breakpoints $0 = t_0 < t_1 < .. < t_N = 1$. Let $\Omega_i := [t_{i-1}, t_i]$. For each domain Ω_i let $\{b_{ij}\}_{j=0}^{N_q}$ be the number N_q of Legendre-Gauss (LG) quadrature points with corresponding quadrature weights w_{ij} (see Figure (10)). Here we assume that the weights w_{ij} have been appropriately scaled to transform the integration domain $[-1, 1]$ of the LGL points to Ω_i. Then, the integral of a function $f(t)$ over interval Ω_i can be approximated as,

$$\int_{\Omega_i} f(t)dt \approx \sum_{j=0}^{N_q} w_{ij} f(b_{ij}). \tag{4.17}$$

Also, given N_q, the integral evaluation using LG rule is exact for polynomials of

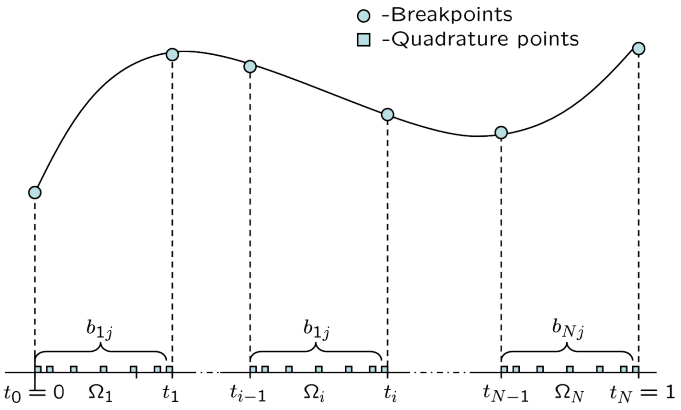


Fig. 10. The arrangement of breakpoints and quadrature points in domain $t \in [0, 1]$. $\{t_i\}_{i=0}^{N}$ are the breakpoints. $\{b_{ij}\}_{j=0}^{N_q}$ are the quadrature points for domain $\Omega_i = [t_{i-1}, t_i]$.

degree $2N_q - 1$ or less,

$$p(t) \in P^{2N_q-1}; \quad \int_{\Omega_i} p(t)dt = \sum_{j=0}^{N_q} w_{ij} p(b_{ij}), \tag{4.18}$$

where P^n represents the space of all polynomials of degree less than or equal to n. The B-spline basis functions are piecewise polynomials such that $\{B_{k,r}(t) \in P^{r-1}; t \in \Omega_i\}_{k=1}^{N}$ over domains $\{\Omega_i\}_{i=1}^{N}$. We choose $N_q \geq r-1$, for which the following integration by parts formula holds,

$$\sum_{i=1}^{N}\sum_{j=0}^{N_q} w_{ij}\left[\dot{B}_{k,r}(b_{ij})B_{l,r}(b_{ij}) + \dot{B}_{l,r}(b_{ij})B_{k,r}(b_{ij})\right] = [B_{k,r}(1)B_{l,r}(1) - B_{k,r}(0)B_{l,r}(0)]. \tag{4.19}$$

A nonlinear programming problem $\mathcal{A}_N$ is formulated based on Section B and by using Eqn. (4.17) to approximate the integrals. The problem $\mathcal{A}_N$ is to determine $\{\alpha_k \in \mathbb{R}^n, \beta_k \in \mathbb{R}^m\}_{k=1}^{N}$ that minimize,

$$J = \widehat{\Psi}(\alpha_k, B_{k,r}(1)), \tag{4.20}$$

subject to the constraints,

$$\sum_{i=1}^{N}\{\sum_{j=0}^{N_q} w_{ij}[\widehat{F}(\alpha_k, \beta_k, B_{k,r}(b_{ij})) - \sum_{k=1}^{N} \alpha_k \dot{B}_{k,r}(b_{ij})]B_{l,r}(b_{ij})\} = 0; \;\; l = 1, .., N, \tag{4.21}$$

$$\widehat{\psi}(\alpha_k, B_{k,r}(0), B_{k,r}(1)) = 0. \tag{4.22}$$

Next, we derive the KKT conditions for problem $\mathcal{A}_N$. The augmented cost is defined as,

$$\begin{aligned} J' = {} & \widehat{\Psi}(\alpha_k, B_{k,r}(1)) + \nu^T \widehat{\psi}(\alpha_k, B_{k,r}(0), B_{k,r}(1)) \\ & + \sum_{l=1}^{N} \gamma_l \left[\sum_{i=1}^{N}\{\sum_{j=0}^{N_q} w_{ij}[\widehat{F}(\alpha_k, \beta_k, B_{k,r}(b_{ij})) - \sum_{k=1}^{N} \alpha_k \dot{B}_{k,r}(b_{ij})]B_{l,r}(b_{ij})\}\right], \end{aligned}$$

where $\gamma_l \in \mathbb{R}^n$ and $\nu \in \mathbb{R}^p$ are the KKT multipliers associated with the constraints given by Eqns. (4.21) and (4.22) respectively. The KKT conditions denoted by $\mathcal{A}_{N\lambda}$ are derived by setting the partial derivatives of J' with respect to free variables equal to zero. So that for $m = 1, .., N$,

$$0 = \frac{\partial J'}{\partial \alpha_m} = \sum_{l=1}^{N} \gamma_l \left[\sum_{i=1}^{N} \{ \sum_{j=0}^{N_q} w_{ij} [\widehat{F}_{\mathbf{x}} B_{m,r}(b_{ij}) B_{l,r}(b_{ij}) - \dot{B}_{m,r}(b_{ij}) B_{l,r}(b_{ij})] \} \right] \tag{4.23}$$

$$+ [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)} \nu] B_{m,r}(1) + \widehat{\psi}_{\mathbf{x}(0)} \nu B_{m,r}(0). \tag{4.24}$$

Using Eqn. (4.19), we get,

$$\begin{aligned} 0 = \frac{\partial J'}{\partial \alpha_m} = & \sum_{l=1}^{N} \gamma_l \left[\sum_{i=1}^{N} \{ \sum_{j=0}^{N_q} w_{ij} [\widehat{F}_{\mathbf{x}} B_{m,r}(b_{ij}) B_{l,r}(b_{ij}) + \dot{B}_{l,r}(b_{ij}) B_{m,r}(b_{ij})] \} \right] \\ & - \sum_{l=1}^{N} \gamma_l \left[B_{m,r}(1) B_{l,r}(1) - B_{m,r}(0) B_{l,r}(0) \right] \\ & + [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)} \nu] B_{m,r}(1) + \widehat{\psi}_{\mathbf{x}(0)} \nu B_{m,r}(0). \end{aligned} \tag{4.25}$$

Re-arranging Eqn. (4.25),

$$\begin{aligned} 0 = \frac{\partial J'}{\partial \alpha_m} = & \sum_{i=1}^{N} \{ \sum_{j=0}^{N_q} w_{ij} [\widehat{F}_{\mathbf{x}} \sum_{l=1}^{N} \gamma_l B_{l,r}(b_{ij}) + \sum_{l=1}^{N} \gamma_l \dot{B}_{l,r}(b_{ij})] B_{m,r}(b_{ij}) \} \\ & + [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)} \nu - \sum_{l=1}^{N} \gamma_l B_{l,r}(1)] B_{m,r}(1) \\ & + [\widehat{\psi}_{\mathbf{x}(0)} \nu + \sum_{l=1}^{N} \gamma_l B_{l,r}(0)] B_{m,r}(0). \end{aligned} \tag{4.26}$$

Also,

$$0 = \frac{\partial J'}{\partial \beta_m} = \sum_{i=1}^{N}\{\sum_{j=0}^{N_q} w_{ij}\widehat{F}_{\mathbf{u}}[\sum_{l=1}^{N}\gamma_l B_{l,r}(b_{ij})]B_{m,r}(b_{ij})\}, \tag{4.27}$$

$$0 = \frac{\partial J'}{\partial \gamma_m} = \sum_{i=1}^{N}\{\sum_{j=0}^{N_q} w_{ij}[\widehat{F}(\alpha_k, \beta_k, B_{k,r}(b_{ij})) - \sum_{k=1}^{N}\alpha_k \dot{B}_{k,r}(b_{ij})]B_{m,r}(b_{ij})\}, \tag{4.28}$$

$$0 = \frac{\partial J'}{\partial \nu} = \widehat{\psi}(\alpha_k, B_{k,r}(0), B_{k,r}(1)). \tag{4.29}$$

Thus, Eqns. (4.26) to (4.29) constitute the KKT conditions $\mathcal{A}_{N\lambda}$. Based on Chapter III Section D and using Eqn. (4.17), the discretized first-order optimality conditions $\mathcal{A}_{\lambda N}$ are,

$$\begin{aligned}
\sum_{i=1}^{N}\{\sum_{j=0}^{N_q} w_{ij}[\widehat{F}(\alpha_k, \beta_k, B_{k,r}(b_{ij})) - \sum_{k=1}^{N}\alpha_k \dot{B}_{k,r}(b_{ij})]B_{m,r}(b_{ij})\} &= 0,\\
\sum_{i=1}^{N}\{\sum_{j=0}^{N_q} w_{ij}[\widehat{F}_{\mathbf{x}}\sum_{l=1}^{N}\widetilde{\gamma}_l B_{l,r}(b_{ij}) + \sum_{l=1}^{N}\widetilde{\gamma}_l \dot{B}_{l,r}(b_{ij})]B_{m,r}(b_{ij})\} &= 0,\\
\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)}\upsilon - \sum_{l=1}^{N}\widetilde{\gamma}_l B_{l,r}(1) &= 0,\\
\widehat{\psi}_{\mathbf{x}(0)}\upsilon + \sum_{l=1}^{N}\widetilde{\gamma}_l B_{l,r}(0) &= 0,\\
\widehat{\psi}(\alpha_k, B_{k,r}(0), B_{k,r}(1)) &= 0,
\end{aligned} \tag{4.30}$$

for $m = 1, .., N$. Comparing $\mathcal{A}_{N\lambda}$ and $\mathcal{A}_{\lambda N}$, we see that the results form Section C and D hold. The equivalence conditions are,

$$\sum_{k=1}^{N}\gamma_k B_{k,r}(0) = -\widehat{\psi}^T_{\mathbf{x}(0)}\nu, \tag{4.31}$$

$$\sum_{k=1}^{N}\gamma_k B_{k,r}(1) = [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}^T_{\mathbf{x}(1)}\nu]. \tag{4.32}$$

The costates can be estimates as,

$$\widehat{\lambda}(t) = \sum_{k=1}^{N} \gamma_k B_{k,r}(t). \tag{4.33}$$

Thus, the analysis carried out in this section verifies the claim that the results on costate estimation and equivalence conditions can be derived if the quadrature scheme is chosen so that the integration by parts formula in Eqn. (3.24) holds.

F. Numerical Convergence Analysis

In this section, convergence properties of MHSP are demonstrated numerically by solving four example problems which have known analytical solutions. The MHSP is implemented using three types of approximation schemes and the effect of selecting local/global basis functions is analyzed. The first two examples have smooth solutions, while the third example has a discontinuity in one of the costates and corners in the control solution. The fourth example has a discontinuous control solution and one of the states has a corner. We expect that for the first two problems, global approximating functions would have higher accuracy while for the problems with discontinuities and corners, local approximating functions would perform better. All the examples are programmed in MATLAB® with SNOPT as the NLP solver with 10^{-9} as feasibility and optimality tolerances. The results are compared with the available analytic solutions. The measure for accuracy is defined as the ∞-norm of the error with respect to the analytic solution.

To implement MHSP, we define the following three types of approximation schemes:

1. B-Spline approximation with fixed order ($r = 4$) and smoothness ($s = 3$). The order of approximation is increased by increasing the number of intervals N_i. This can be seen as an h-refinement scheme and is denoted as SPL-34 for the

subsequent treatment. For a given N_i, the number of unknown variables for each trajectory is $N = N_i + 3$.

2. B-Spline approximation with fixed number of intervals ($N_i = 5$) and smoothness ($s = 3$). The order of approximation is increased by increasing the spline order (r). This can be seen as a p-refinement scheme and is denoted as SPL-3r. For a given r, the number of unknown variables for each trajectory is $N = N_i(r-3)+3$.

3. A global approximation scheme using Legendre polynomials as basis functions. The order of approximation is increased by increasing the number N of Legendre basis functions. This approximation scheme is denoted as LEG.

Example problems defined in this section are used throughout this dissertation for validating numerical convergence of various algorithms. For MHSP, each example is solved by using the above three types of approximations, and N is increased from 8 to 48 with in the increments of 5.

1. Example 1: Nonlinear Plant with Terminal Cost

$$\text{Minimize: } J = -x(t_f) \tag{4.34}$$

$$\text{Subject to: } \dot{x}(t) = x(t)u(t) - x(t) - u^2(t) \tag{4.35}$$

$$x(0) = 1; \quad t_f = 5 \tag{4.36}$$

where $\{x(t), u(t)\}$ is the state-control pair and t_f is the final time. The analytic solution given by Huntington[58] is,

$$\begin{aligned} x^*(t) &= \frac{4}{1+3e^t} \\ \lambda^*(t) &= \frac{-e^{(2ln(1+3e^t)-t)}}{(e^{-5}+6+9e^5)} \\ u^*(t) &= 0.5x^*(t) \end{aligned} \tag{4.37}$$

where $\lambda^*(t)$ is the costate associated with the optimal solution. The analytic optimal cost is $J = -0.009$.

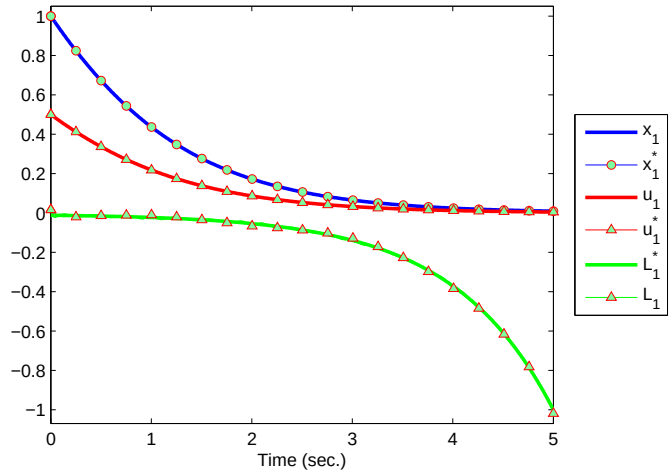


Fig. 11. MHSP solution of Example 1 using SPL3r approximation with $N = 38$.

This example is solved using MHSP with SPL-34, SPL-3r and LEG approximations. Figure (11) shows the analytical and numerical solutions for $N = 38$ with SPL-3r approximation. It is seen that the costates are not very well approximated. Convergence results are depicted in Figure (12). We see that for this example, overall performance of SPL-3r approximation is the best. All three approximations do a poor job in estimating the costates. However, SPL-3r costate estimation is relatively better for high N.

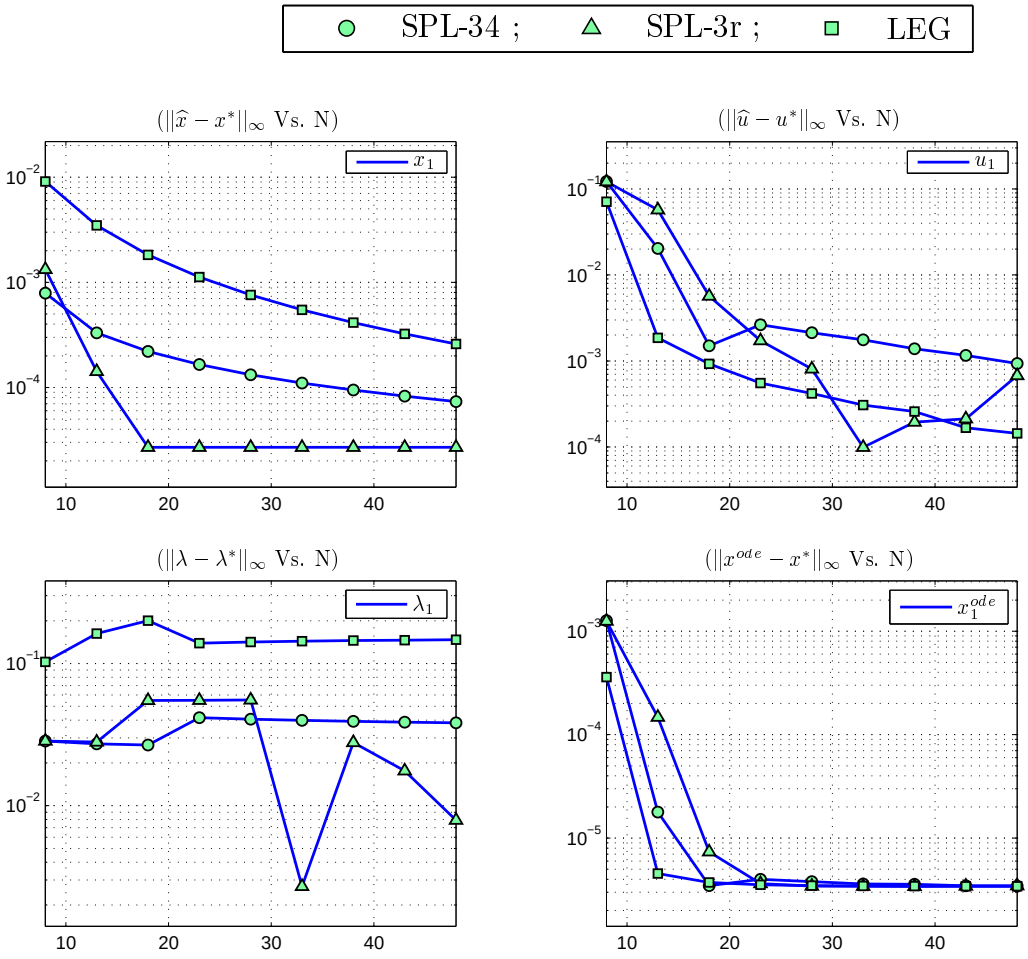


Fig. 12. Convergence of MHSP solution for Example 1.

2. Example 2: Two-state Nonlinear Plant with Terminal Constraint

$$\text{Minimize: } J = x_2(t_f) \tag{4.38}$$

$$\text{Subject to: } \dot{x_1} = 0.5x_1 + u \tag{4.39}$$

$$\dot{x_2} = x_1^2 + 0.5u^2 \tag{4.40}$$

$$x_1(0) = 1, \quad x_1(t_f) = 0.5, \quad x_2(0) = 0, \tag{4.41}$$

where $\{[x_1(t), x_2(t)], u(t)\}$ is the state-control pair and $t_f = 5$ is the final time. The

analytic solution given by Huntington [58] is,

$$x_1^*(t) = a_1 e^{\frac{3}{2}t} + a_2 e^{-\frac{3}{2}t}; \quad x_2^*(t) = a_3(e^{\frac{3}{2}t})^2 + a_4(e^{-\frac{3}{2}t})^2 + c_1, \tag{4.42}$$

$$\lambda_1^*(t) = a_5 e^{\frac{3}{2}t} + a_6 e^{-\frac{3}{2}t}; \quad \lambda_2^*(t) = 1 \tag{4.43}$$

$$u^*(t) = -\lambda_1^*(t), \tag{4.44}$$

where,

$$a_1 = \frac{\frac{1}{2} - e^{-\frac{15}{2}}}{e^{\frac{15}{2}} - e^{-\frac{15}{2}}}, \quad a_2 = \frac{e^{\frac{15}{2}} - \frac{1}{2}}{e^{\frac{15}{2}} - e^{-\frac{15}{2}}}, \tag{4.45}$$

$$a_3 = \frac{1}{2}a_1^2, \quad a_4 = a_2^2 \quad c_1 = a_2^2 - \frac{1}{2}a_1^2, \quad a_5 = -a_1, \quad a_6 = -2a_2. \tag{4.46}$$

MHSP solution obtained by using SPL-3r approximation and with $N = 38$ is com-

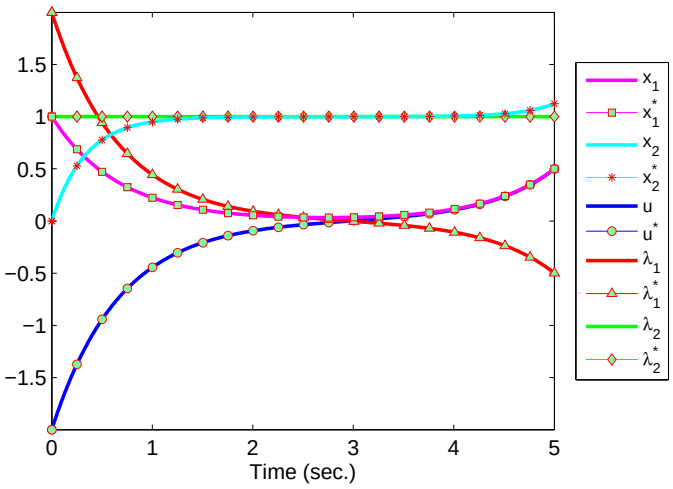


Fig. 13. MHSP solution of Example 2 using SPL3r approximation with $N = 38$.

pared with the true solution in Figure (13). Convergence results for this example are shown in Figure (14). It is clearly seen that SPL-3r approximation gives best performance for all the trajectory variables. Slight upward increase of error curves for SPL-3r is due to fixed number of quadrature nodes. As the spline order is in-

creased, the number of quadrature points should also be increased. However, in the present implementation, the number of quadrature nodes are fixed for a given interval. Since the number of intervals in SPL-3r approximation remain the same, there is some drop in accuracy as the spline order is increased. LEG approximation performs poorly in all the cases, which is due to bad approximation of the trajectory variables at boundaries.

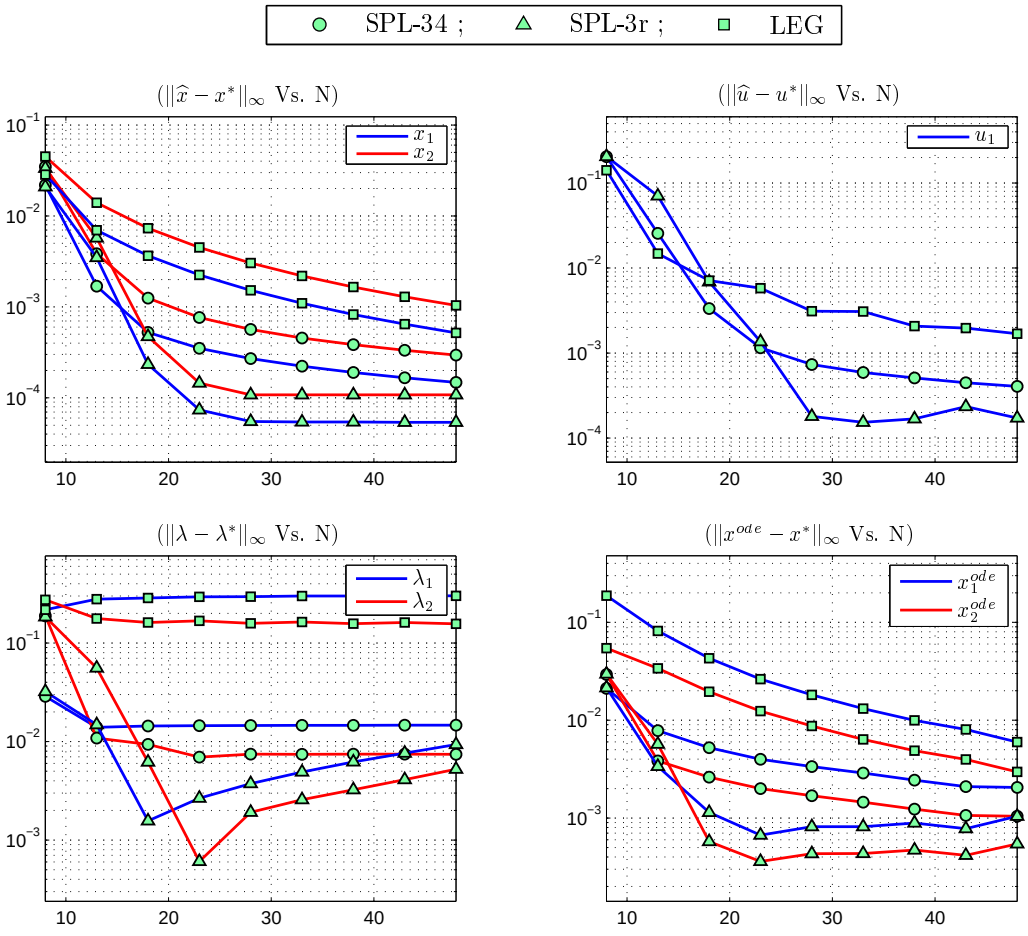


Fig. 14. Convergence of MHSP solution for Example 2.

3. Example 3: Linear Plant with State Inequality Constraint

$$\text{Minimize: } J = \frac{1}{2}\int_0^1 u^2 dt$$

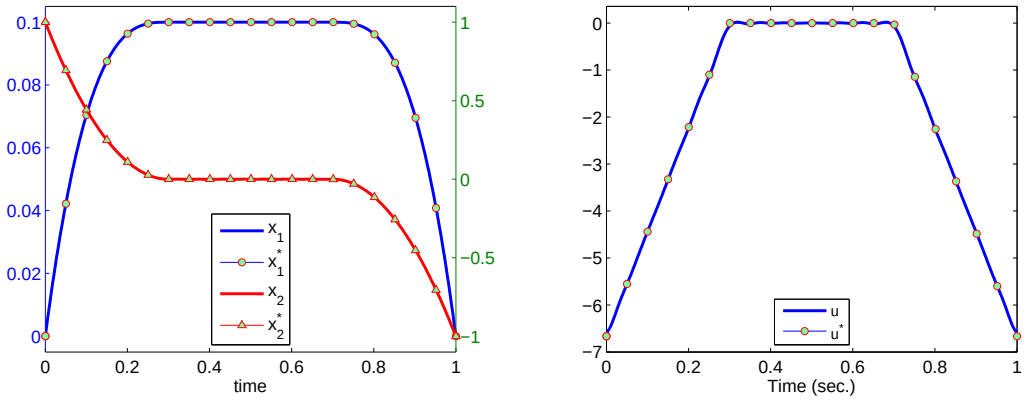
$$\text{Subject to: } \dot{x_1} = x_2 \tag{4.47}$$

$$\dot{x}_2 = u \tag{4.48}$$

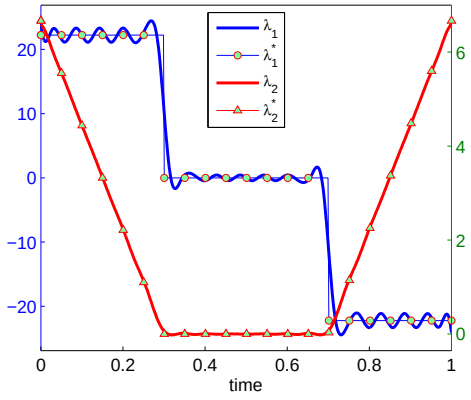
$$x_1(0) = 0, \quad x_1(1) = 0, \quad x_2(0) = 1, \quad x_2(1) = -1 \tag{4.49}$$

$$x_1(t) \leq l = 0.1 \tag{4.50}$$

In the literature[10], this problem is known as the Breakwell problem. We use this problem to demonstrate the applicability of the MHSP in the presence of pure state inequality constraints, which were not considered in our problem formulation for simplicity. The Breakwell problem has a second-order state inequality constraint. The analytic solution to this problem is given in Ref.[10]. The optimal cost is $J = \frac{4}{9l} = 4.4444$. The optimal switching structure for this problem is *free-constrained-free*, and the costate $\lambda_1(t)$ has jump discontinuities at times $t_1 = 3l = 0.3$ and $t_2 = 1 - 3l = 0.7$. Figure (15) shows the comparison of numerical solution of the Breakwell problem with the true solution. The numerical solution is obtained by using SPL-34 approximation and $N = 38$. We see that the jump discontinuity in costate $\lambda_1(t)$ is captured by the numerical solution. However, the approximation of $\lambda_1(t)$ exhibits Gibb's like phenomenon.



(a) States (b) Controls



(c) Costates

Fig. 15. Comparison of MHSP results with the analytical solution for Example 3. SPL-34 approximation with $N = 38$.

Convergence results for this problem are shown in Figure (16). It is clearly seen that the global approximation scheme SPL-3r performs poorly in this case compared to the SPL-34 approximations. This is due to the presence of discontinuities and corners in the solution.

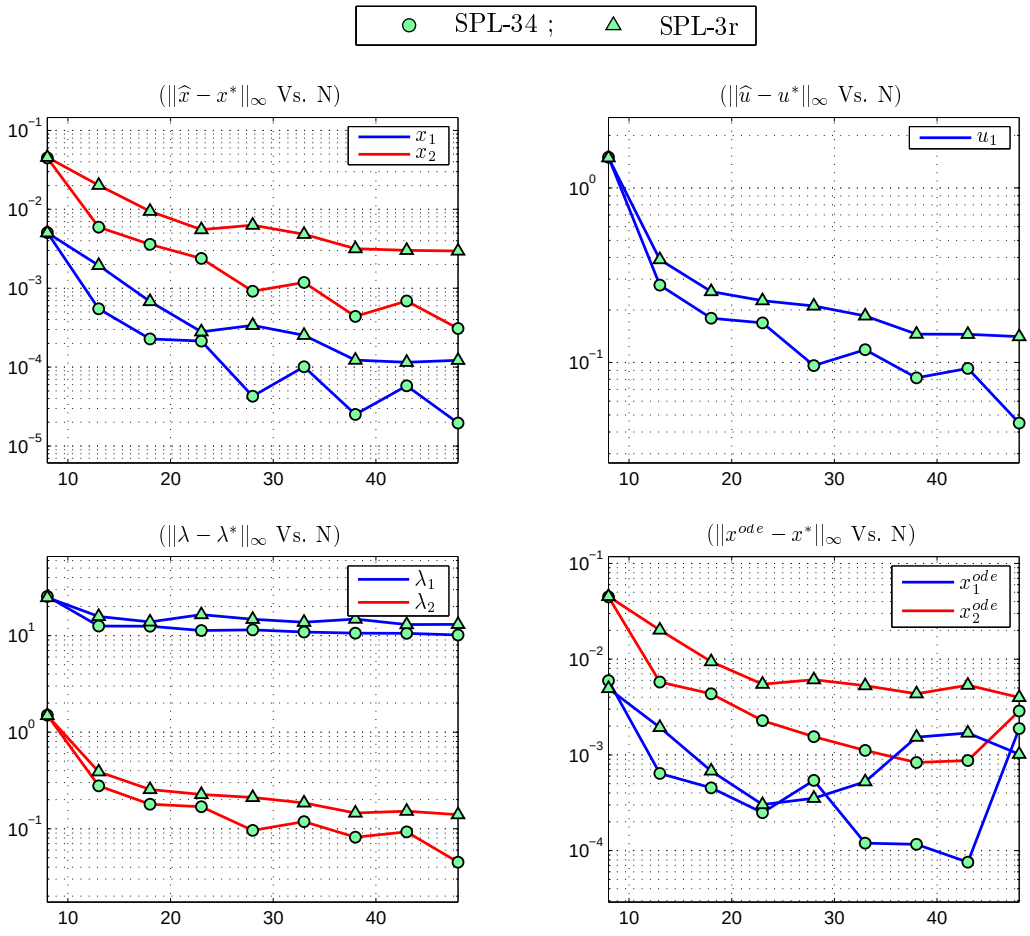


Fig. 16. Convergence of MHSP solution for Example 3.

4. Example 4: Minimum Time Problem with Bounded Control

$$\text{Minimize: } J = t_f \tag{4.51}$$

$$\text{Subject to: } \dot{x_1}(t) = x_2(t) \tag{4.52}$$

$$\dot{x_2}(t) = u(t) \tag{4.53}$$

$$x_1(0) = 3, \quad x_1(t_f) = 0, \quad x_2(0) = 2 \quad x_2(t_f) = 0 \tag{4.54}$$

$$|u(t)| \leq 1. \tag{4.55}$$

The analytic solution given by Benson [59] is,

$$t_0 = 2 + \sqrt{5}, \qquad t_f = 2(1 + \sqrt{5}) \tag{4.56}$$

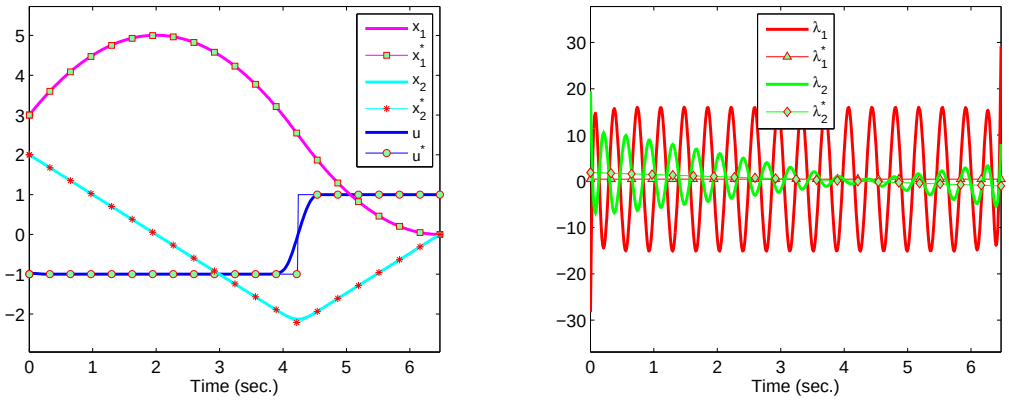
$$x_1^*(t) = \begin{cases} -t^2 \frac{1}{2} + 2t + 3 & t \in [0, t_0] \\ t^2 \frac{1}{2} - t_f t + \frac{t_f^2}{2} & t \in [t_0, t_f] \end{cases}, \tag{4.57}$$

$$x_2^*(t) = \begin{cases} -t + 2 & t \in [0, t_0], \\ t - t_f & t \in [t_0, t_f] \end{cases}, \tag{4.58}$$

$$u_1^*(t) = \begin{cases} -1 & t \in [0, t_0], \\ 1 & t \in [t_0, t_f] \end{cases}, \tag{4.59}$$

$$\lambda_1^*(t) = -\frac{1}{t_0 - t_f}, \qquad \lambda_2^*(t) = \frac{1}{t_0 - t_f} t - \frac{t_0}{t_0 - t_f}. \tag{4.60}$$

For this problem, numerical solution obtained by using SPL-3r approximation and with $N = 38$ is compared with the true solution in Figure (17). It is seen that the costates do not converge for this problem, while the state and control trajectories are approximated well by SPL-34 scheme. The reason for the non-convergence of costates is that the equivalence conditions are not satisfied in this case.



(a) States (b) Controls

Fig. 17. Comparison of MHSP results with the analytical solution for Example 4. SPL-34 approximation with $N = 38$.

Convergence results for this problem are depicted in Figure (18). We see that SPL-3r approximation performs poorly compared to the SPL-34 approximation. This is again due to the discontinuity in the control solution.

G. Conclusions

This chapter introduced the formulation of the method of Hilbert space projection. This method is flexible with respect to the choice of approximating functions for trajectory variables, where both local and global functions can be employed. The costates can be estimated from the KKT multipliers if a set of equivalence conditions are satisfied. Convergence of MHSP is demonstrated numerically as the order of approximation increases. It is observed that the choice of approximating functions effects the MHSP solution. For problems with smooth solution, global approximating functions or higher order B-Splines give accurate results. However, for problems having discontinuities and corners in their solutions, local approximating low order B-Splines work best. There are some cases where equivalence conditions are not

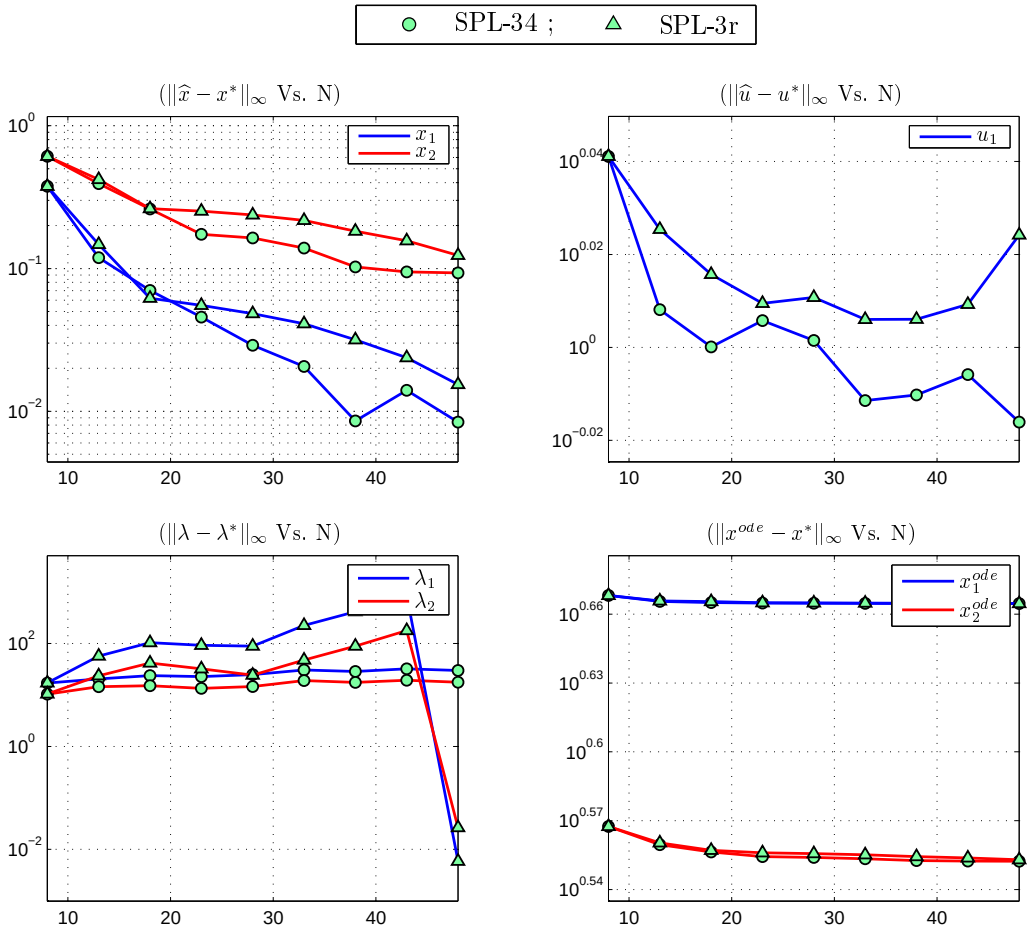


Fig. 18. Convergence of MHSP solution for Example 4.

satisfied by the nonlinear program in which case costates do not converge to the true solution. We shall see in the next chapter that this limitation does not exist with the least square method for optimal control.

CHAPTER V

THE LEAST SQUARE METHOD FOR OPTIMAL CONTROL

The method of Hilbert space projection presented in the previous chapter sets the trajectory residuals to zero in an average sense. For linear systems, it can be shown that MHSP minimizes the L^2-norm of trajectory residuals. However, this is not true for the nonlinear systems. In this chapter, a least square method (LSM_{oc}) for direct transcription of optimal control problems is presented which is based on the L^2-minimization of the residual in state dynamics. We develop this method in the framework of WRM so that the optimality analysis presented in Chapter III directly applies. The equivalence conditions are derived and the relationship between the costates and the KKT multipliers of the nonlinear programming problem is established. Further, numerical implementation of LSM_{oc} using B-Splines as approximating functions is described in detail. A global polynomial version of LSM_{oc} is developed by taking the trial functions as Lagrange interpolating polynomials. Examples problems are solved using LSM_{oc} transcription and numerical convergence is demonstrated as the order of approximation increases.

A. Direct Transcription Formulation

In LSM_{oc}, the state dynamics is approximated based on the following theorem,

Theorem A.1 *Consider an initial value problem (IVP),*

$$\mathbf{r}(t) = \mathbf{g}(\mathbf{z}(t)) - \dot{\mathbf{z}}(t) = 0, \quad t \in [0,1]; \quad \mathbf{z}(0) = \mathbf{a}.$$

Let $\mathbb{H}$ be an ∞-dimensional Hilbert space equipped with a norm $\|\cdot\|_{\mathbb{H}}$ and an inner-product $\langle\cdot,\cdot\rangle_{\mathbb{H}}$, spanned by a set of linearly independent basis functions $\{\phi_j(t), t \in$

$[0,1]\}_{j=1}^{\infty}$. *If* $\mathbf{z}(t) = \sum_{j=1}^{\infty} \alpha_j \phi_j(t) \in \mathbb{H}$, *then* $\mathbf{z}(t)$ *is the stationary solution of the functional,*

$$J = \| \mathbf{r} \|_{\mathbb{H}}^2 + \nu(\mathbf{z}(0) - \mathbf{a}), \tag{5.1}$$

where ν *is a Lagrange multiplier.*

Proof The stationary conditions are given by,

$$\langle \mathbf{r}, \frac{\partial \mathbf{r}}{\partial \alpha_j} \rangle_{\mathbb{H}} + \frac{1}{2} \nu \phi_j(0) = 0, \quad j = 1, .., \infty, \tag{5.2}$$

$$\mathbf{z}(0) = \mathbf{a}, \tag{5.3}$$

which are trivially satisfied with $\nu = 0$. ∎

In the framework of WRM, conditions given in Eqn. (5.2) are applied by truncating the infinite terms to a finite number $N_{\mathbf{x}}$ and selecting the test functions as,

$$W_j(t) = \frac{\partial \mathbf{r}(t)}{\partial \alpha_j}. \tag{5.4}$$

Here $\mathbf{r}(t)$ is the residual in state dynamics defined as,

$$\mathbf{r}(t) = (\tau_f - \tau_0) \mathbf{f}(\widehat{\mathbf{x}}(t), \widehat{\mathbf{u}}(t)) - \dot{\widehat{\mathbf{x}}}(t), \tag{5.5}$$

so that,

$$W_j(t) = \frac{\partial \mathbf{r}(t)}{\partial \alpha_j} = [(\tau_f - \tau_0) \mathbf{f}_{\mathbf{x}}(\widehat{\mathbf{x}}(t), \widehat{\mathbf{u}}(t)) \phi_j^{\mathbf{x}}(t) - \mathbf{I} \dot{\phi}_j^{\mathbf{x}}(t)]. \tag{5.6}$$

B. Nonlinear Programming Problem

Using Eqn. (5.6), the nonlinear programming problem defined in Chapter III Section B takes the form: Determine $\{\alpha_k \in \mathbb{R}^n\}_{k=1}^{N_{\mathbf{x}}}, \{\beta_k \in \mathbb{R}^m\}_{k=1}^{N_{\mathbf{u}}}, \{\varsigma_k \in \mathbb{R}^q\}_{k=1}^{N_{\mathbf{s}}}$, $\nu_0 \in \mathbb{R}^a$,

$\nu_1 \in \mathbb{R}^{n-a}$ and time instances τ_0 and τ_f, that minimize the cost,

$$\widehat{J} = \widehat{\Psi}(\alpha_k, \phi_k^{\mathbf{x}}(0), \phi_k^{\mathbf{x}}(1), \tau_0, \tau_f), \tag{5.7}$$

subject to the constraints,

$$\int_0^1 [(\tau_f - \tau_0)\widehat{\mathbf{f}}_{\mathbf{x}}^T(\alpha_k, \beta_k, \phi_k^{\mathbf{x}}, \phi_k^{\mathbf{u}})\phi_j^{\mathbf{x}} - \mathbf{I}\dot{\phi}_j^{\mathbf{x}}][(\tau_f - \tau_0)\widehat{\mathbf{f}}(\alpha_k, \beta_k, \phi_k^{\mathbf{x}}, \phi_k^{\mathbf{u}}) - \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \dot{\phi}_k^{\mathbf{x}}]dt$$
$$+\frac{1}{2}K_0^T\nu_0\phi_j^{\mathbf{x}}(0) + \frac{1}{2}K_1^T\nu_1\phi_j^{\mathbf{x}}(1) = 0, \tag{5.8}$$

$$K_0\sum_{k=1}^{N_{\mathbf{x}}} \alpha_k\phi_k^{\mathbf{x}}(0) - \mathbf{x}_0 = 0, \quad K_1\sum_{k=1}^{N_{\mathbf{x}}} \alpha_k\phi_k^{\mathbf{x}}(1) - \mathbf{x}_f = 0, \tag{5.9}$$

$$\widehat{\psi}(\alpha_k, \phi_k^{\mathbf{x}}(0), \phi_k^{\mathbf{x}}(1), \tau_0, \tau_f) = 0, \tag{5.10}$$

$$\int_0^1 [\widehat{\mathbf{h}}(\alpha_k, \beta_k, \phi_k^{\mathbf{x}}, \phi_k^{\mathbf{u}}) + \sum_{k=1}^{N_{\mathbf{s}}} \varsigma_k\phi_k^{\mathbf{s}} \circ \sum_{l=1}^{N_{\mathbf{s}}} \varsigma_l\phi_l^{\mathbf{s}}]\phi_p^{\mathbf{s}} = 0, \tag{5.11}$$

where $j = 1, .., N_x$ and $p = 1, .., N_s$. This approximation scheme as defined by Eqns. (5.7)-(5.11) represents the least square method for optimal control.

Next, we derive the equivalence conditions and costate estimation results for this method.

C. Equivalence Conditions

With regard to the equivalence conditions for LSM_{oc}, we make the following observations form Chapter III Section E:

1. Conditions in Eqns. (3.54)-(3.57) are satisfied in the limit because because the residual $\mathbf{r}(t) \to 0$ as the order of approximation increases.

2. Since not all basis functions $\phi_k^{\mathbf{x}}|_{k=1}^{N_{\mathbf{x}}}$ are zero at the boundaries, the conditions

in Eqns. (3.52) and (3.53) require the following to be satisfied,

$$\sum_{j=1}^{N_{\mathbf{x}}} W_j(0)\gamma_j = (\tau_f - \tau_0)\widehat{\mathbf{f}}_{\mathbf{x}} \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j \phi_j^{\mathbf{x}}(0) - \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j \dot{\phi}_j^{\mathbf{x}}(0) = -[\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}_{\mathbf{x}(0)}^T \eta + \mu_0 K_0], \tag{5.12}$$

$$\sum_{j=1}^{N_{\mathbf{x}}} W_j(1)\gamma_j = (\tau_f - \tau_0)\widehat{\mathbf{f}}_{\mathbf{x}} \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j \phi_j^{\mathbf{x}}(1) - \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j \dot{\phi}_j^{\mathbf{x}}(1) = [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)}^T \eta + \mu_1 K_1]. \tag{5.13}$$

3. Since ν_0 and ν_1 are unknowns in the LSM_{oc} formulation, conditions in Eqn. (3.58) need special attention. To understand their significance, consider the Eqn. 5.6 and write,

$$\sum_{j=1}^{N_{\mathbf{x}}} W_j \widetilde{\gamma}_j = (\tau_f - \tau_0)\mathbf{f}_{\mathbf{x}} \sum_{j=1}^{N_{\mathbf{x}}} \widetilde{\gamma}_j \phi_j^{\mathbf{x}} - \sum_{j=1}^{N_{\mathbf{x}}} \widetilde{\gamma}_j \dot{\phi}_j^{\mathbf{x}} \tag{5.14}$$

Let $\sum_{j=1}^{N_{\mathbf{x}}} \widetilde{\gamma}_j \phi_j^{\mathbf{x}}(t) = \widehat{\rho}(t)$. Then Eqn. (5.14) represents an approximation of the following differential equation,

$$(\tau_f - \tau_0)\mathbf{f}_{\mathbf{x}} \rho(t) - \dot{\rho}(t) = \lambda(t), \tag{5.15}$$

where $\rho(t)$ is referred as "auxiliary costates". However, notice that the above differential equation lacks boundary conditions. These boundary conditions are supplied by the extra conditions found in the KKT system, i.e.,

$$K_0 \sum_{j=1}^{N_{\mathbf{x}}} \widetilde{\gamma}_j \phi_j^{\mathbf{x}}(0) = 0, \qquad K_1 \sum_{j=1}^{N_{\mathbf{x}}} \widetilde{\gamma}_j \phi_j^{\mathbf{x}}(1) = 0. \tag{5.16}$$

Thus, we see that the conditions in Eqn. (3.58) do not introduce any inconsistency in the KKT system of LSM_{oc} formulation. They are absorbed in the definition of auxiliary costates and do not form a part of equivalence conditions for LSM_{oc}.

D. Costate Estimates

The equivalence conditions defined in Eqns. (5.12) and (5.13) provide the mapping between KKT multipliers $\{\gamma_k\}$, and the costate approximation $\widehat{\lambda}(t)$. Similarly, a mapping exists between $\{\xi(t), \mu_0, \mu_1, \eta\}$ and the corresponding KKT multipliers. We summarize the costate estimation results for the LSM_{oc} via the following theorem:

Theorem D.1 (Costate Mapping Theorem for the LSM_{oc}) *Assume that an optimal control problem is solved using the LSM_{oc} and the equivalence conditions hold. Then, the estimates of the costates $\widehat{\lambda}(t)$, and the Lagrange multiplier functions $\widehat{\xi}(t)$ can be obtained from the KKT multipliers (γ_k, ζ_k) of the associated NLP as,*

$$\begin{aligned} \widehat{\lambda}(t) &= (\tau_f - \tau_0)\widehat{\mathbf{f}}_{\mathbf{x}} \sum_{k=1}^{N_{\mathbf{x}}} \gamma_k \phi_k^{\mathbf{x}}(t) - \sum_{k=1}^{N_{\mathbf{x}}} \gamma_k \dot{\phi}_k^{\mathbf{x}}(t) \\ \widehat{\xi}(t) &= \sum_{k=1}^{N_{\mathbf{s}}} \zeta_k \phi_k^{\mathbf{s}}(t). \end{aligned} \tag{5.17}$$

Proof The solution to Problem $\mathcal{M}_{\lambda\phi}$ exists by assumption. Since equivalence conditions hold, the results in Chapter III Section F are valid. ∎

E. LSM_{oc} Using B-Spline Approximation

In this section, we describe the implementation details of LSM_{oc} using B-Splines as approximating functions. The integrals in Eqns. (5.8) and (5.11) are evaluated numerically by using a numerical quadrature scheme which changes the structure of the resulting NLP and in turn the KKT conditions. However, it is shown that the results on costate estimation and equivalence conditions can still be derived.

Consider an optimal control problem $\mathcal{A}$, to determine the state-control pair

$\{\mathbf{x}(t) \in \mathbb{R}^n, \mathbf{u}(t) \in \mathbb{R}^m; t \in [0,1]\}$, that minimizes,

$$J = \Psi(\mathbf{x}(1)),$$

subject to,

$$\dot{\mathbf{x}}(t) = F(\mathbf{x}(t), \mathbf{u}(t)), \quad \mathbf{x}(0) = \mathbf{x}_0,$$
$$\psi(\mathbf{x}(1)) = 0,$$

where, $\Psi : \mathbb{R}^n \to \mathbb{R}, F : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n, \psi : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^p$. To solve this problem using the LSM_{oc}, we approximate the state and control trajectories as B-Splines. So that,

$$\widehat{\mathbf{x}}(t) = \sum_{k=1}^{N_c} \alpha_k B^{k,r}(t), \qquad \widehat{\mathbf{u}}(t) = \sum_{k=1}^{N_c} \beta_k B^{k,r}(t),$$

for some $N, r, s, N_c = N(r-s)+s$, and breakpoints $0 = T_0 < T_1 < .. < T_N = 1$ with $\Omega_i := [T_{i-1}, T_i]$. We define N_q number of LG points $\{t_{ij}\}_{j=1}^{N_q}$ with corresponding quadrature weights $\{w_{ij}\}$ suitably mapped over each domain $\{\Omega_i\}_{i=1}^{N}$(see figure on page 61 in Chapter IV). A nonlinear programming problem $\mathcal{A}_N$ is formulated based on Section B in this chapter and using LG quadrature scheme for integration. The problem $\mathcal{A}_N$ is to determine $\{\alpha_k \in \mathbb{R}^n, \beta_k \in \mathbb{R}^m\}_{k=1}^{N}$ that minimize,

$$J = \widehat{\Psi}(\alpha_k, B^{k,r}(1)),$$

subject to the constraints,

$$\sum_{i=1}^{N}\sum_{j=1}^{N_q} w_{ij}[(\widehat{F}_{\mathbf{x}}^T B^{l,r})_{ij} - \mathbf{I}\dot{B}_{ij}^{l,r}][\widehat{F}_{ij} - \sum_{k=1}^{N_c} \alpha_k \dot{B}_{ij}^{k,r}] + \frac{1}{2}\nu B_0^{l,r} = 0, \tag{5.18}$$

$$\sum_{k=1}^{N_c} \alpha_k B_0^{k,r} - \mathbf{x}_0 = 0, \quad \widehat{\psi} = 0, \tag{5.19}$$

where $l = 1, 2, .., N_c$. Here $(\cdot)_{ij}$ denotes the evaluation of the underlying expression at time t_{ij}. Similarly, $B_0^{k,r}, B_1^{k,r}$ denote $B^{k,r}(0), B^{k,r}(1)$ respectively. For the prescribed approximation scheme the following lemmas hold,

Lemma E.0.1 *If P^n represents the space of all polynomials of degree less than or equal to n, then for $i = 1, .., N$,*

$$\int_{T_{i-1}}^{T_i} p(t)dt = \sum_{j=1}^{N_q} w_{ij} p(t_{ij}), \quad \forall p(t) \in P^{2N_q-1}.$$

Lemma E.0.2 *For $i = 1, .., N$, let $f : [T_{i-1}, T_i] \to \mathbb{R}$ be a continuous function. Then, for every $\delta > 0$, there exists N_f such that $\forall N_q > N_f$,*

$$\left| \int_{T_{i-1}}^{T_i} f(t)dt - \sum_{j=1}^{N_q} w_{ij} p(t_{ij}) \right| \leq \delta.$$

Proof Let $\Delta T_i = T_i - T_{i-1}$. Using Kreyszig.1989 approximation theorem, every $\delta > 0$, there exists N_p such that $\forall n > N_p$,

$$|f(t) - p^n(t)| \leq \frac{1}{\Delta T_i}\delta,$$

where p^n is a polynomial of order n. Integrating both sides and using Lemma E.0.1, we have $N_q \geq \frac{N_p+1}{2}$. ∎

Theorem E.1 *For every $\delta > 0$, N_q can be chosen such that,*

$$\sum_{i=1}^{N} \sum_{j=1}^{N_q} w_{ij} \left[\dot{B}_{ij}^{k,r} \dot{B}_{ij}^{l,r} + \ddot{B}_{ij}^{l,r} B_{ij}^{k,r} \right] = (B^{k,r} \dot{B}^{l,r})|_0^1, \tag{5.20}$$

$$\left| \sum_{i=1}^{N} \sum_{j=1}^{N_q} w_{ij} \left\{ (\tau_f \widehat{F}_{\mathbf{x}} B^{l,r})_{ij} \dot{B}_{ij}^{k,r} + (\tau_f \dot{\widehat{F}}_{\mathbf{x}} B^{l,r})_{ij} B_{ij}^{k,r} \right\} - (\tau_f \widehat{F}_{\mathbf{x}} B^{l,r} B^{k,r})|_0^1 \right| \leq \delta. \tag{5.21}$$

Proof Using integration by parts,

$$\int_0^1 (\dot{B}^{k,r}\dot{B}^{l,r} + \ddot{B}^{l,r}B^{k,r})dt = (B^{k,r}\dot{B}^{l,r})|_0^1 \tag{5.22}$$

$$\int_0^1 \{(\tau_f \widehat{F}_{\mathbf{x}} B^{l,r})\dot{B}^{k,r} + (\tau_f \dot{\widehat{F}}_{\mathbf{x}} B^{l,r})B^{k,r}\}dt = (\tau_f \widehat{F}_{\mathbf{x}} B^{l,r}B^{k,r})|_0^1. \tag{5.23}$$

By definition, $\{\dot{B}^{k,r}\dot{B}^{l,r}, \ddot{B}^{l,r}B^{k,r} \in P^{2(r-2)}; t \in \Omega_i\}_{k=1}^{N_c}$ over domains $\{\Omega_i\}_{i=1}^{N}$. Using Eqn. (5.22) and Lemma E.0.1, Eqn. (5.20) holds for $N_q > r$. From Eqn. (5.23) and Lemma (E.0.2), there exists some N_f such that Eqn. (5.21) holds for $N_q \geq N_f$. Thus we can choose $N_q = \text{max.}\{r, N_f\}$. ∎

Next, we derive the KKT conditions for problem $\mathcal{A}_N$. The augmented cost is defined as,

$$\begin{aligned} J' = &\sum_{l=1}^{N_c} \gamma_l^T \left[\sum_{i=1}^{N} \left\{\sum_{j=1}^{N_q} w_{ij}[(\widehat{F}_{\mathbf{x}}^T B^{l,r})_{ij} - \mathbf{I}\dot{B}_{ij}^{l,r}][\widehat{F} - \sum_{k=1}^{N_c} \alpha_k \dot{B}_{ij}^{k,r}]\right\}\right] \\ &+ \sum_{l=1}^{N_c} \gamma_l^T \left[\frac{1}{2}\nu B_0^{l,r}\right] + \mu^T(\sum_{k=1}^{N_c} \alpha_k B_0^{k,r} - \mathbf{x}_0) + \widehat{\Psi} + \eta^T \widehat{\psi}, \end{aligned} \tag{5.24}$$

where $\gamma_l \in \mathbb{R}^n$, $\mu \in \mathbb{R}^n$ and $\eta \in \mathbb{R}^p$ are the KKT multipliers associated with the constraints given by Eqn. Eqn. (5.18), Eqn. (5.19). The KKT conditions $\mathcal{A}_{N\lambda}$ are derived by setting the partial derivatives of J' with respect to free variables equal to zero. Since the NLP is solved numerically, the KKT conditions are satisfied within a specified tolerance δ. So that for $m = 1, .., N$,

$$\begin{aligned} \left|\frac{\partial J'}{\partial \alpha_m}\right| = \Bigg| &\sum_{l=1}^{N_c} \Bigg[\sum_{i=1}^{N}\sum_{j=1}^{N_q} w_{ij}\{\dot{B}_{ij}^{m,r}\dot{B}_{ij}^{l,r} - \tau_f(\widehat{F}_{\mathbf{x}}^T)_{ij}B_{ij}^{m,r}\dot{B}_{ij}^{l,r} - \tau_f(\widehat{F}_{\mathbf{x}})_{ij}B_{ij}^{l,r}\dot{B}_{ij}^{m,r} \\ &+\tau_f^2(\widehat{F}_{\mathbf{x}}^T\widehat{F}_{\mathbf{x}})_{ij}B_{ij}^{m,r}B_{ij}^{l,r}\}\Bigg]\gamma_l \\ &+ \sum_{l=1}^{N_c}\sum_{i=1}^{N}\sum_{j=1}^{N_q} w_{ij}[\tau_f(\mathbf{r}^T\widehat{F}_{\mathbf{xx}})_{ij}B_{ij}^{m,r}B_{ij}^{l,r}]\gamma_l + \mu B_0^{m,r} + [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)}^T\eta]B_1^{m,r}\Bigg| \leq \delta. \end{aligned} \tag{5.25}$$

Using Eqns. (5.20), (5.21) and re-arranging, we get,

$$\Bigg|\sum_{l=1}^{N_c}\sum_{i=1}^{N}\sum_{j=1}^{N_q} w_{ij}[-\ddot{B}_{ij}^{l,r}\gamma_l - \tau_f(\widehat{F}_{\mathbf{x}}^T)_{ij}\dot{B}_{ij}^{l,r}\gamma_l + (\tau_f\widehat{F}_{\mathbf{x}}\dot{B}^{l,r})_{ij}\gamma_l + \tau_f^2(\widehat{F}_{\mathbf{x}}^T\widehat{F}_{\mathbf{x}})_{ij}B_{ij}^{l,r}\gamma_l]B_{ij}^{m,r}$$
$$+\sum_{l=1}^{N_c}\sum_{i=1}^{N}\sum_{j=1}^{N_q} w_{ij}[(\tau_f\mathbf{r}^T\widehat{F}_{\mathbf{xx}})_{ij}B_{ij}^{l,r}\gamma_l]B_{ij}^{m,r} + [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)}^T\eta]B_1^{m,r}$$
$$-[\tau_f(\widehat{F}_{\mathbf{x}})_1\sum_{l=1}^{N_c}\gamma_l B_1^{l,r} - \sum_{l=1}^{N_c}\gamma_l\dot{B}_1^{l,r}]B_1^{m,r} + \mu B_0^{m,r} \tag{5.26}$$
$$+[\tau_f(\widehat{F}_{\mathbf{x}})_0\sum_{l=1}^{N_c}\gamma_l B_0^{l,r} - \sum_{l=1}^{N_c}\gamma_l\dot{B}_0^{l,r}]B_0^{m,r}\Bigg| \leq \delta. \tag{5.27}$$

also,

$$\left|\frac{\partial J'}{\partial\mu}\right| = \left|\sum_{k=1}^{N_c}\alpha_k B_0^{k,r} - \mathbf{x}_0\right| \leq \delta, \tag{5.28}$$

$$\left|\frac{\partial J'}{\partial\beta_m}\right| = \Bigg|\sum_{l=1}^{N_c}\sum_{i=1}^{N}\sum_{j=1}^{N_q} w_{ij}\tau_f[\tau_f(\widehat{F}_{\mathbf{u}}^T\widehat{F}_{\mathbf{x}})_{ij}\gamma_l B_{ij}^{l,r} - (\widehat{F}_{\mathbf{u}}^T)_{ij}\gamma_l\dot{B}_{ij}^{l,r}]B_{ij}^{m,r}$$
$$+\sum_{l=1}^{N_c}\sum_{i=1}^{N}\sum_{j=1}^{N_q} w_{ij}[\tau_f(\mathbf{r}^T\widehat{F}_{\mathbf{xu}})_{ij}\gamma_l B_{ij}^{l,r}]B_{ij}^{m,r}\Bigg| \leq \delta, \tag{5.29}$$

$$\left|\frac{\partial J'}{\partial\gamma_m}\right| = \left|\sum_{i=1}^{N}\{\sum_{j=1}^{N_q} w_{ij}[(\widehat{F}_{\mathbf{x}}^T B^{m,r})_{ij} - \mathbf{I}\dot{B}_{ij}^{m,r}][\widehat{F} - \sum_{k=1}^{N_c}\alpha_k\dot{B}_{ij}^{k,r}]\} + \frac{1}{2}\nu B_0^{m,r}\right| \leq \delta, \tag{5.30}$$

$$\left|\frac{\partial J'}{\partial\nu}\right| = \left|\sum_{l=1}^{N_c}\gamma_l B_0^{l,r}\right| \leq \delta, \quad \left|\frac{\partial J'}{\partial\eta}\right| = \left|\widehat{\psi}\right| \leq \delta. \tag{5.31}$$

Thus, Eqns. (5.26)-(5.31) constitute the KKT conditions $\mathcal{A}_{N\lambda}$. Based on Chapter III Section D and using numerical quadrature, the discretized first-order optimality

conditions $\mathcal{A}_{\lambda N}$ are,

$$\left|\sum_{i=1}^{N}\{\sum_{j=1}^{N_q} w_{ij}[(\widehat{F}_{\mathbf{x}}^T B^{l,r})_{ij} - \mathbf{I}\dot{B}_{ij}^{l,r}][\widehat{F} - \sum_{k=1}^{N_c}\alpha_k \dot{B}_{ij}^{k,r}]\} + \frac{1}{2}\pi B_0^{l,r}\right| \leq \delta,$$

$$\left|\sum_{k=1}^{N_c}\alpha_k B_0^{k,r} - \mathbf{x}_0\right| \leq \delta,$$

$$\left|\sum_{l=1}^{N_c}\sum_{i=1}^{N}\sum_{j=1}^{N_q} w_{ij}[-\widetilde{\gamma}_l \ddot{B}_{ij}^{l,r} - \tau_f(\widehat{F}_{\mathbf{x}}^T)_{ij}\widetilde{\gamma}_l \dot{B}_{ij}^{l,r} + (\tau_f \widehat{F}_{\mathbf{x}}\dot{\widetilde{\gamma}}_l B_{ij}^{l,r}) + \tau_f^2(\widehat{F}_{\mathbf{x}}^T\widehat{F}_{\mathbf{x}})_{ij}\widetilde{\gamma}_l B_{ij}^{l,r}]B_{ij}^{m,r}\right| \leq \delta,$$

$$\left|\sum_{l=1}^{N_c}\sum_{i=1}^{N}\sum_{j=1}^{N_q} w_{ij}\tau_f[\tau_f(\widehat{F}_{\mathbf{u}}^T\widehat{F}_{\mathbf{x}})_{ij}\widetilde{\gamma}_l B_{ij}^{l,r} - (\widehat{F}_{\mathbf{u}}^T)_{ij}\widetilde{\gamma}_l \dot{B}_{ij}^{l,r}]B_{ij}^{m,r}\right| \leq \delta,$$

$$\left|\widehat{\psi}\right| \leq \delta, \qquad \left|\sum_{k=1}^{N_c}\widetilde{\gamma}_k B_0^{k,r}\right| \leq \delta,$$

$$\left|\tau_f(\widehat{F}_{\mathbf{x}})_0\sum_{l=1}^{N_c}\widetilde{\gamma}_l B_0^{l,r} - \sum_{l=1}^{N_c}\widetilde{\gamma}_l \dot{B}_0^{l,r} + \kappa\right| \leq \delta,$$

$$\left|\tau_f(\widehat{F}_{\mathbf{x}})_1\sum_{l=1}^{N_c}\gamma_l B_1^{l,r} - \sum_{l=1}^{N_c}\gamma_l \dot{B}_1^{l,r} - [\Psi_{\mathbf{x}(1)} + \upsilon^T\psi_{\mathbf{x}(1)}]\right| \leq \delta. \tag{5.32}$$

for $m = 1, .., N$. Comparing $\mathcal{A}_{N\lambda}$ and $\mathcal{A}_{\lambda N}$, we see that the results from Section B and C hold. The equivalence conditions are,

$$\left|\tau_f(\widehat{F}_{\mathbf{x}})_0\sum_{l=1}^{N_c}\gamma_l B_0^{l,r} - \sum_{l=1}^{N_c}\gamma_l \dot{B}_0^{l,r} + \mu\right| \leq \delta, \tag{5.33}$$

$$\left|\tau_f(\widehat{F}_{\mathbf{x}})_1\sum_{l=1}^{N_c}\gamma_l B_1^{l,r} - \sum_{l=1}^{N_c}\gamma_l \dot{B}_1^{l,r} - [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)}^T\eta]\right| \leq \delta. \tag{5.34}$$

The costates can be estimates as,

$$\widehat{\lambda}(t) \;=\; \tau_f\widehat{F}_{\mathbf{x}}\sum_{l=1}^{N_c}\gamma_l B^{l,r}(t) - \sum_{l=1}^{N_c}\gamma_l \dot{B}^{l,r}(t). \tag{5.35}$$

F. LSM_{oc} Using Global Interpolating Polynomials: s-LSM_{oc}

In this section we formulate the LSM_{oc} using global interpolating Lagrange polynomials. Further, we choose the quadrature nodes to be same as the interpolation nodes. As we shall see later in this chapter, use of global polynomials as test and trial functions results in achieving "spectral accuracy", i.e. very high rates of convergence, with $\text{LSM}_{\text{oc}}$for smooth problems. Therefore, we name this version of LSM_{oc} as spectral-LSM_{oc}, denoted as s-LSM_{oc}.

Let us take the interpolation/quadrature nodes to be N number of LGL points $\{t_k\}_{k=1}^N$ suitably mapped on the interval $[0,1]$ with corresponding quadrature weights $\{w_k\}_{k=1}^N$. The corresponding Lagrange interpolation basis is $\{\mathcal{L}_i(t)\}_{i=1}^N$. All trajectory variables are approximated using the Lagrange basis. So that,

$$\phi_i^{\mathbf{x}}(t) = \phi_i^{\mathbf{u}}(t) = \phi_i^{\mathbf{s}}(t) = \mathcal{L}_i(t); \quad t \in [-1,1]. \tag{5.36}$$

Using the Kronecker delta property of Lagrange polynomials $L_j(t_k) = \delta_{jk}$, we get for $i = 1, 2, .., N$,

$$\mathbf{x}_i = \widehat{\mathbf{x}}(t_i) = \sum_{k=1}^{N} \alpha_k L_k(t_i) = \alpha_i, \quad \mathbf{u}_i = \widehat{\mathbf{u}}(t_i) = \sum_{k=1}^{N} \beta_k L_k(t_i) = \beta_i, \tag{5.37}$$

$$\mathbf{s}_i = \widehat{\mathbf{s}}(t_i) = \sum_{k=1}^{N} \varsigma_k L_k(t_i) = \varsigma_i. \tag{5.38}$$

Further, a differentiation matrix D is defined as,

$$D_{ij} = \dot{\mathcal{L}}_j(t_i), \tag{5.39}$$

so that,

$$\dot{\widehat{\mathbf{x}}}(t_i) = \dot{\mathbf{x}}_i = \sum_{j=1}^{N} D_{ij}\mathbf{x}_j; \quad i = 1, .., N. \tag{5.40}$$

Using Eqns. (5.36)-(5.40) with the NLP formulation given in section B of this chapter, the s-LSM$_{\text{oc}}$ transcription of an optimal control problem is defined as following: Determine $\{\mathbf{x}_k \in \mathbb{R}^n\}_{k=1}^N, \{\mathbf{u}_k \in \mathbb{R}^m\}_{k=1}^N, \{\mathbf{s}_k \in \mathbb{R}^q\}_{k=1}^N$, $\nu_0 \in \mathbb{R}^a$, $\nu_1 \in \mathbb{R}^{n-a}$ and time instances τ_0 and τ_f, that minimize the cost,

$$\widehat{J} = \widehat{\Psi}(\mathbf{x}_1, \mathbf{x}_N, \tau_0, \tau_f), \tag{5.41}$$

subject to the constraints,

$$\widehat{\mathbf{f}}_{\mathbf{x}}^T(\mathbf{x}_1, \mathbf{u}_1)R_1 - \sum_{k=1}^{N} w_k R_k D_{k1} + \frac{1}{2}K_0^T \nu_0 = 0, \tag{5.42}$$

$$\widehat{\mathbf{f}}_{\mathbf{x}}^T(\mathbf{x}_N, \mathbf{u}_N)R_N - \sum_{k=1}^{N} w_k R_k D_{kN} + \frac{1}{2}K_1^T \nu_1 = 0, \tag{5.43}$$

$$\widehat{\mathbf{f}}_{\mathbf{x}}^T(\mathbf{x}_i, \mathbf{u}_i)R_i - \sum_{k=1}^{N} w_k R_k D_{ki} = 0; \quad i = 2, .., N-1, \tag{5.44}$$

$$K_0\mathbf{x}_1 - \mathbf{x}_0 = 0, \quad K_1\mathbf{x}_N - \mathbf{x}_f = 0, \tag{5.45}$$

$$\widehat{\psi}(\mathbf{x}_1, \mathbf{x}_N, \tau_0, \tau_f) = 0, \tag{5.46}$$

$$\widehat{\mathbf{h}}(\mathbf{x}_j, \mathbf{u}_j) + \mathbf{s}_j^2 = 0; \quad j = 1, .., N, \tag{5.47}$$

where,

$$R_k = \widehat{\mathbf{f}}(\mathbf{x}_k, \mathbf{u}_k) - \sum_{l=1}^{N} D_{kl}\mathbf{x}_l; \quad k = 1, .., N. \tag{5.48}$$

Eqns. (5.41)-(5.47) represent the s-LSM$_{\text{oc}}$ discretization of the original optimal control problem. It can be easily shown that for the given quadrature scheme, the integration

by parts operation is valid. Therefore, the equivalence conditions for s-LSM$_{\text{oc}}$ are,

$$\tau_f \widehat{\mathbf{f}}_{\mathbf{x}}(X_1, \mathbf{u}_1)\gamma_1 - \sum_{k=1}^{N} D_{1k}\gamma_k = -[\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}^T_{\mathbf{x}(0)}\eta + \mu_0 K_0], \tag{5.49}$$

$$\tau_f \widehat{\mathbf{f}}_{\mathbf{x}}(X_N, \mathbf{u}_N)\gamma_N - \sum_{k=1}^{N} D_{Nk}\gamma_k = [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}^T_{\mathbf{x}(1)}\eta + \mu_1 K_1], \tag{5.50}$$

where γ_1, γ_N, $\{\gamma_j\}_{j=2}^{N-1}$ are the KKT multipliers associated with the constraints in Eqns. (5.42), (5.43) and (5.44) respectively. Under the equivalence conditions, the costates can be estimated as,

$$\widehat{\lambda}(t_j) = \tau_f \widehat{\mathbf{f}}_{\mathbf{x}}(\mathbf{x}_j, \mathbf{u}_j)\gamma_j - \sum_{k=1}^{N} D_{jk}\gamma_k. \tag{5.51}$$

G. Numerical Convergence Analysis

In this section, we numerically demonstrate the convergence properties of LSM$_{\text{oc}}$. Four example problems are taken from Chapter IV and results are generated using the same methodology as presented in Chapter IV Section F.

1. Convergence Results for LSM$_{\text{oc}}$

- Example 1: Convergence results for Example 1 are shown in Figure (19). It is seen that higher polynomial order approximation scheme SPL-3r performs better than the low order scheme SPL-34. Further, LEG scheme does not converge for many cases and performs worst.

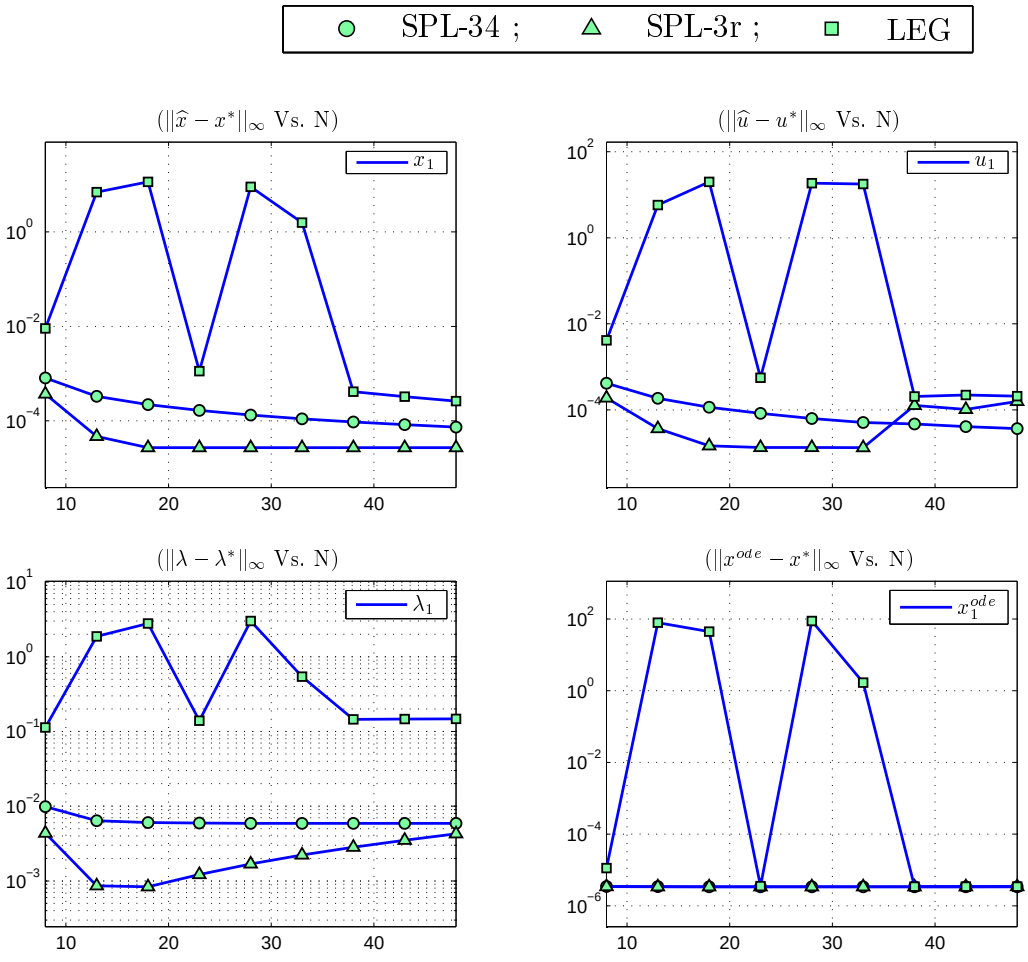


Fig. 19. Convergence of LSM_{oc} solution for Example 1.

- Example 2: Convergence results for Example 2 are shown in Figure (20). All the test cases converge and it is very well seen that the SPL-3r approximation works best for this example.

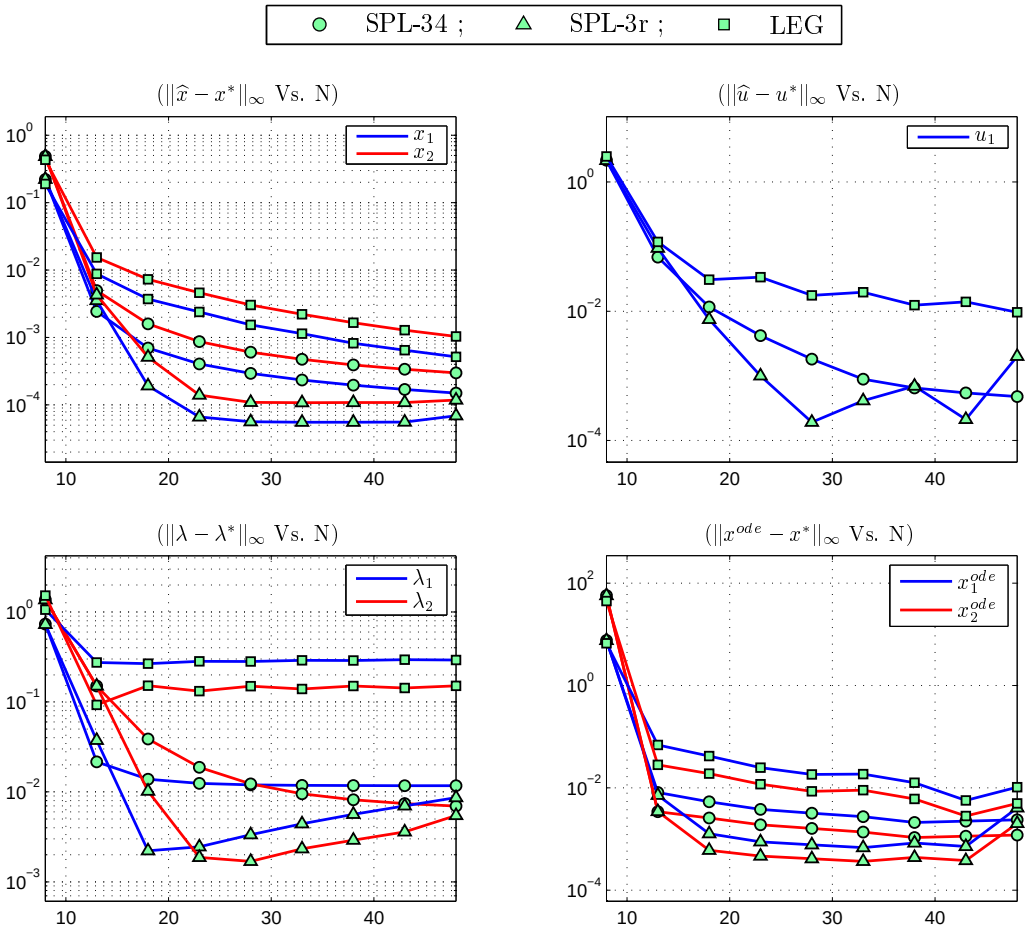


Fig. 20. Convergence of LSM_{oc} solution for Example 2.

- Example 3: This problem is solved using SPL-34 and SPL-3r approximations and the convergence results are depicted in Figure (21). This example shows that the local approximation scheme SPL-34 performs better than the global SPL-3r scheme. This is due to the presence of discontinuities and corners in the solution of this example.

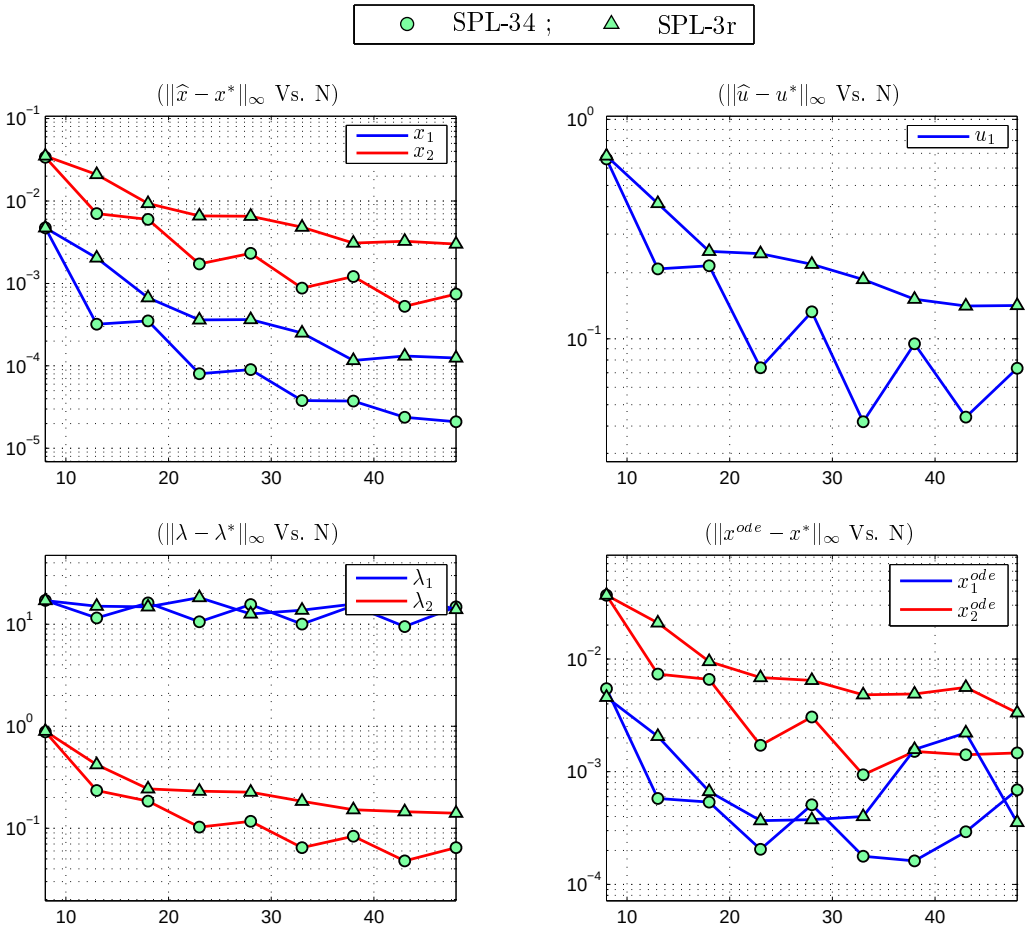


Fig. 21. Convergence of LSM_{oc} solution for Example 3.

- Example 4: Example 4 has a discontinuous control solution. Convergence results in Figure (22) show that SPL-34 gives better solutions than SPL-3r. Also, LSM_{oc} gives quite accurate costate estimates for this example. It was noted in Chapter IV that the MHSP failed to provide accurate costate estimates for this example. This demonstrates that LSM_{oc} has better dual-convergence than MHSP.

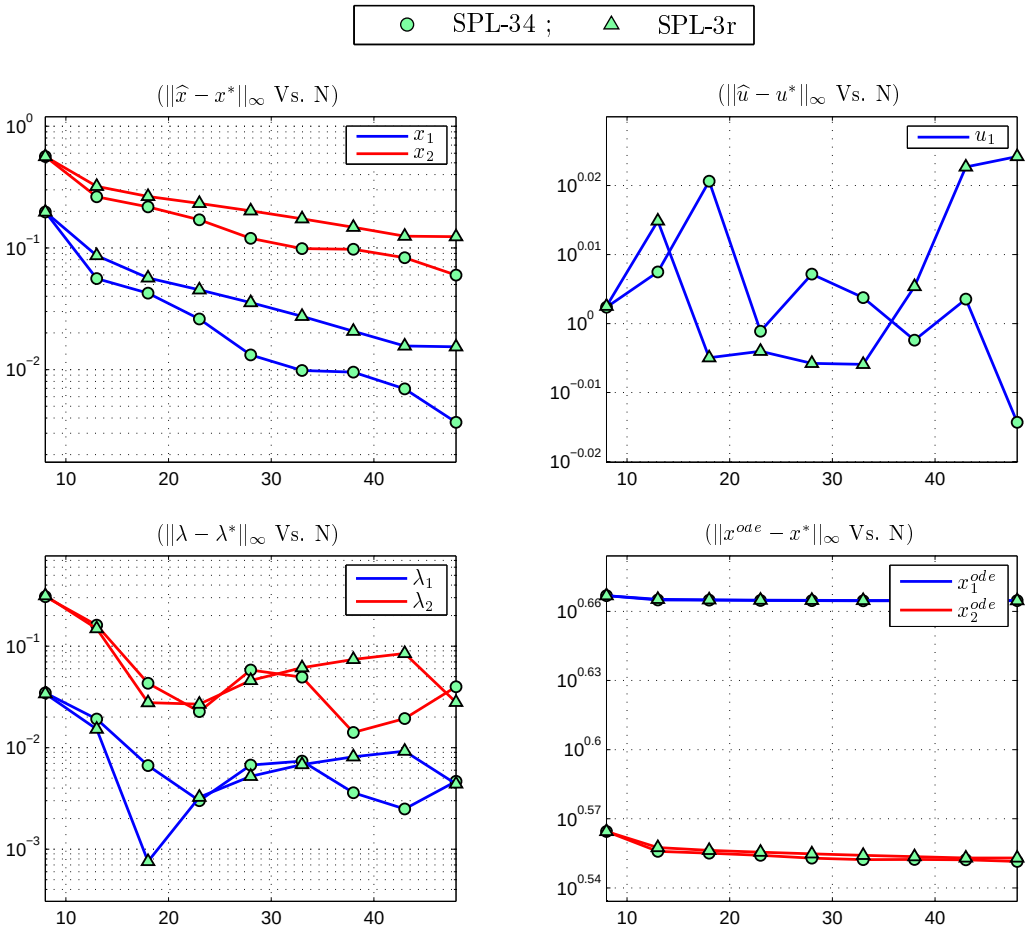


Fig. 22. Convergence of LSM_{oc} solution for Example 4.

2. Convergence Results for s-LSM$_{oc}$

- Example 1: Convergence results for Example 1 are shown in Figure (23). Very high convergence rates are achieved as N goes from 8 to 28. It will be shown in Chapter VIII that these convergence rates are comparable to the existing pseudospectral methods.

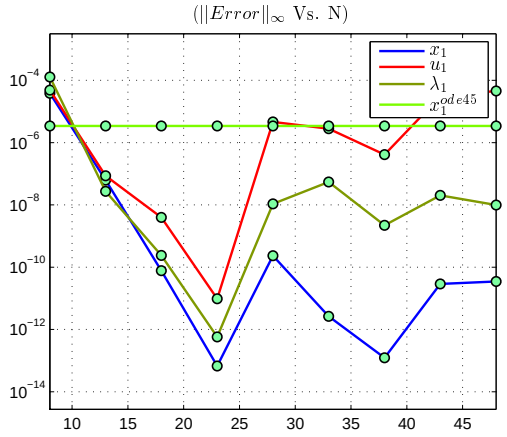


Fig. 23. Convergence of s-LSM$_{oc}$ solution for Example 1.

- Example 2: Convergence results for Example 2 are shown in Figure (24).

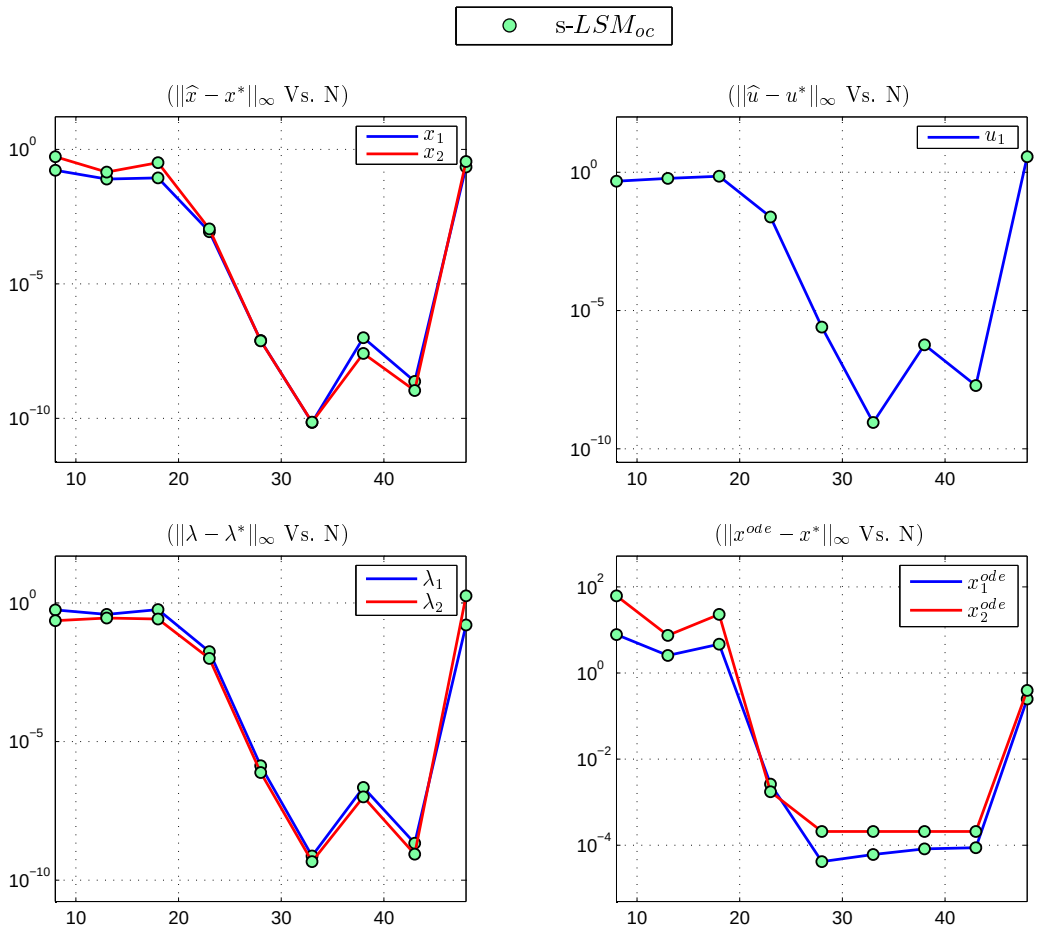


Fig. 24. Convergence of s-LSM$_{oc}$ solution for Example 2.

- Example 3: This problem converges at a slower rate due to the irregularities in its solution. Still, convergence is achieved to a reasonable accuracy using s-LSM$_{oc}$. Results are shown in Figure (25).

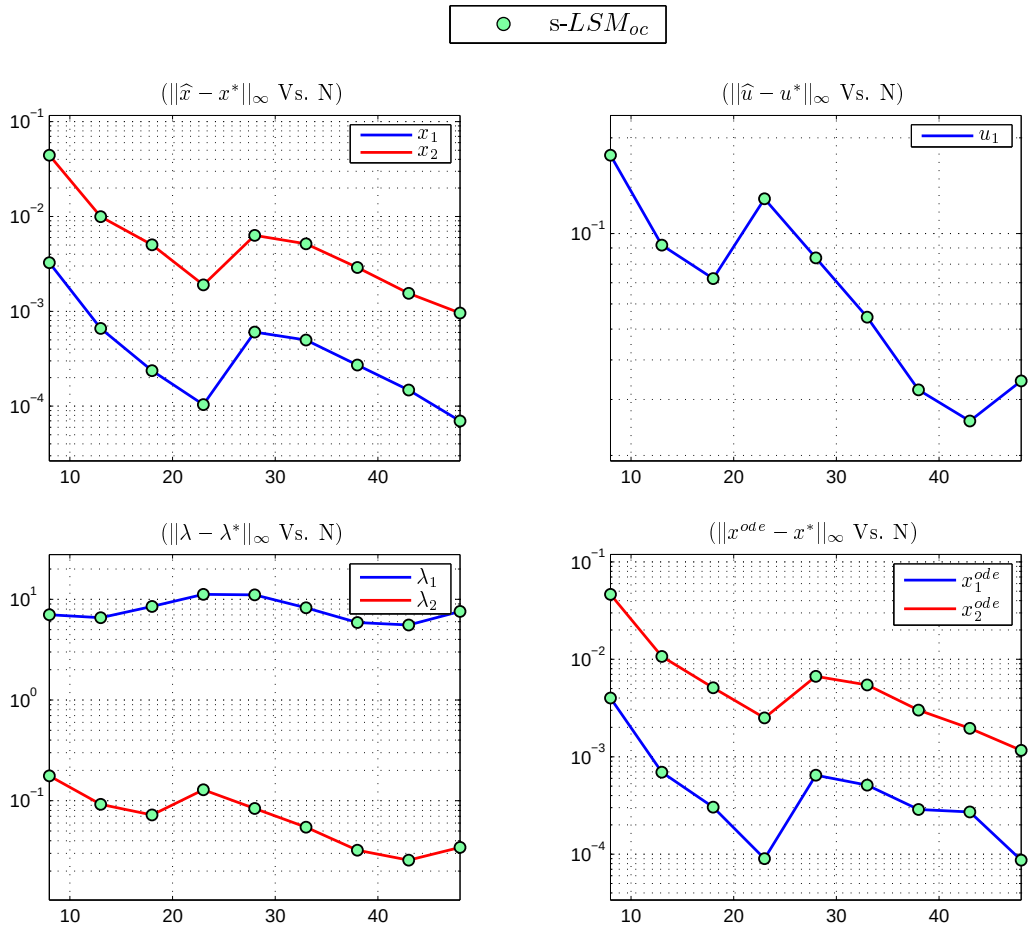


Fig. 25. Convergence of s-LSM$_{oc}$ solution for Example 3.

- Example 4: Example 4 also has a slower but smooth convergence. Results are depicted in Figure (26).

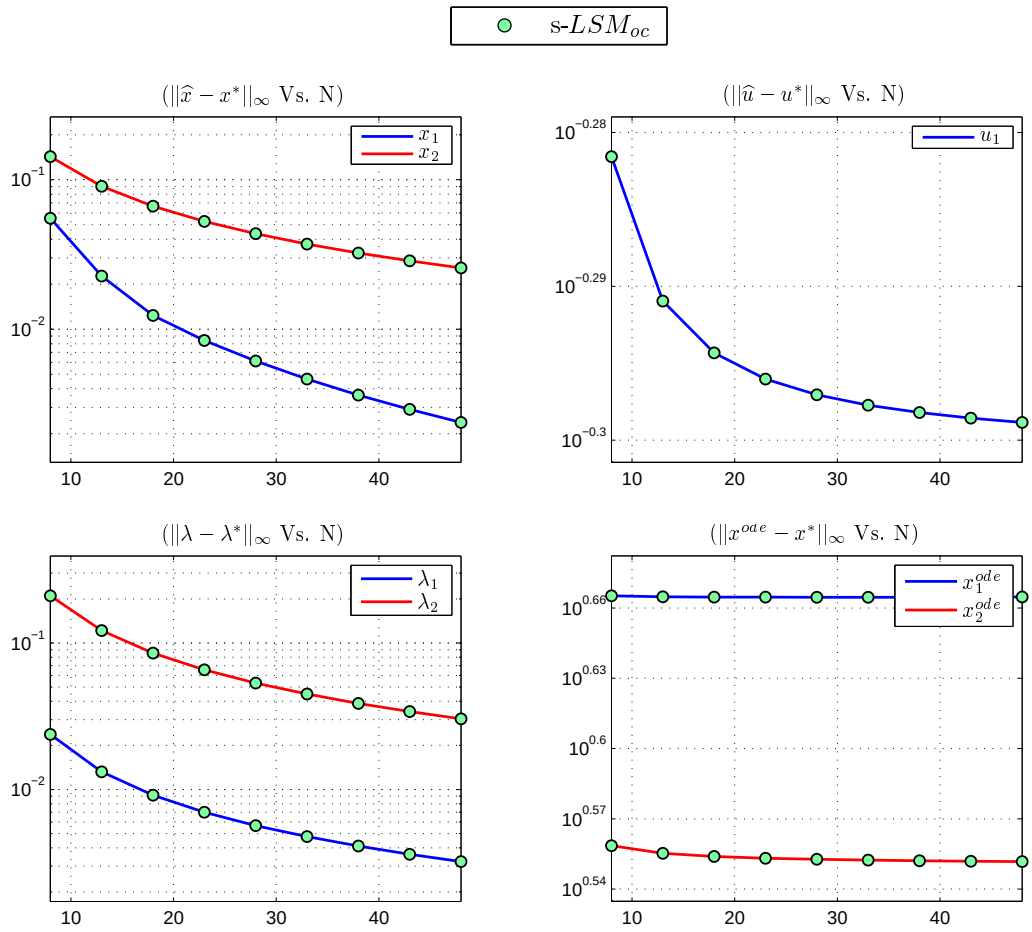


Fig. 26. Convergence of s-LSM$_{oc}$ solution for Example 4.

H. Conclusions

A least-square method for direct transcription of optimal control problems was presented in this chapter. A set of equivalence conditions were derived under which indirect-direct mapping exists and costates can be estimated form the Lagrange multipliers of the nonlinear programming problem. LSM$_{oc}$ was implemented using both

local (B-Splines) and global (Lagrange polynomials) basis functions. Numerical convergence was demonstrated by solving example problems. It was shown that for smooth problems, the global polynomial version of LSM_{oc}, s-LSM_{oc}, exhibits very high rates of convergence.

CHAPTER VI

GENERALIZED MOMENT METHOD FOR OPTIMAL CONTROL

In this chapter, another variation of the WRM is developed by selecting the test functions as the derivatives of the trial functions. This formulation corresponds to the generalized moment method used to solve boundary value problems [51]. We shall see that GMM_{oc} closely resembles LSM_{oc} in its formulation. Its advantage over LSM_{oc} is the ease of implementation. GMM_{oc} method is developed in the framework of WRM so that the optimality analysis presented in Chapter III directly applies. The equivalence conditions are derived and the relationship between the costates and the KKT multipliers of the nonlinear programming problem is established. Examples problems are solved using GMM_{oc} transcription and numerical convergence is demonstrated as the order of approximation increases.

A. Formulation of Nonlinear Programming Problem

In GMM_{oc}, test functions are chosen to be derivatives of the trial functions. So that in the framework of WRM,

$$W_j(t) = \dot{\phi}_j^{\mathbf{x}}(t). \tag{6.1}$$

Using Eqn. (6.1), the nonlinear programming problem defined in Chapter III Section B takes the form: Determine $\{\alpha_k \in \mathbb{R}^n\}_{k=1}^{N_{\mathbf{x}}}, \{\beta_k \in \mathbb{R}^m\}_{k=1}^{N_{\mathbf{u}}}, \{\varsigma_k \in \mathbb{R}^q\}_{k=1}^{N_{\mathbf{s}}}$, $\nu_0 \in \mathbb{R}^a$, $\nu_1 \in \mathbb{R}^{n-a}$ and time instances τ_0 and τ_f, that minimize the cost,

$$\widehat{J} = \widehat{\Psi}(\alpha_k, \phi_k^{\mathbf{x}}(0), \phi_k^{\mathbf{x}}(1), \tau_0, \tau_f), \tag{6.2}$$

subject to the constraints,

$$\int_0^1 [\tau_f \widehat{\mathbf{f}}(\alpha_k, \beta_k, \phi_k^{\mathbf{x}}, \phi_k^{\mathbf{u}}) - \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \dot{\phi}_k^{\mathbf{x}}] \dot{\phi}_j^{\mathbf{x}}(t) dt + \frac{1}{2} K_0^T \nu_0 \phi_j^{\mathbf{x}}(0) + \frac{1}{2} K_1^T \nu_1 \phi_j^{\mathbf{x}}(1) = 0, \tag{6.3}$$

$$K_0 \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \phi_k^{\mathbf{x}}(0) - \mathbf{x}_0 = 0, \quad K_1 \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \phi_k^{\mathbf{x}}(1) - \mathbf{x}_f = 0, \tag{6.4}$$

$$\widehat{\psi}(\alpha_k, \phi_k^{\mathbf{x}}(0), \phi_k^{\mathbf{x}}(1), \tau_0, \tau_f) = 0, \tag{6.5}$$

$$\int_0^1 [\widehat{\mathbf{h}}(\alpha_k, \beta_k, \phi_k^{\mathbf{x}}, \phi_k^{\mathbf{u}}) + \sum_{k=1}^{N_{\mathbf{s}}} \varsigma_k \phi_k^{\mathbf{s}} \circ \sum_{l=1}^{N_{\mathbf{s}}} \varsigma_l \phi_l^{\mathbf{s}}] \phi_p^{\mathbf{s}} = 0, \tag{6.6}$$

where $j = 1, .., N_x$ and $p = 1, .., N_s$. This approximation scheme as defined by Eqns. (6.2)-(6.6) represents the generalized moment method for optimal control.

Next, we derive the equivalence conditions and costate estimation results for this method.

B. Equivalence Conditions

In the context of GMM_{oc}, we make the following observations form Chapter III Section E:

1. Conditions in Eqns. (3.54)-(3.57) are trivially satisfied because,

$$\frac{\partial \dot{\phi}_j^{\mathbf{x}}}{\partial \mathbf{u}} = \frac{\partial \dot{\phi}_j^{\mathbf{x}}}{\partial \tau_0} = \frac{\partial \dot{\phi}_j^{\mathbf{x}}}{\partial \tau_f} = 0. \tag{6.7}$$

2. Since not all basis functions $\phi_k^{\mathbf{x}}|_{k=1}^{N_{\mathbf{x}}}$ are zero at the boundaries, the conditions in Eqns. (3.52) and (3.53) require the following to be satisfied,

$$\sum_{j=1}^{N_{\mathbf{x}}} W_j(0) \gamma_j = \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j \dot{\phi}_j^{\mathbf{x}}(0) = -[\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}_{\mathbf{x}(0)}^T \eta + \mu_0 K_0], \tag{6.8}$$

$$\sum_{j=1}^{N_{\mathbf{x}}} W_j(1) \gamma_j = \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j \dot{\phi}_j^{\mathbf{x}}(1) = [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)}^T \eta + \mu_1 K_1]. \tag{6.9}$$

3. Since ν_0 and ν_1 are unknowns in the LSM_{oc} formulation, conditions in Eqn.

(3.58) supply the boundary conditions for the "auxiliary costates" $\rho(t)$ defined by the differential equation,

$$\dot{\rho}(t) = \lambda(t). \tag{6.10}$$

Let $\sum_{j=1}^{N_{\mathbf{x}}} \widetilde{\gamma}_j \phi_j^{\mathbf{x}}(t) = \widehat{\rho}(t)$. Then the boundary conditions for Eqn.(6.10) are given by the extra conditions found in the KKT system, i.e.,

$$K_0\widehat{\rho}(0) = 0, \qquad K_1\widehat{\rho}(1) = 0. \tag{6.11}$$

Thus, the conditions in Eqn. (3.58) do not introduce any inconsistency in the KKT system of $\mathrm{GMM_{oc}}$ formulation.

C. Costate Estimates

The equivalence conditions defined in Eqns. (6.8) and (6.9) provide the mapping between KKT multipliers $\{\gamma_k\}$, and the costate approximation $\widehat{\lambda}(t)$. Similarly, a mapping exists between $\{\xi(t), \mu_0, \mu_1, \eta\}$ and the corresponding KKT multipliers. We summarize the costate estimation results for the $\mathrm{GMM_{oc}}$ via the following theorem:

Theorem C.1 (Costate Mapping Theorem for the $\mathbf{GMM_{oc}}$) *Assume that an optimal control problem is solved using the GMM_{oc} and the equivalence conditions hold. Then, the estimates of the costates $\widehat{\lambda}(t)$, and the Lagrange multiplier functions $\widehat{\xi}(t)$ can be obtained from the KKT multipliers (γ_k, ζ_k) of the associated NLP as,*

$$\begin{aligned} \widehat{\lambda}(t) &= \sum_{k=1}^{N_{\mathbf{x}}} \gamma_k \dot{\phi}_k^{\mathbf{x}}(t) \\ \widehat{\xi}(t) &= \sum_{k=1}^{N_{\mathbf{s}}} \zeta_k \phi_k^{\mathbf{s}}(t). \end{aligned} \tag{6.12}$$

Proof The solution to Problem $\mathcal{M}_{\lambda\phi}$ exists by assumption. Since equivalence conditions hold, the results in Chapter III Section F are valid. ∎

D. GMM_{oc} Using B-Spline Approximation

Numerical implementation of GMM_{oc} using B-Spline basis functions is on the same lines as presented for LSM_{oc} in the previous chapter. It can be shown that a valid numerical quadrature can be used for GMM_{oc} for which the results for equivalence conditions and costate estimation hold. For brevity, the details are skipped and the reader is referred to Chapter V Section E.

E. GMM_{oc} Using Global Interpolating Polynomials: s-GMM_{oc}

In this section we formulate the spectral version of GMM_{oc} on the same lines as s-LSM_{oc} was developed in the previous chapter. The trial functions are chosen to be the global interpolating Lagrange polynomials. Also, the quadrature nodes are taken to be the same as the interpolation nodes.

Let us take the interpolation/quadrature nodes to be N number of LGL points $\{t_k\}_{k=1}^N$ suitably mapped on the interval $[0,1]$ with corresponding quadrature weights $\{w_k\}_{k=1}^N$. The corresponding Lagrange interpolation basis is $\{\mathcal{L}_i(t)\}_{i=1}^N$. All trajectory variables are approximated using the Lagrange basis. So that,

$$\phi_i^{\mathbf{x}}(t) = \phi_i^{\mathbf{u}}(t) = \phi_i^{\mathbf{s}}(t) = \mathcal{L}_i(t); \quad t \in [0,1]. \tag{6.13}$$

Using the Kronecker delta property of Lagrange polynomials $L_j(t_k) = \delta_{jk}$, we get for

$i = 1, 2, .., N,$

$$\mathbf{x}_i = \widehat{\mathbf{x}}(t_i) = \sum_{k=1}^{N} \alpha_k L_k(t_i) = \alpha_i, \quad \mathbf{u}_i = \widehat{\mathbf{u}}(t_i) = \sum_{k=1}^{N} \beta_k L_k(t_i) = \beta_i, \tag{6.14}$$

$$\mathbf{s}_i = \widehat{\mathbf{s}}(t_i) = \sum_{k=1}^{N} \varsigma_k L_k(t_i) = \varsigma_i. \tag{6.15}$$

Further, a differentiation matrix D is defined as,

$$D_{ij} = \dot{\mathcal{L}}_j(t_i), \tag{6.16}$$

so that,

$$\dot{\widehat{\mathbf{x}}}(t_i) = \dot{\mathbf{x}}_i = \sum_{j=1}^{N} D_{ij} \mathbf{x}_j; \quad i = 1, .., N. \tag{6.17}$$

Using Eqns. (6.13)-(6.17) with the NLP formulation given in section A of this chapter, the s-GMM$_{\text{oc}}$ transcription of an optimal control problem is defined as following: Determine $\{\mathbf{x}_k \in \mathbb{R}^n\}_{k=1}^N, \{\mathbf{u}_k \in \mathbb{R}^m\}_{k=1}^N, \{\mathbf{s}_k \in \mathbb{R}^q\}_{k=1}^N$, $\nu_0 \in \mathbb{R}^a$, $\nu_1 \in \mathbb{R}^{n-a}$ and time instances τ_0 and τ_f, that minimize the cost,

$$\widehat{J} = \widehat{\Psi}(\mathbf{x}_1, \mathbf{x}_N, \tau_0, \tau_f), \tag{6.18}$$

subject to the constraints,

$$\sum_{k=1}^{N} w_k R_k D_{k1} + \frac{1}{2} K_0^T \nu_0 = 0, \tag{6.19}$$

$$\sum_{k=1}^{N} w_k R_k D_{kN} + \frac{1}{2} K_1^T \nu_1 = 0, \tag{6.20}$$

$$\sum_{k=1}^{N} w_k R_k D_{ki} = 0; \quad i = 2, .., N-1, \tag{6.21}$$

$$K_0\mathbf{x}_1 - \mathbf{x}_0 = 0, \quad K_1\mathbf{x}_N - \mathbf{x}_f = 0, \tag{6.22}$$

$$\widehat{\psi}(\mathbf{x}_1, \mathbf{x}_N, \tau_0, \tau_f) = 0, \tag{6.23}$$

$$\widehat{\mathbf{h}}(\mathbf{x}_j, \mathbf{u}_j) + \mathbf{s}_j^2 = 0; \quad j = 1, .., N, \tag{6.24}$$

where,

$$R_k = \widehat{\mathbf{f}}(\mathbf{x}_k, \mathbf{u}_k) - \sum_{l=1}^{N} D_{kl}\mathbf{x}_l; \quad k = 1, .., N. \tag{6.25}$$

Eqns. (6.18)-(6.24) represent the s-GMM$_{\text{oc}}$ discretization of the original optimal control problem. It can be easily shown that for the given quadrature scheme, the integration by parts operation is valid. Therefore, the equivalence conditions for s-GMM$_{\text{oc}}$ are,

$$\tau_f \sum_{k=1}^{N} D_{1k}\gamma_k = -[\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}_{\mathbf{x}(0)}^T\eta + \mu_0 K_0], \tag{6.26}$$

$$\tau_f \sum_{k=1}^{N} D_{Nk}\gamma_k = [\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)}^T\eta + \mu_1 K_1], \tag{6.27}$$

where $\gamma_1, \gamma_N, \{\gamma_j\}_{j=2}^{N-1}$ are the KKT multipliers associated with the constraints in Eqns. (5.42), (5.43) and (5.44) respectively. Under the equivalence conditions, the costates can be estimated as,

$$\widehat{\lambda}(t_j) = \sum_{k=1}^{N} D_{jk}\gamma_k. \tag{6.28}$$

F. Numerical Convergence Analysis

Convergence properties of GMM$_{\text{oc}}$ are numerically demonstrated in this section. Four example problems are taken from Chapter IV and results are generated using the same methodology as presented in Chapter IV Section F.

1. Convergence Results for LSM_{oc}

- Example 1: Convergence results for Example 1 are shown in Figure (27). It is seen that higher polynomial order approximation scheme SPL-3r performs better than the low order scheme SPL-34. Further, LEG scheme does not converge for many cases.

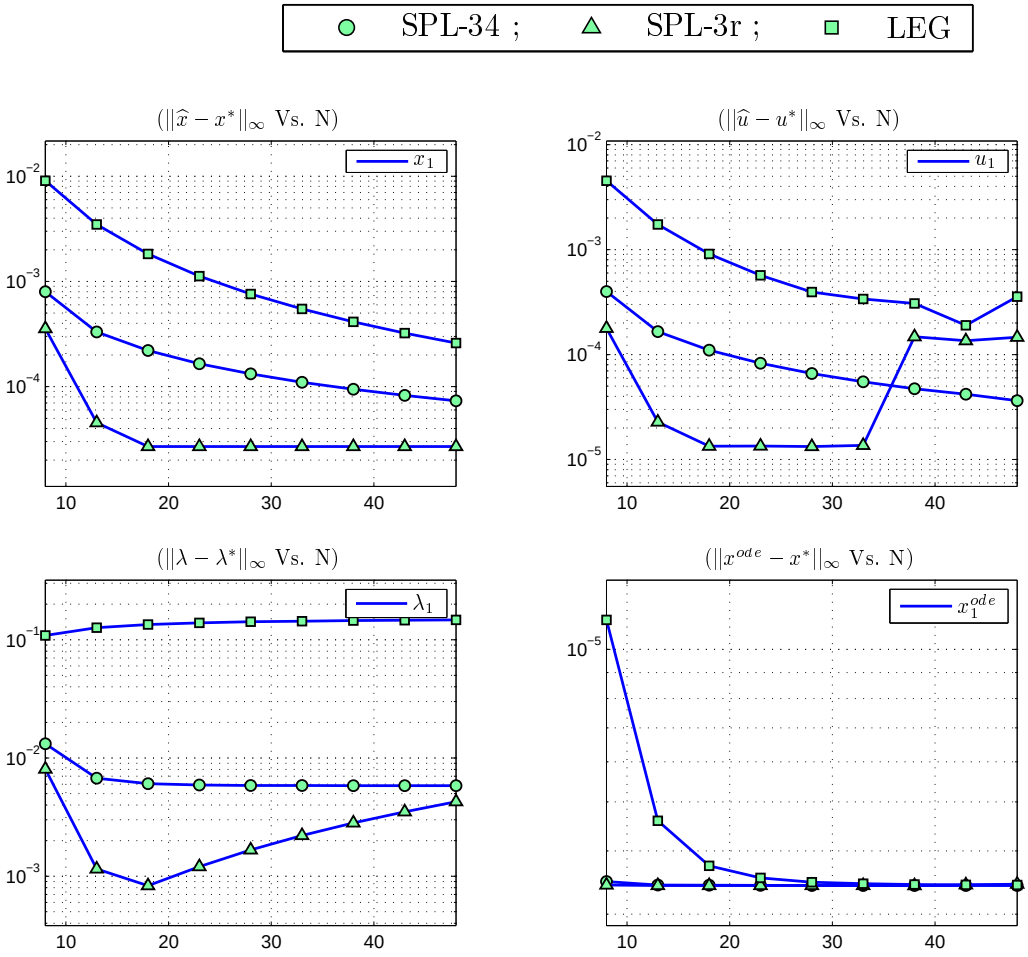


Fig. 27. Convergence of GMM_{oc} solution for Example 1.

- Example 2: Convergence results for Example 2 are shown in Figure (28). All the test cases converge and it is very well seen that the SPL-3r approximation works best for this example.

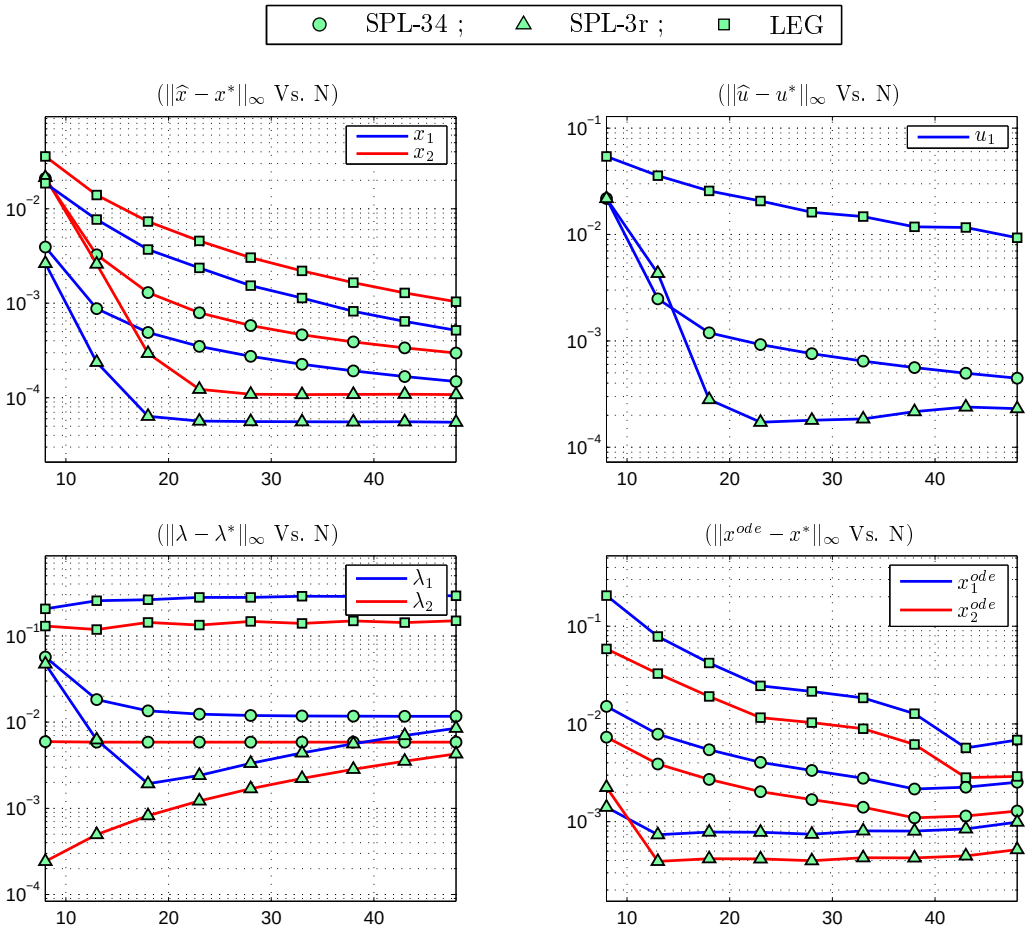


Fig. 28. Convergence of GMM_{oc} solution for Example 2.

- Example 3: Example 3 is solved using SPL-34 and SPL-3r approximations and the convergence results are depicted in Figure (29). This example shows that the local approximation scheme SPL-34 performs better than the global SPL-3r scheme. This is due to the presence of discontinuities and corners in the solution of this example.

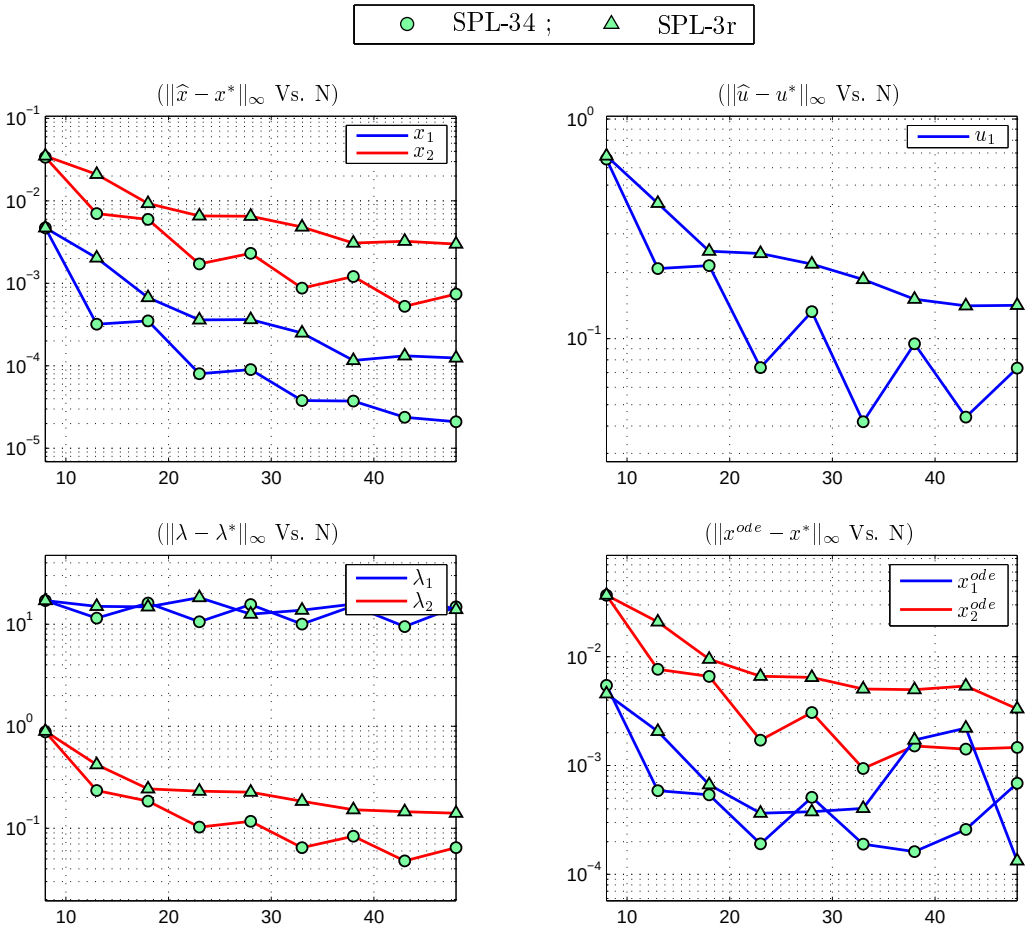


Fig. 29. Convergence of GMM_{oc} solution for Example 3.

- Example 4: Example 4 has a discontinuous control solution. Convergence results in Figure (30) show that SPL-34 gives better solutions than SPL-3r. Also, GMM_{oc} gives quite accurate costate estimates for this example.

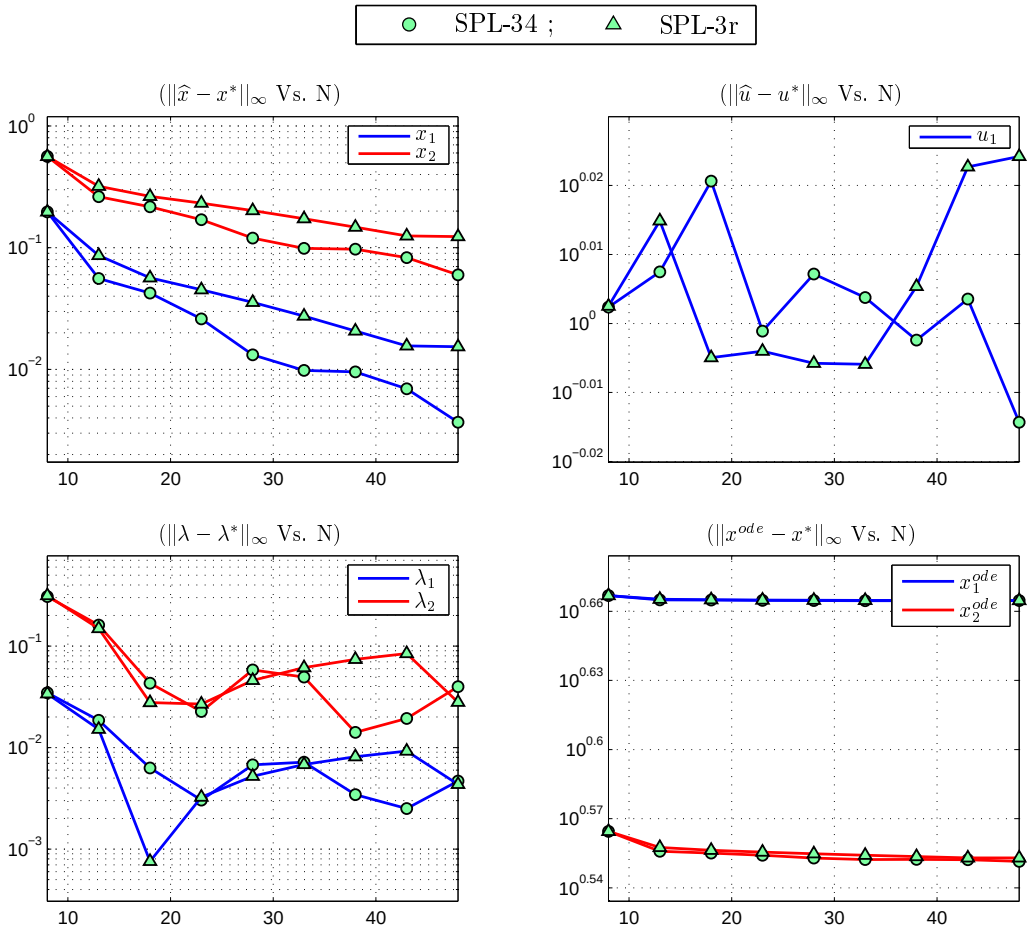


Fig. 30. Convergence of GMM_{oc} solution for Example 4.

2. Convergence Results for s-GMM$_{oc}$

- Example 1: Convergence results for Example 1 are shown in Figure (31). Very high convergence rates are achieved as N goes from 8 to 28. It will be shown in Chapter VIII that these convergence rates are comparable to the existing pseudospectral methods.

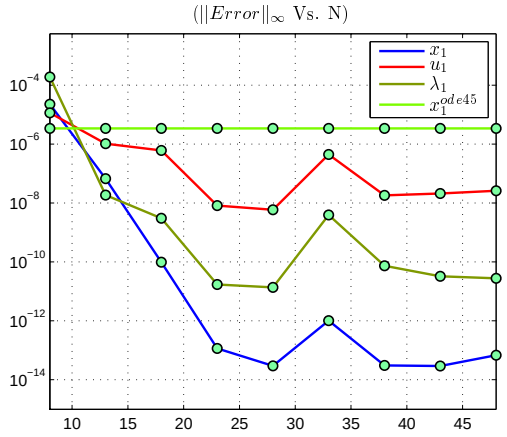


Fig. 31. Convergence of s-GMM$_{oc}$ solution for Example 1.

- Example 2: Convergence results for Example 2 are shown in Figure (32).

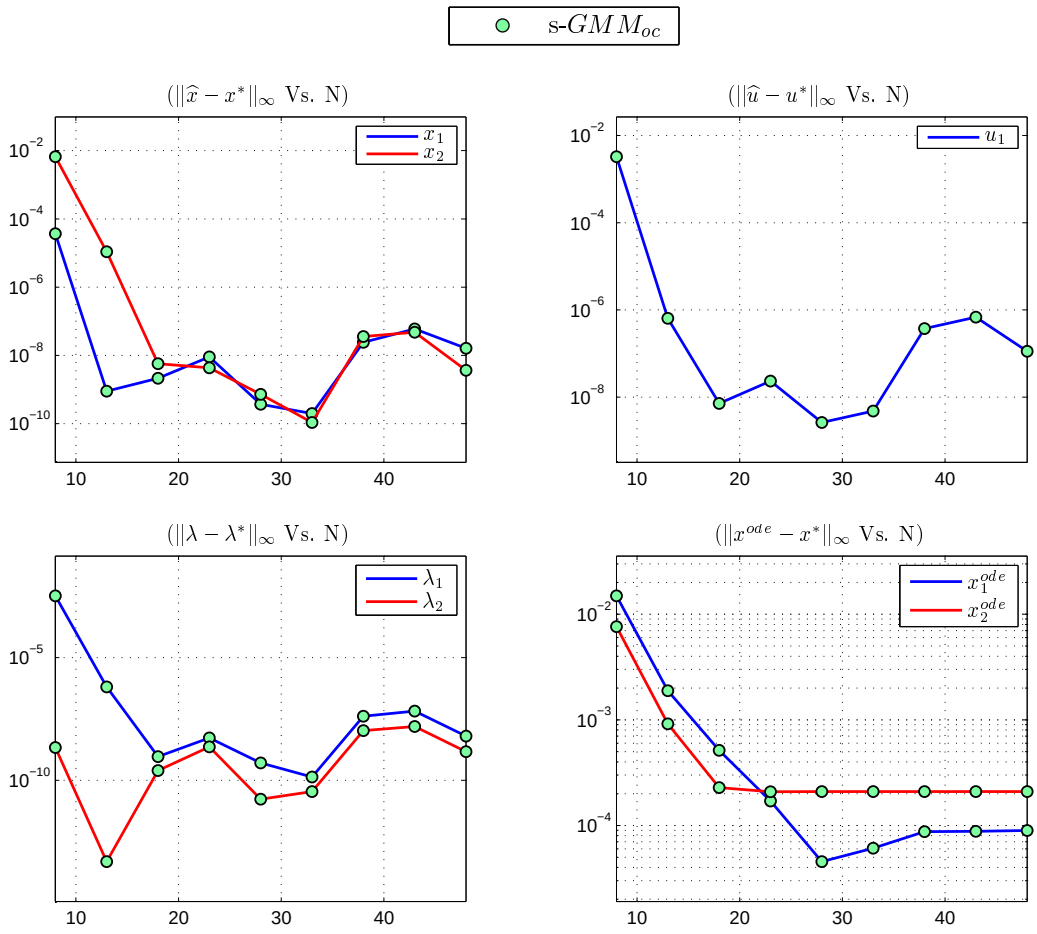


Fig. 32. Convergence of s-GMM$_{\text{oc}}$ solution for Example 2.

- Example 3: Example 3 convergence rates are slower due to the irregularities in its solution. Still, convergence is achieved to a reasonable accuracy using s-GMM$_{\text{oc}}$. Results are shown in Figure (33).

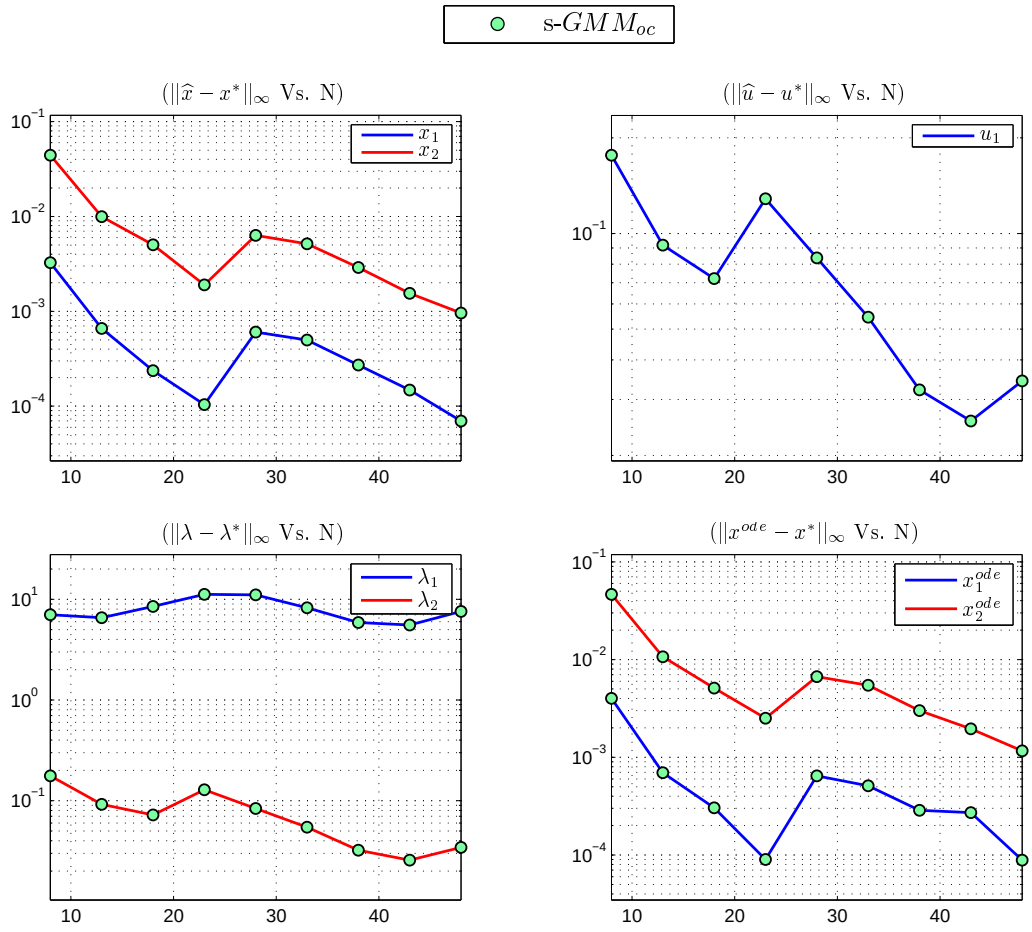


Fig. 33. Convergence of s-GMM$_{\text{oc}}$ solution for Example 3.

- Example 4: Example 4 also has a slower but smooth convergence. Results are depicted in Figure (34).

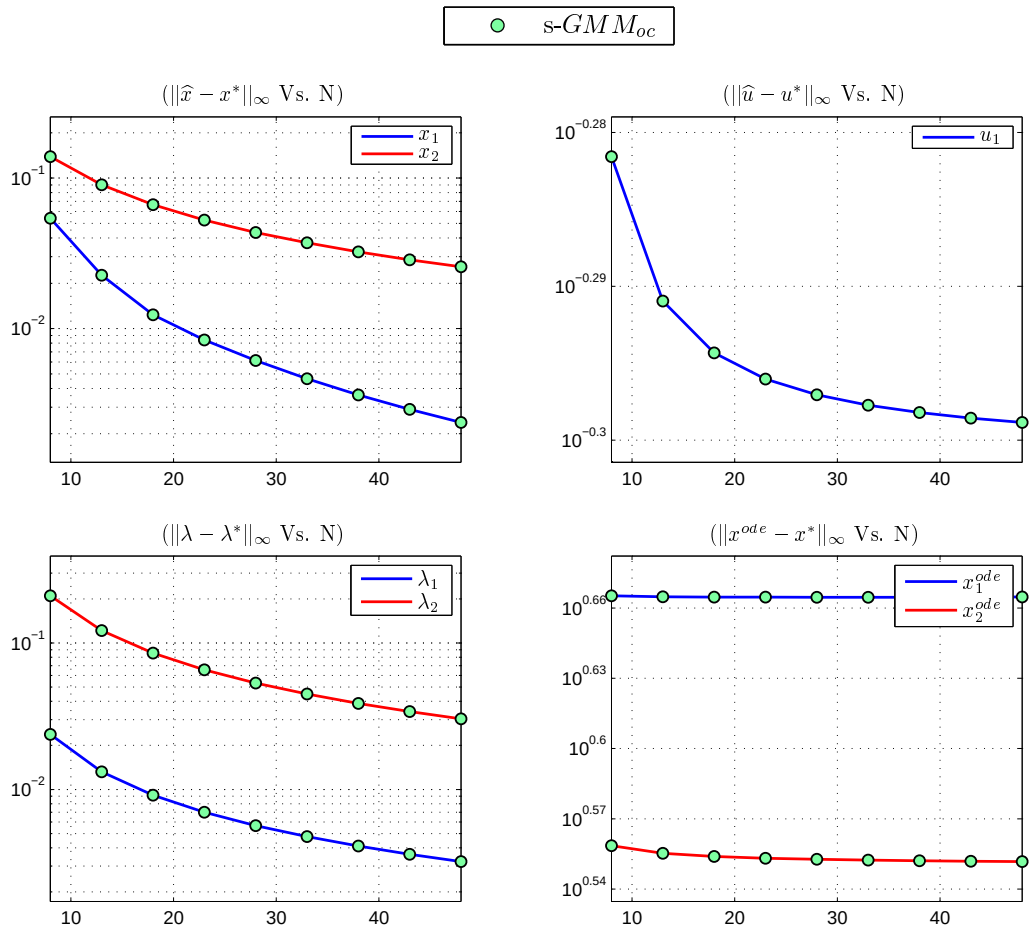


Fig. 34. Convergence of s-GMM_{oc} solution for Example 4.

G. Conclusions

This chapter presented the development of GMM_{oc} for direct transcription of optimal control problems. Equivalence conditions were derived for indirect-direct mapping and costate estimates were obtained from the Lagrange multipliers of the nonlinear programming problem. GMM_{oc} was implemented with both local and global ba-

sis functions. A spectral version of $\mathrm{GMM_{oc}}$ was derived using global interpolating polynomials as approximating functions. Convergence properties of $\mathrm{GMM_{oc}}$ were demonstrated through numerical examples.

CHAPTER VII

PSEUDOSPECTRAL METHODS IN THE FRAMEWORK OF WEIGHTED RESIDUAL APPROXIMATION

In recent years, pseudospectral methods have gained wide popularity for direct transcription of optimal control problems. In a pseudospectral method, state and control trajectories are approximated as global interpolating polynomials at a set of interpolation points. The interpolation points are also the collocation sites where the state dynamics is imposed. Further, the interpolation/collocation points are chosen as the roots of an orthogonal polynomial from Jacobi family such as Legendre or Chebyshev polynomials. That is why pseudospectral methods are also known as orthogonal collocation methods. Pseudospecral methods exhibit very fast convergence rates for smooth problems, a property known as "spectral accuracy".

A number of pseudospectral methods have been proposed in the literature. In this chapter, we show that some of the most popular pseudospectral methods, Legendre, Radau and Gauss pseudospectral method, can be derived from the proposed weighted residual formulation by proper selection of test/trial functions and an associated quadrature scheme. We shall see that orthogonal collocation actually represents a Galerkin/Tau type approach where test functions are selected from the set of trial functions. The already established costate approximation results for these pseudospectral methods will also be derived under the unifying weighted residual framework.

A. Legendre Pseudospectral Method (LPS)

For WRM, take the computational domain as $t \in [-1, 1]$ containing a set of N number of LGL node points $\{t_k\}_{k=1}^{N}$ with corresponding weights $\{w_k\}_{k=1}^{N}$. Let L_k be the k^{th}

Lagrange interpolation basis corresponding to node t_k. Assume that $\{L_k\}_{k=1}^N$ define the trial functions for all the trajectory variables in the optimal control problem, i.e.,

$$\phi_k^{\mathbf{x}}(t) = \phi_k^{\mathbf{u}}(t) = \phi_k^{\mathbf{s}}(t) = L_k(t) = \prod_{j=1, j\neq k}^{N} \frac{(t-t_j)}{(t_k-t_j)}, \tag{7.1}$$

and the test functions are taken as,

$$W_k = V_k = \frac{L_k}{w_k}. \tag{7.2}$$

with,

$$\nu_0 = \nu_1 = 0. \tag{7.3}$$

Use the property of Lagrange interpolating basis $L_j(t_k) = \delta_{jk}$ to obtain for $i = 1, 2, .., N$,

$$\mathbf{x}_i = \widehat{\mathbf{x}}(t_i) = \sum_{k=1}^{N} \alpha_k L_k(t_i) = \alpha_i, \quad \mathbf{u}_i = \widehat{\mathbf{u}}(t_i) = \sum_{k=1}^{N} \beta_k L_k(t_i) = \beta_i, \tag{7.4}$$

$$\mathbf{s}_i = \widehat{\mathbf{s}}(t_i) = \sum_{k=1}^{N} \varsigma_k L_k(t_i) = \varsigma_i. \tag{7.5}$$

Then, from Eqns. (3.18) and (3.21), the weighted residual approximation of state dynamics and path constraints has the form,

$$\int_{-1}^{1} \left[\frac{(\tau_f - \tau_0)}{2} \widehat{\mathbf{f}}(\mathbf{x}_k, \mathbf{u}_k, L_k(t)) - \sum_{k=1}^{N} \mathbf{x}_k \dot{L}_k(t) \right] \frac{L_j(t)}{w_j} dt = 0 \tag{7.6}$$

$$\int_{-1}^{1} [\widehat{\mathbf{h}}(\mathbf{x}_k, \mathbf{u}_k, L_k(t)) + \sum_{k=1}^{N} \mathbf{s}_k L_k \circ \sum_{l=1}^{N} \mathbf{s}_l L_l] \frac{L_j(t)}{w_j} dt = 0, \tag{7.7}$$

where $j = 1, .., N$. Using Gauss quadrature for evaluating the integral expressions in Eqns. (7.6) and (7.7) results in the conditions,

$$\frac{(\tau_f - \tau_0)}{2}\widehat{\mathbf{f}}(\mathbf{x}_j, \mathbf{u}_j) - \sum_{k=1}^{N} \mathbf{x}_k \dot{L}_k(t_j) = 0 \tag{7.8}$$

$$\widehat{\mathbf{h}}(\mathbf{x}_j, \mathbf{u}_j) + \mathbf{s}_j \cdot \mathbf{s}_j = 0. \tag{7.9}$$

Based on the formulation given in Chapter III Section A and using the above substitutions, the nonlinear programming problem is to determine $\{\mathbf{x}_k \in \mathbb{R}^n\}_{k=1}^N, \{\mathbf{u}_k \in \mathbb{R}^m\}_{k=1}^N, \{\mathbf{s}_k \in \mathbb{R}^q\}_{k=1}^N, \nu_0 \in \mathbb{R}^a, \nu_1 \in \mathbb{R}^{n-a}$ and time instances τ_0 and τ_f, that minimize the cost,

$$\widehat{J} = \widehat{\Psi}(\mathbf{x}_1, \mathbf{x}_N, \tau_0, \tau_f), \tag{7.10}$$

subject to the constraints,

$$\frac{(\tau_f - \tau_0)}{2}\widehat{\mathbf{f}}(\mathbf{x}_j, \mathbf{u}_j) - \sum_{k=1}^{N} \mathbf{x}_k \dot{L}_k(t_j) = 0 \tag{7.11}$$

$$K_0\mathbf{x}_1 - \mathbf{x}_0 = 0, \quad K_1\mathbf{x}_N - \mathbf{x}_f = 0 \tag{7.12}$$

$$\widehat{\psi}(\mathbf{x}_1, \mathbf{x}_N, \tau_0, \tau_f) = 0, \tag{7.13}$$

$$\widehat{\mathbf{h}}(\mathbf{x}_j, \mathbf{u}_j) + \mathbf{s}_j \cdot \mathbf{s}_j = 0. \tag{7.14}$$

where $j = 1, .., N$. The NLP defined by Eqns. (7.10)-(7.14) represents the Legendre pseudospectral discretization of $\mathcal{M}$. Thus, under the conditions defined in Eqns. (7.1),(7.2) and (7.3), the WRM is equivalent to the Legendre pseudospectral method [25, 60].

Next step is to derive costate mapping results for Legendre pseudospectral method based on Section D of Chapter III. Conditions given in Eqns. (3.54)-(3.57) are trivially

satisfied as,

$$\frac{\partial W_j}{\partial \mathbf{x}} = \frac{\partial W_j}{\partial \mathbf{u}} = \frac{\partial W_j}{\partial \tau_0} = \frac{\partial W_j}{\partial \tau_f} = 0. \tag{7.15}$$

The extra conditions in Eqn. (3.58) are not applicable as $\nu_0 = \nu_1 = 0$. Eqns. (3.52) and (3.53) are not satisfied explicitly for the value of $i = 1$ and N respectively. Therefore, for the complete mapping to exist between the KKT system and the discretized first-order optimality conditions for Legendre pseudospectral method, the following conditions, also known as closure conditions, are required to be satisfied:

$$\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}^T_{\mathbf{x}(0)}\eta + \mu_0 K_0 + \gamma_1 = 0, \tag{7.16}$$

$$\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}^T_{\mathbf{x}(1)}\eta + \mu_1 K_1 - \gamma_N = 0. \tag{7.17}$$

Assuming that the closure conditions are satisfied, the costates can be approximated using Eqn. (3.59) as,

$$\widehat{\lambda}(t) = \sum_{j=1}^{N} \gamma_j \frac{L_j(t)}{w_j}, \quad \text{and} \quad \widehat{\lambda}_k = \widehat{\lambda}(t_k) = \sum_{j=1}^{N} \gamma_j \frac{\delta_{jk}}{w_j} = \frac{\gamma_k}{w_k}. \tag{7.18}$$

B. Radau Pseudospectral Method (RPS)

For WRM, take the computational domain as $t \in [-1, 1]$ containing a set of N number of node points $\{t_k\}_{k=1}^N$ such that,

$$t_1 = -1, w_1 = 0, \qquad \{t_k\}_{k=2}^N : (N-2) \text{ Number of LGR points} \in (-1, 1], \tag{7.19}$$

with corresponding weights $\{w_k\}_{k=2}^N$. Let L_k be the k^{th} Lagrange interpolation basis corresponding to node t_k. Assume that the trial functions for the trajectory variables

in the optimal control problem are,

$$\phi_k^{\mathbf{x}}(t) = L_k(t) : (k = 1, .., N) \tag{7.20}$$

$$\phi_k^{\mathbf{s}}(t) = \phi_k^{\mathbf{u}}(t) = L_k(t) : (k = 2, .., N), \quad \phi_1^{\mathbf{s}}(t) = \phi_1^{\mathbf{u}}(t) = 0, \tag{7.21}$$

and the test functions are taken as,

$$W_1 = 0, \quad W_k = \frac{L_k}{w_k} : (k = 2, .., N), \tag{7.22}$$

with,

$$\nu_0 = \nu_1 = 0. \tag{7.23}$$

Using the property of Lagrange interpolating basis $L_j(t_k) = \delta_{jk}$, for $i = 1, 2, .., N$,

$$\mathbf{x}_i = \widehat{\mathbf{x}}(t_i) = \alpha_i, \quad \mathbf{u}_i = \widehat{\mathbf{u}}(t_i) = \beta_i, \quad \mathbf{s}_i = \widehat{\mathbf{s}}(t_i) = \varsigma_i. \tag{7.24}$$

Then, from Eqns. (3.18) and (3.21), the weighted residual approximation of state dynamics and path constraints has the form,

$$\int_{-1}^{1} \left[\frac{(\tau_f - \tau_0)}{2} \widehat{\mathbf{f}}(\mathbf{x}_k, \mathbf{u}_k, L_k(t)) - \sum_{k=1}^{N} \mathbf{x}_k \dot{L}_k(t) \right] \frac{L_j(t)}{w_j} dt = 0; j = 2, .., N, \tag{7.25}$$

$$\int_{-1}^{1} [\widehat{\mathbf{h}}(\mathbf{x}_k, \mathbf{u}_k, L_k(t)) + \sum_{k=1}^{N} \mathbf{s}_k L_k \circ \sum_{l=1}^{N} \mathbf{s}_l L_l] \frac{L_j(t)}{w_j} dt = 0; j = 2, .., N, \tag{7.26}$$

Using Gauss quadrature for evaluating the integral expressions in Eqns. (7.6) and (7.7) results in the conditions,

$$\frac{(\tau_f - \tau_0)}{2} \widehat{\mathbf{f}}(\mathbf{x}_j, \mathbf{u}_j) - \sum_{k=1}^{N} \mathbf{x}_k \dot{L}_k(t_j) = 0; j = 2, .., N \tag{7.27}$$

$$\widehat{\mathbf{h}}(\mathbf{x}_j, \mathbf{u}_j) + \mathbf{s}_j \cdot \mathbf{s}_j = 0; j = 2, .., N. \tag{7.28}$$

Notice that the trajectory constraints are not collocated at the initial point $t_1 = -1$. Based on the formulation given in Chapter III Section A and using the above substitutions, the nonlinear programming problem is to determine $\{\mathbf{x}_k \in \mathbb{R}^n\}_{k=1}^N, \{\mathbf{u}_k \in \mathbb{R}^m\}_{k=1}^N, \{\mathbf{s}_k \in \mathbb{R}^q\}_{k=1}^N, \nu_0 \in \mathbb{R}^a, \nu_1 \in \mathbb{R}^{n-a}$ and time instances τ_0 and τ_f, that minimize the cost,

$$\widehat{J} = \widehat{\Psi}(\mathbf{x}_1 \mathbf{x}_N, \tau_0, \tau_f), \tag{7.29}$$

subject to the constraints,

$$\frac{(\tau_f - \tau_0)}{2}\widehat{\mathbf{f}}(\mathbf{x}_j, \mathbf{u}_j) - \sum_{k=1}^{N} \mathbf{x}_k \dot{L}_k(t_j) = 0; \quad j = 1, .., N \tag{7.30}$$

$$\widehat{\psi}(\mathbf{x}_1, \mathbf{x}_N, \tau_0, \tau_f) = 0, \tag{7.31}$$

$$\widehat{\mathbf{h}}(\mathbf{x}_j, \mathbf{u}_j) + \mathbf{s}_j \cdot \mathbf{s}_j = 0; \quad j = 1, .., N. \tag{7.32}$$

The NLP defined by Eqns. (7.29)-(7.32) represents the Radau pseudospectral discretization of $\mathcal{M}$. Thus, under the conditions defined in Eqns. (7.20),(7.21),(7.22) and (7.23), the WRM is equivalent to the Radau pseudospectral method [61].

Next step is to derive costate mapping results for Radau pseudospectral method based on Section D of Chapter III. Conditions given in Eqns. (3.54)-(3.57) are trivially satisfied as,

$$\frac{\partial W_j}{\partial \mathbf{x}} = \frac{\partial W_j}{\partial \mathbf{u}} = \frac{\partial W_j}{\partial \tau_0} = \frac{\partial W_j}{\partial \tau_f} = 0. \tag{7.33}$$

The extra conditions in Eqn. (3.58) are not applicable as $\nu_0 = \nu_1 = 0$. Eqns. (3.52)

and (3.53) are explicitly satisfied except for $i = 1$, i.e.

$$[\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}^T_{\mathbf{x}(0)}\eta + \mu_0 K_0 + \sum_{j=1}^{N_{\mathbf{x}}} W_j(0)\gamma_j]\phi^{\mathbf{x}}_i(0) = 0; \quad i = 2, .., N, \quad \text{(explicitly)}, \tag{7.34}$$

$$[\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}^T_{\mathbf{x}(1)}\eta + \mu_1 K_1 - \sum_{j=1}^{N_{\mathbf{x}}} W_j(1)\gamma_j]\phi^{\mathbf{x}}_i(1) = 0; \quad i = 1, .., N-1, \quad \text{(explicitly)}. \tag{7.35}$$

For the first point, $i = 1$ Eqn. (3.52) is given by,

$$\begin{aligned}
&[\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}^T_{\mathbf{x}(0)}\eta + \mu_0 K_0 + \sum_{j=1}^{N_{\mathbf{x}}} W_j(0)\gamma_j] \\
&= -\int_0^1 \left[\frac{(\tau_f - \tau_0)}{2}\widehat{\mathbf{f}}^T_{\mathbf{x}} \sum_{j=1}^{N} W_j\gamma_j + \sum_{j=1}^{N} \dot{W}_j\gamma_j + \sum_{p=1}^{N} \widehat{\mathbf{h}}^T_{\mathbf{x}} V_p \zeta_p\right] L_1(t)dt, \\
&= -\sum_{i=1}^{N}\left[\frac{(\tau_f - \tau_0)}{2}\widehat{\mathbf{f}}^T_{\mathbf{x}} \sum_{j=1}^{N} W_j(t_i)\gamma_j + \sum_{j=1}^{N} \dot{W}_j(t_i)\gamma_j + \sum_{p=1}^{N} \widehat{\mathbf{h}}^T_{\mathbf{x}} V_p(t_i) \zeta_p\right] w_i L_1(t_i) \\
&= -\left[\frac{(\tau_f - \tau_0)}{2}\widehat{\mathbf{f}}^T_{\mathbf{x}} \sum_{j=1}^{N} W_j(0)\gamma_j + \sum_{j=1}^{N} \dot{W}_j(0)\gamma_j + \sum_{p=1}^{N} \widehat{\mathbf{h}}^T_{\mathbf{x}} V_p(0) \zeta_p\right] w_1 \\
&= 0; \quad \text{(explicitly)}.
\end{aligned} \tag{7.36}$$

However, for the last point $i = N$, Eqn. (3.53) is not explicitly satisfied. Therefore, for the complete mapping to exist between the KKT system and the discretized first-order optimality conditions for Radau pseudospectral method, the following condition is required to be satisfied:

$$[\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}^T_{\mathbf{x}(1)}\eta + \mu_1 K_1 - \sum_{j=1}^{N_{\mathbf{x}}} W_j(1)\gamma_j] = 0. \tag{7.37}$$

Assuming that Eqn. (7.37) is satisfied, the costates can be approximated using Eqn.

(3.59) as,

$$\widehat{\lambda}(t) = \sum_{j=1}^{N} \gamma_j W_j = \sum_{j=2}^{N} \gamma_j \frac{L_j(t)}{w_j}, \tag{7.38}$$

$$\widehat{\lambda}_k = \widehat{\lambda}(t_k) = \sum_{j=2}^{N} \gamma_j \frac{\delta_{jk}}{w_j} = \frac{\gamma_k}{w_k}; \quad k = 2, .., N. \tag{7.39}$$

λ_1 is obtained from the boundary condition,

$$\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}^T_{\mathbf{x}(0)}\eta + \mu_0 K_0 + \gamma_1 = 0. \tag{7.40}$$

C. Gauss Pseudospectral Method (GPS)

For WRM, take the computational domain as $t \in [-1, 1]$ containing a set of N number of node points $\{t_k\}_{k=1}^{N}$ such that,

$$t_1 = -1,\ t_N = 1, \qquad \{t_k\}_{k=2}^{N-1} : (N-2) \text{ Number of LG points} \in (-1, 1), \tag{7.41}$$

with corresponding weights $\{w_k\}_{k=2}^{N-1}$. Let L_k be the k^{th} Lagrange interpolation basis corresponding to node t_k. Assume that the trial functions for the trajectory variables in the optimal control problem are,

$$\phi^{\mathbf{x}}_k(t) = L_k(t) : (k = 1, .., N-1), \qquad \phi^{\mathbf{x}}_N(t) = 0, \tag{7.42}$$

$$\phi^{\mathbf{s}}_k(t) = \phi^{\mathbf{u}}_k(t) = L_k(t) : (k = 2, .., N-1), \quad \phi^{\mathbf{s}}_1(t) = \phi^{\mathbf{u}}_1(t) = \phi^{\mathbf{s}}_N(t) = \phi^{\mathbf{u}}_N(t) = 0, \tag{7.43}$$

and the test functions are taken as,

$$W_1 = 0, \quad W_k = \frac{L_k}{w_k} : (k = 2, .., N-1), \quad W_N = 1, \tag{7.44}$$

with,

$$\nu_0 = \nu_1 = 0. \tag{7.45}$$

Use the property of Lagrange interpolating basis $L_j(t_k) = \delta_{jk}$ to obtain for $i = 1, 2, .., N$,

$$\mathbf{x}_i = \widehat{\mathbf{x}}(t_i) = \alpha_i, \quad \mathbf{u}_i = \widehat{\mathbf{u}}(t_i) = \beta_i, \quad \mathbf{s}_i = \widehat{\mathbf{s}}(t_i) = \varsigma_i. \tag{7.46}$$

Then, from Eqns. (3.18) and (3.21), the weighted residual approximation of state dynamics and path constraints has the form,

$$\int_{-1}^{1} \left[\frac{(\tau_f - \tau_0)}{2} \widehat{\mathbf{f}}(\mathbf{x}_k, \mathbf{u}_k, L_k(t)) - \sum_{k=1}^{N} \mathbf{x}_k \dot{L}_k(t) \right] \frac{L_j(t)}{w_j} dt = 0; j = 2, .., N-1, \tag{7.47}$$

$$\int_{-1}^{1} \left[\frac{(\tau_f - \tau_0)}{2} \widehat{\mathbf{f}}(\mathbf{x}_k, \mathbf{u}_k, L_k(t)) - \sum_{k=1}^{N} \mathbf{x}_k \dot{L}_k(t) \right] dt = 0 \tag{7.48}$$

$$\int_{-1}^{1} [\widehat{\mathbf{h}}(\mathbf{x}_k, \mathbf{u}_k, L_k(t)) + \sum_{k=1}^{N} \mathbf{s}_k L_k \circ \sum_{l=1}^{N} \mathbf{s}_l L_l] \frac{L_j(t)}{w_j} dt = 0, \tag{7.49}$$

Using Gauss quadrature for evaluating the integral expressions in Eqns. (7.48) and (7.49) results in the conditions,

$$\frac{(\tau_f - \tau_0)}{2} \widehat{\mathbf{f}}(\mathbf{x}_j, \mathbf{u}_j) - \sum_{k=1}^{N} \mathbf{x}_k \dot{L}_k(t_j) = 0; j = 2, .., N-1 \tag{7.50}$$

$$\frac{(\tau_f - \tau_0)}{2} \sum_{i=2}^{N-1} w_i \widehat{\mathbf{f}}(\mathbf{x}_i, \mathbf{u}_i) - \mathbf{x}_N + \mathbf{x}_1 = 0 \tag{7.51}$$

$$\widehat{\mathbf{h}}(\mathbf{x}_j, \mathbf{u}_j) + \mathbf{s}_j \cdot \mathbf{s}_j = 0. \tag{7.52}$$

Based on the formulation given in Chapter III Section A and using the above substitutions, the nonlinear programming problem is to determine $\{\mathbf{x}_k \in \mathbb{R}^n\}_{k=1}^N$, $\{\mathbf{u}_k \in \mathbb{R}^m\}_{k=1}^N$, $\{\mathbf{s}_k \in \mathbb{R}^q\}_{k=1}^N$, $\nu_0 \in \mathbb{R}^a$, $\nu_1 \in \mathbb{R}^{n-a}$ and time instances τ_0 and τ_f, that minimize

the cost,

$$\widehat{J} = \widehat{\Psi}(\mathbf{x}_1, \mathbf{x}_N, \tau_0, \tau_f), \tag{7.53}$$

subject to the constraints,

$$\frac{(\tau_f - \tau_0)}{2}\widehat{\mathbf{f}}(\mathbf{x}_j, \mathbf{u}_j) - \sum_{k=1}^{N} \mathbf{x}_k \dot{L}_k(t_j) = 0; \quad j = 2, .., N-1, \tag{7.54}$$

$$\frac{(\tau_f - \tau_0)}{2}\sum_{i=2}^{N-1} w_i \widehat{\mathbf{f}}(\mathbf{x}_i, \mathbf{u}_i) - \mathbf{x}_N + \mathbf{x}_1 = 0 \tag{7.55}$$

$$K_0\mathbf{x}_1 - \mathbf{x}_0 = 0, \quad K_1\mathbf{x}_N - \mathbf{x}_f = 0 \tag{7.56}$$

$$\widehat{\psi}(\mathbf{x}_1, \mathbf{x}_N, \tau_0, \tau_f) = 0, \tag{7.57}$$

$$\widehat{\mathbf{h}}(\mathbf{x}_j, \mathbf{u}_j) + \mathbf{s}_j \cdot \mathbf{s}_j = 0. \tag{7.58}$$

The NLP defined by Eqns. (7.53)-(7.58) represents the Gauss pseudospectral discretization of $\mathcal{M}$. Thus, under the conditions defined in Eqns. (7.42),(7.43),(7.44) and (7.45), the WRM is equivalent to the Gauss pseudospectral method [27].

Next step is to derive costate mapping results for Gauss pseudospectral method based on Section D of Chapter III. Conditions given in Eqns. (3.54)-(3.57) are trivially satisfied as,

$$\frac{\partial W_j}{\partial \mathbf{x}} = \frac{\partial W_j}{\partial \mathbf{u}} = \frac{\partial W_j}{\partial \tau_0} = \frac{\partial W_j}{\partial \tau_f} = 0. \tag{7.59}$$

The extra conditions in Eqn. (3.58) are not applicable as $\nu_0 = \nu_1 = 0$. Eqns. (3.52) and (3.53) are explicitly satisfied as following,

$$[\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}^T_{\mathbf{x}(0)}\eta + \mu_0 K_0 + \sum_{j=1}^{N_{\mathbf{x}}} W_j(0)\gamma_j]\phi_i^{\mathbf{x}}(0) = 0; \quad i = 2, .., N, \quad \text{(explicitly)}, \tag{7.60}$$

$$[\widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}^T_{\mathbf{x}(1)}\eta + \mu_1 K_1 - \sum_{j=1}^{N_{\mathbf{x}}} W_j(1)\gamma_j]\phi_i^{\mathbf{x}}(1) = 0; \quad i = 1, .., N, \quad \text{(explicitly)}. \tag{7.61}$$

for $i = 1$, Eqn. (3.52) becomes,

$$
\begin{aligned}
&[\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}^T_{\mathbf{x}(0)}\eta + \mu_0 K_0 + \sum_{j=1}^{N_{\mathbf{x}}} W_j(0)\gamma_j] \\
&= -\int_0^1 \left[\frac{(\tau_f - \tau_0)}{2}\widehat{\mathbf{f}}^T_{\mathbf{x}} \sum_{j=1}^{N} W_j\gamma_j + \sum_{j=1}^{N}\dot{W}_j\gamma_j + \sum_{p=1}^{N}\widehat{\mathbf{h}}^T_{\mathbf{x}}V_p\zeta_p\right] L_1(t)dt, \\
&= -\sum_{i=1}^{N}\left[\frac{(\tau_f - \tau_0)}{2}\widehat{\mathbf{f}}^T_{\mathbf{x}} \sum_{j=1}^{N} W_j(t_i)\gamma_j + \sum_{j=1}^{N}\dot{W}_j(t_i)\gamma_j + \sum_{p=1}^{N}\widehat{\mathbf{h}}^T_{\mathbf{x}}V_p(t_i)\zeta_p\right] w_i L_1(t_i) \\
&= -\left[\frac{(\tau_f - \tau_0)}{2}\widehat{\mathbf{f}}^T_{\mathbf{x}} \sum_{j=1}^{N} W_j(0)\gamma_j + \sum_{j=1}^{N}\dot{W}_j(0)\gamma_j + \sum_{p=1}^{N}\widehat{\mathbf{h}}^T_{\mathbf{x}}V_p(0)\zeta_p\right] w_1 \\
&= 0; \quad \text{(explicitly)}.
\end{aligned} \tag{7.62}
$$

Thus, for the Gauss pseudospecral method, all the closure conditions are explicitly satisfied. The costates can always be approximated using Eqn. (3.59) as,

$$\widehat{\lambda}(t) = \sum_{j=1}^{N}\gamma_j W_j = \sum_{j=2}^{N-1}\gamma_j\frac{L_j(t)}{w_j} + \gamma_N, \tag{7.63}$$

$$\widehat{\lambda}_k = \widehat{\lambda}(t_k) = \sum_{j=2}^{N-1}\gamma_j\frac{\delta_{jk}}{w_j} + \gamma_N = \frac{\gamma_k}{w_k} + \gamma_N; \quad k = 2, .., N-1, \tag{7.64}$$

$$\widehat{\lambda}_N = \widehat{\lambda}(t_N) = \gamma_N. \tag{7.65}$$

λ_1 is obtained from the boundary condition,

$$\widehat{\Psi}_{\mathbf{x}(0)} + \widehat{\psi}^T_{\mathbf{x}(0)}\eta + \mu_0 K_0 + \gamma_1 = 0. \tag{7.66}$$

D. Conclusions

It was shown that the pseudospectral methods can be formulated as a special case of weighted residual method presented in Chapter III. Already established costate estimation results for pseudospectral methods were also derived in the weighted residual framework.

CHAPTER VIII

PERFORMANCE COMPARISON RESULTS FOR DIRECT TRANSCRIPTION METHODS

This chapter presents the hierarchical relationship between various direct transcription methods discussed so far in this dissertation and compares their performance with respect to one another. The comparison is done at three levels. First, MHSP, LSM_{oc}, and GMM_{oc} are compared with each other using spline approximations SPL-34 and SPL-3r for example problems defined in Chapter IV. Next, spectral methods s-LSM_{oc} and s-GMM_{oc} are compared with the existing methods in the same category, i.e. LPS, GPS and RPS. Finally, performance of local approximation methods is compared with the spectral methods for a number of non-smooth application problems.

A. Hierarchy of Direct Transcription Methods

In past four chapters, three new transcription methods and the existing pseudospectral methods were derived under a unifying framework of weighted residual formulation. The hierarchical relationship between all the methods discussed in the previous chapters is shown in Figure (35). There are two broad categories of these methods. In the first category, both local and global approximating functions can be employed and the number of quadrature nodes are independent from the number of approximating basis functions. This category includes MHSP, LSM_{oc} and GMM_{oc}. In the second category, global interpolating polynomials are used as approximating functions and the quadrature scheme is derived from the interpolation nodes. This category includes all the pseudospectral methods along with s-LSM_{oc} and s-GMM_{oc}.

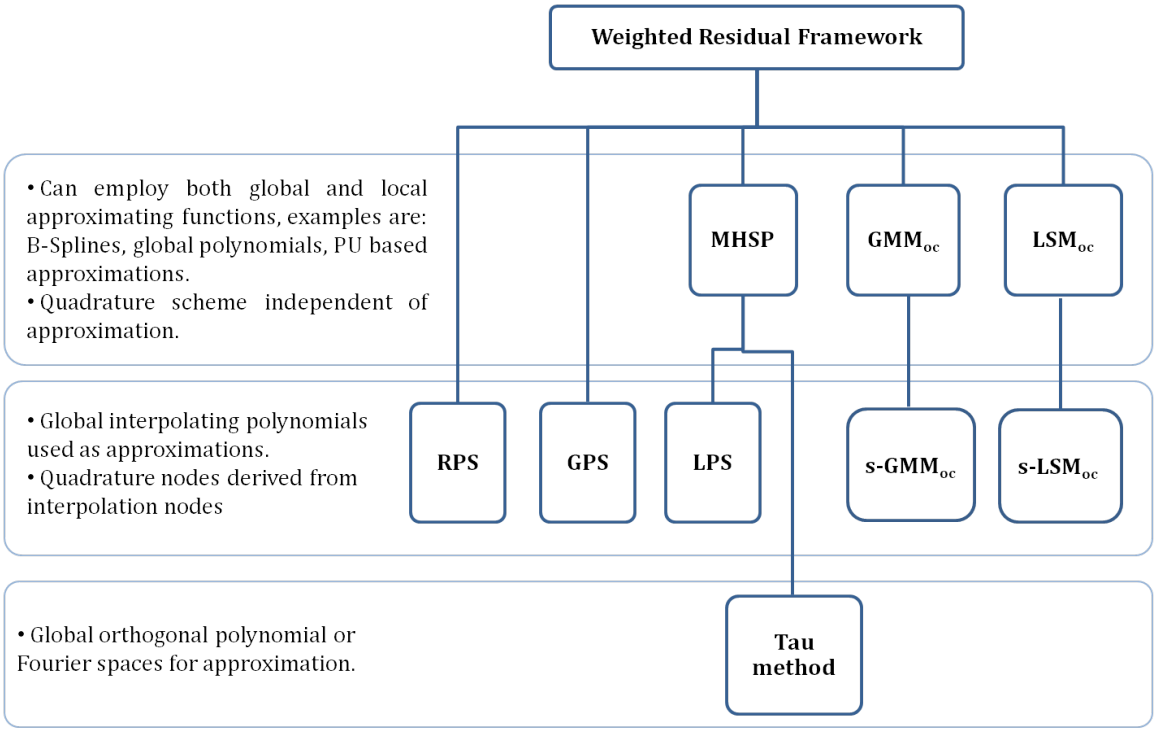


Fig. 35. Hierarchy of direct methods under weighted residual framework.

B. Comparison between MHSP, LSM$_{oc}$ and GMM$_{oc}$

In this section, we compare the relative performance of MHSP, LSM$_{oc}$ and GMM$_{oc}$ in solving four example problems defined in Chapter IV. For the first two examples, SPL-3r approximation is used, while SPL-34 approximation is used for examples 3 and 4.

Comparison results for Example 1 are shown in Figure (36). It is seen that MHSP performs poorly compared to LSM$_{oc}$ and GMM$_{oc}$. The performance of LSM$_{oc}$ and

GMM_{oc} is comparable to each other.

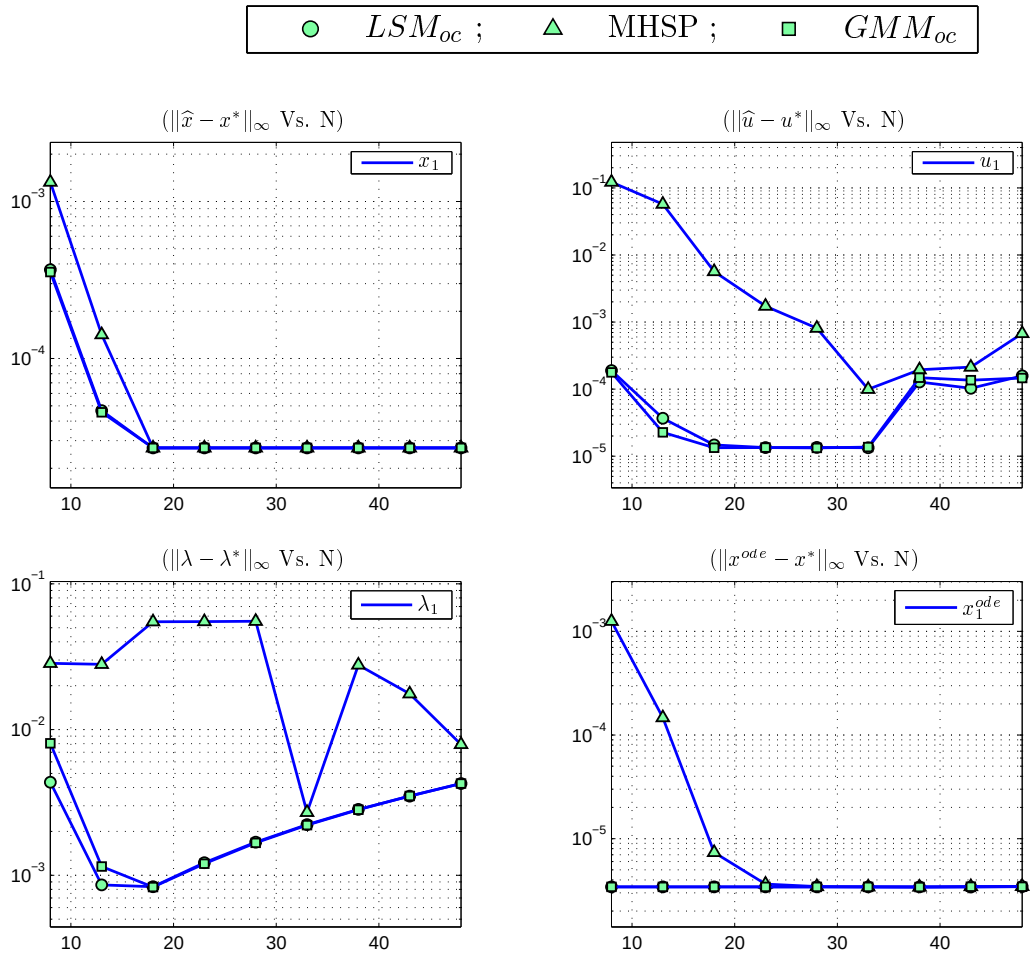


Fig. 36. Example 1: Comparison of results for MHSP, LSM_{oc} and GMM_{oc}.

Figure (37) shows the results for Example 2. For this example, it is seen that GMM_{oc} performs better than MHSP and LSM_{oc} for lower orders of approximation. At higher levels of approximation, all three methods are equally good.

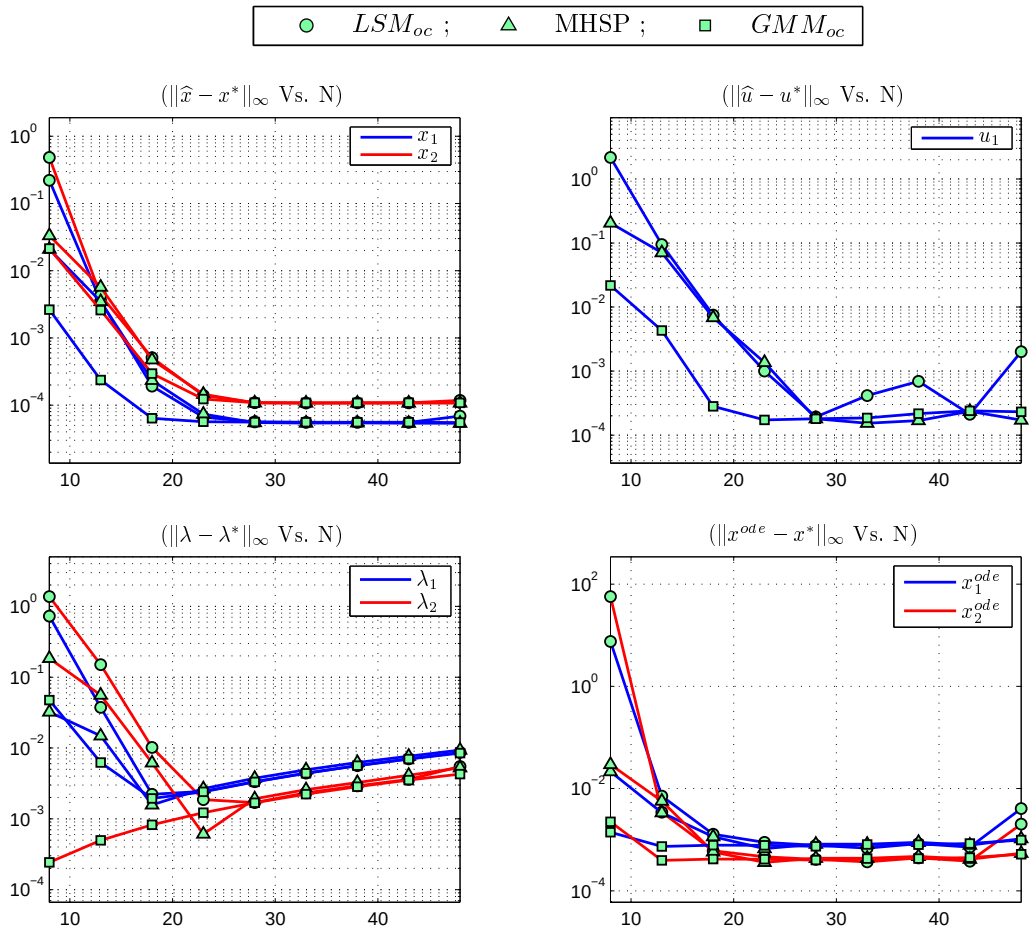


Fig. 37. Example 2: Comparison of results for MHSP, LSM_{oc} and GMM_{oc}.

Results for Example 3 are depicted in Figure (38). For this problem, overall performance of all three methods is comparable to each other. Example 4 results are shown in Figure (39). It is seen that primal convergence is comparable for all three methods, but MHSP costates do not converge.

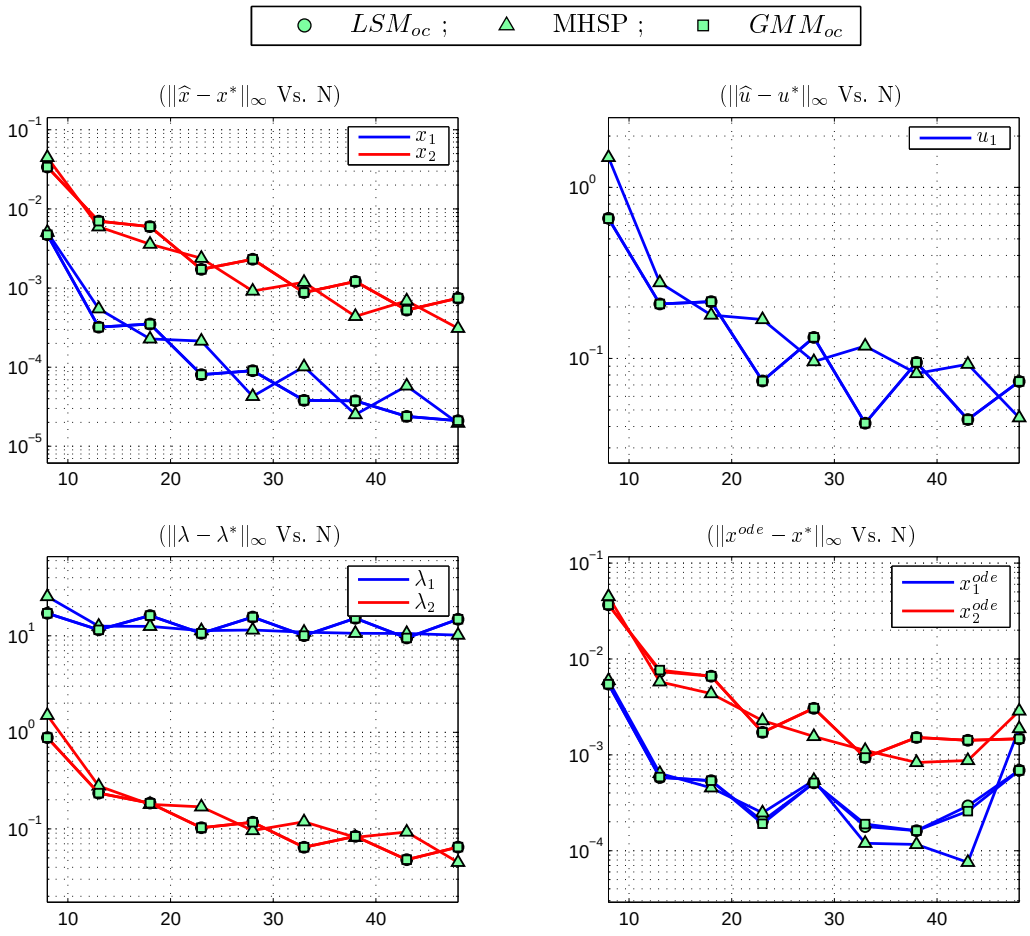


Fig. 38. Example 3: Comparison of results for MHSP, LSM_{oc} and GMM_{oc}.

C. Comparison between s-LSM_{oc}, s-GMM_{oc} and PS Methods

In this section, performance of all spectral methods is compared for the example problems. Five different methods are implemented under the same computational

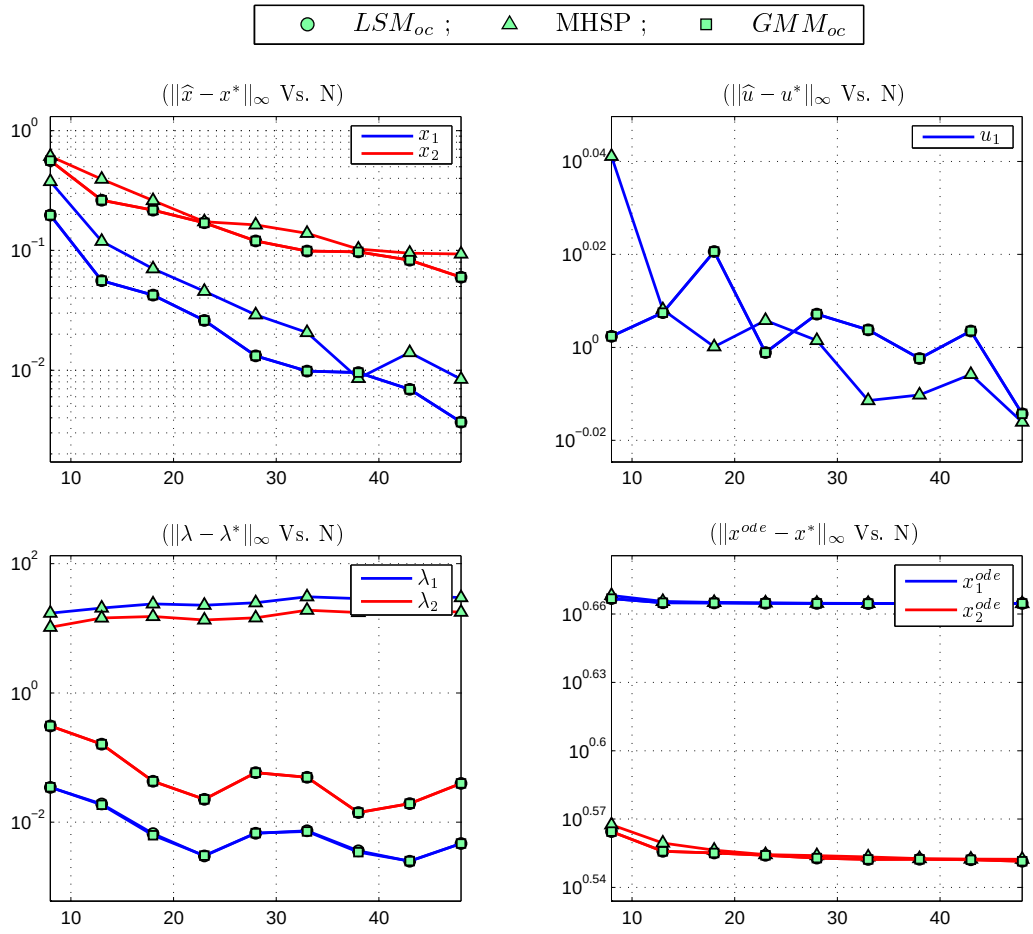


Fig. 39. Example 4: Comparison of results for MHSP, LSM_{oc} and GMM_{oc}.

environment using the same initial guess. These methods are: s-LSM_{oc}, s-GMM_{oc}, LPS, GPS and RPS. Two of these methods, s-LSM_{oc} and s-GMM_{oc}, are developed in this dissertation while the other three are well established methods in the literature.

Results form Example 1 are shown in Figure (40). It is seen that primal convergence is comparable for all five methods. However, LPS does not perform very well in estimating the costates.

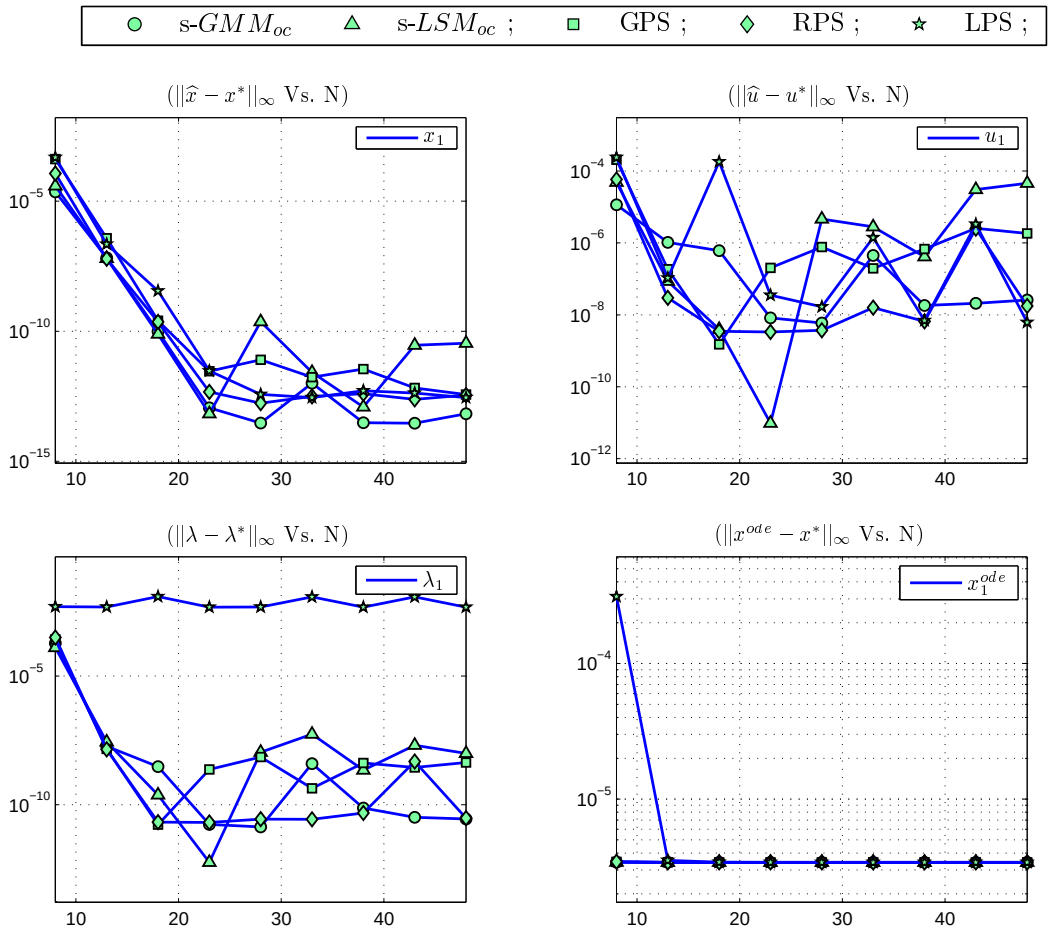


Fig. 40. Example 1: Comparison of results for s-LSM_{oc}, s-GMM_{oc}, GPS, RPS and LPS.

Figure (41) shows the results for Example 2. Performance of all five methods is comparable while s-GMM$_{oc}$ performs relatively better for lower order of approximations.

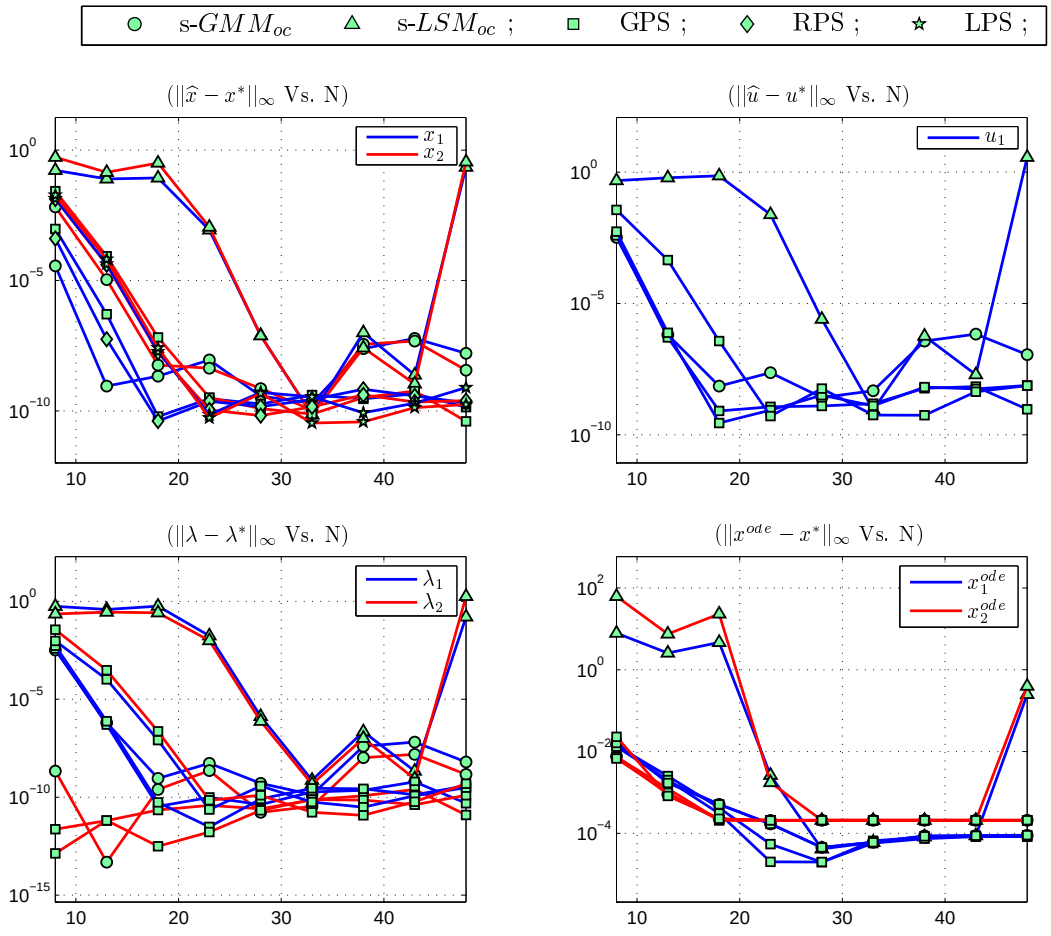


Fig. 41. Example 2: Comparison of results for s-LSM$_{oc}$, s-GMM$_{oc}$, GPS, RPS and LPS.

Results for Example 3 are depicted in Figure (42). For this example, all five methods perform comparably. Example 4 results are shown in Figure (43). In this example, we see that LPS consistently lags in performance compared to other four methods and at the same time costates for LPS do not converge.

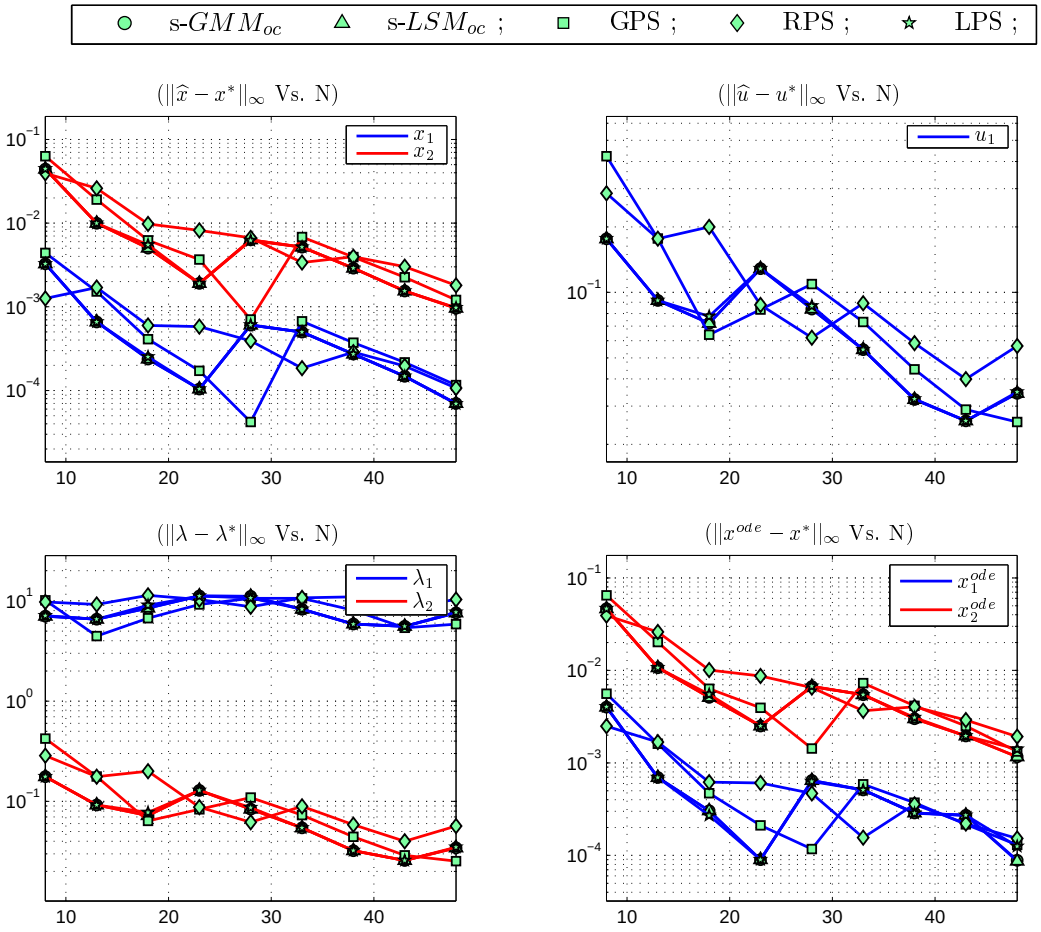


Fig. 42. Example 3: Comparison of results for s-LSM_{oc}, s-GMM_{oc}, GPS, RPS and LPS.

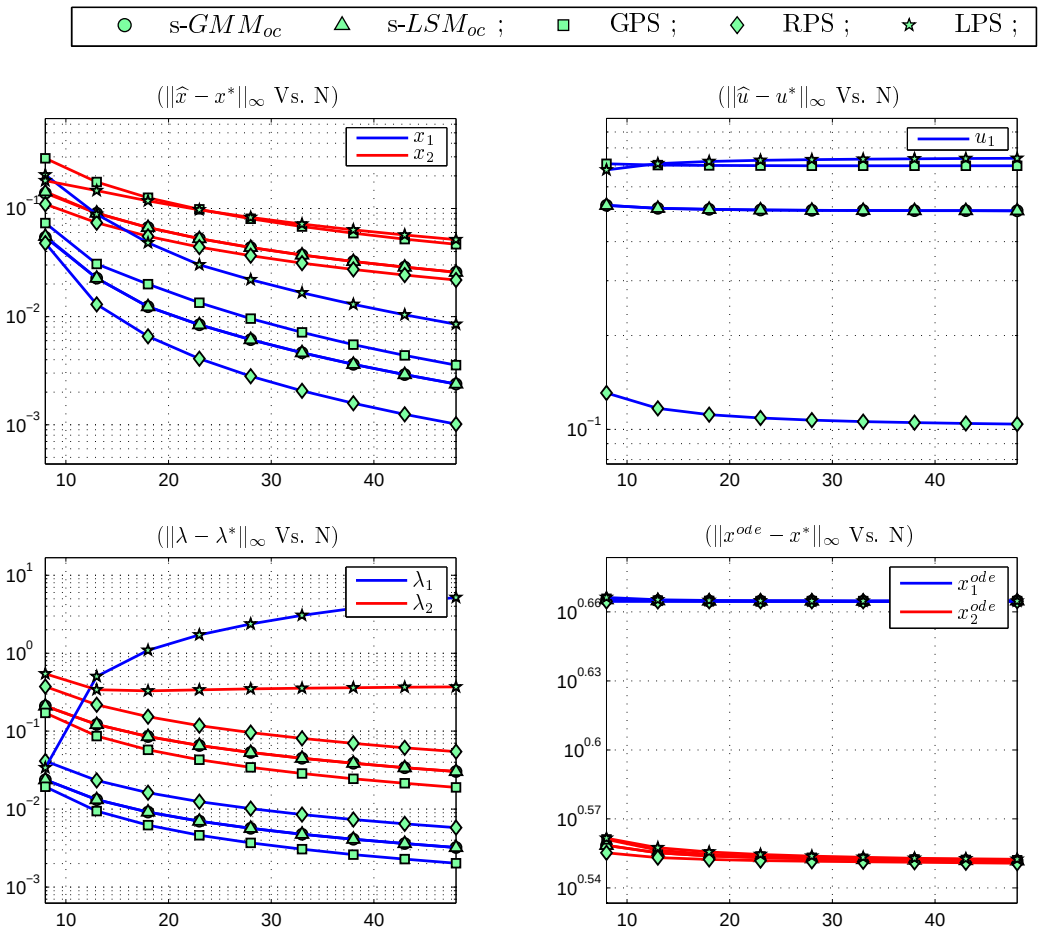


Fig. 43. Example 4: Comparison of results for s-LSM_{oc}, s-GMM_{oc}, GPS, RPS and LPS.

D. Comparison between Global and Local Methods

From the previous discussion, it seems that GMM_{oc} is the best choice for implementation with local basis. Also, amongst the existing spectral methods, GPS and RPS demonstrate higher accuracy. In this section, we compare the performance of GMM_{oc} with SPL-34 approximation and the spectral methods. First, results are compared for Example 2. We see that spectral methods exhibit very high accuracy compared to GMM_{oc} for this example. The results for this example are depicted in

Figure (44).

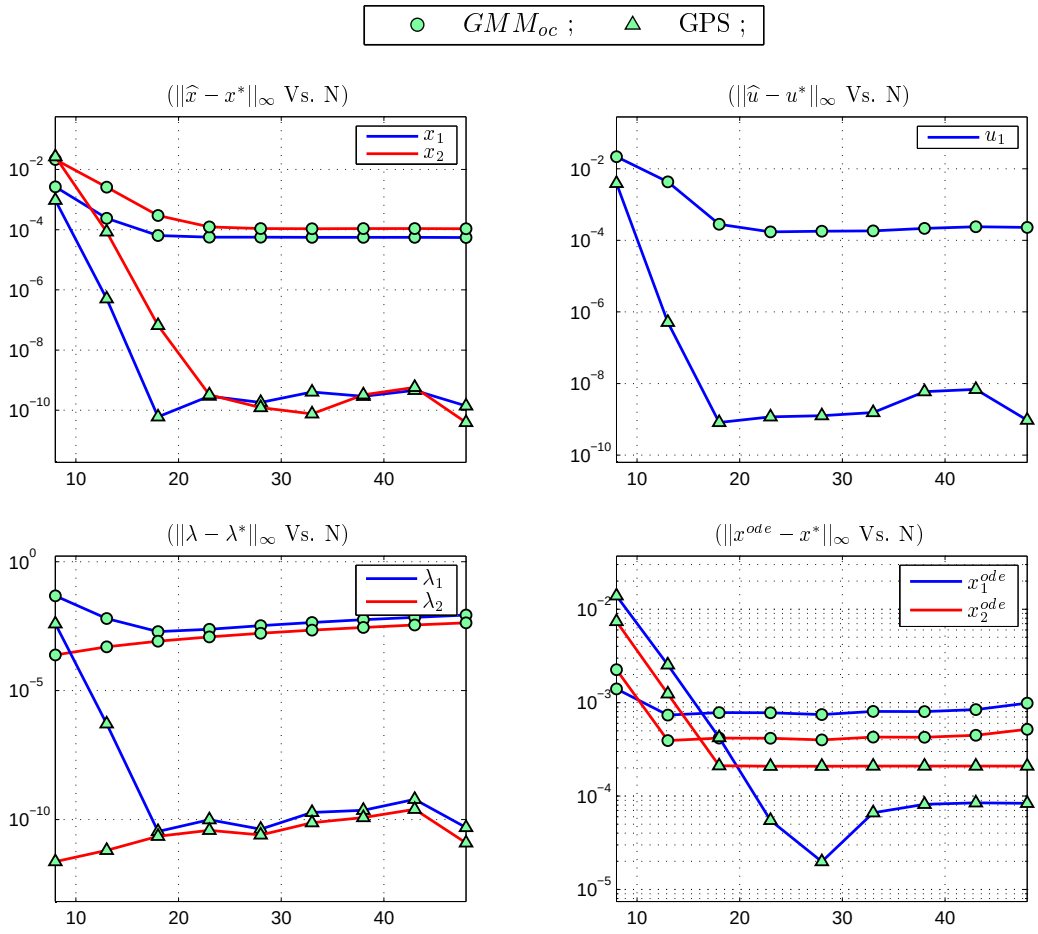


Fig. 44. Example 2: Comparison of results for $\mathrm{GMM_{oc}}$ and GPS.

Next, we take a non-smooth problem defined in Ref. [62]. Since no analytical solution exists for this problem, the indirect solution given in Ref. [62] is taken as the true solution for comparisons. The optimal control problem is to find the history of T and β which drive a spacecraft form its initial state to the final, while maximizing the final mass of the spacecraft.

$$\text{Maximize: } m(t_f) \tag{8.1}$$

$$\text{Subject to: } \dot{r} = u, \qquad \dot{\theta} = \frac{v}{r} \tag{8.2}$$

$$\dot{u} = \frac{v^2}{r} - \frac{\mu}{r^2} + \frac{T}{m}\sin(\beta), \qquad \dot{v} = -\frac{uv}{r} + \frac{T}{m}\cos(\beta) \tag{8.3}$$

$$\dot{m} = -\frac{T}{g_0 I_{sp}}, \qquad 0 \leq T \leq T_{max}, \quad t_f = 4.1285. \tag{8.4}$$

Values of all constants are taken from Ref. [62] with initial and final conditions specified as,

r(0)	1	r(t_f)	1.05242919219003
$\theta(0)$	0	$\theta(t_f)$	3.99191781862267
u(0)	0	u(t_f)	0
v(0)	1	v(t_f)	0.97477314754443
m(0)	1	m(t_f)	maximum

The results for this example are shown in Figure (45). It is seen that GMM_{oc} exhibits smooth convergence towards the true solution, while GPS convergence is highly irregular. Also, even at high values of N, GPS and LPS solutions do not converge well to the benchmark solution.

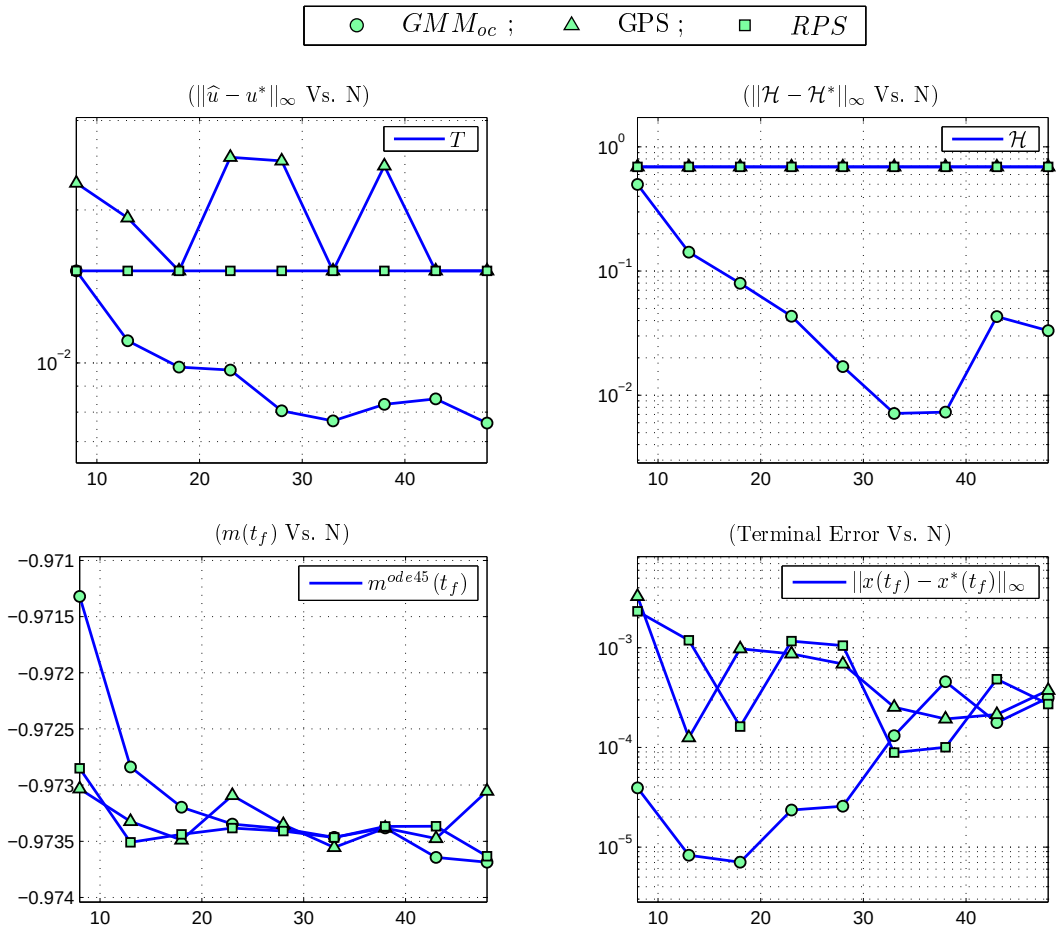


Fig. 45. Rendezvous Problem: Comparison of results for $\mathrm{GMM_{oc}}$

The evolution of control profile is depicted in Figure (46). The smooth convergence of solution for GMM_{oc} demonstrates its potential of implementation in an h-adaptive setting.

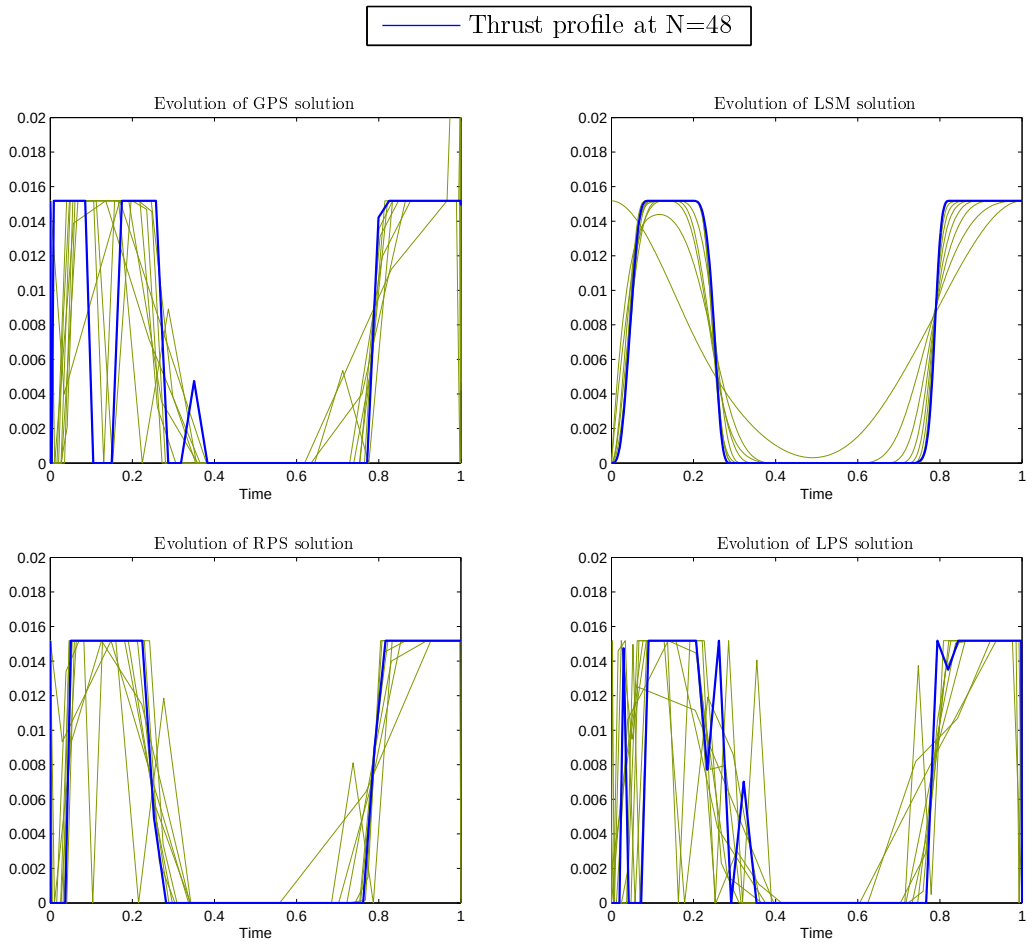


Fig. 46. Rendezvous Problem: Evolution of control profile with increasing N

E. Conclusions

We draw the following conclusions from the observations in this chapter:

1. For B-Spline approximation, GMM_{oc} is the best of choice.

2. s-LSM_{oc} and s-GMM_{oc} perform comparably with the existing pseudospectral methods. Their performance is significantly better than LPS for all the test examples. Further, s-LSM_{oc} and s-GMM_{oc} do not suffer from any boundary defects like in GPS and RPS where extrapolations are required to find control values at boundary points.

3. For non-smooth problems, GMM_{oc} is quite stable and has higher convergence rates than the spectral methods.

4. GMM formulation turn out to be the best choice for both local and global implementations. It does not suffer from any boundary defects, provides excellent costate results, can be implemented in an h-adaptive refinement setting using B-Splines and can be implemented in global polynomial form for spectral accuracy.

CHAPTER IX

A-POSTERIORI ERROR ESTIMATION AND H-ADAPTIVE GRID REFINEMENT

Most of the differentiation based direct methods reported in the literature either use global approximations, like polynomials and harmonics, or B-Splines. Global approximating functions should work well for problems which have smooth solutions without large local variations in time. To approximate solutions having localized irregular features such as corners and discontinuities, the approximating space should have local refinement capability. Another desirable feature would be the inclusion of some special functions, such as exponentials or sinusoids, that characterize the known local behavior of the system, into the basis. While B-Splines have local approximation property, there is no direct mechanism of local p-refinement or inclusion of any special functions into the basis.

In this chapter, we describe the use of partition of unity based approximations for nonlinear optimal control using the LSM_{oc}. The advantages of using PU approximations in the present setting are, (i) local support, (ii) hp adaptivity, (iii) ability to incorporate any local approximations and (iv) control over local smoothness.

A. Numerical Implementation of LSM_{oc}

To implement PU approximations in the LSM_{oc}, let $\mathbf{x}(t)$ be approximated using Eqn. (2.62) as,

$$\widehat{\mathbf{x}}(t) = \sum_{i=1}^{N} \sum_{j=1}^{n_i} \psi_{ij}(t) W_i(t) \tilde{\alpha}_{ij}, \tag{9.1}$$

where N and n_i are defined based on the discussion in Section IV-A. By defining an index transformation we can write,

$$\phi_k^{\mathbf{x}} = \psi_{ij} W_i \Big|_{k=\sum_{l=1}^{(i-1)} n_l + j}, \quad \alpha_k = \tilde{\alpha}_{ij} \Big|_{k=\sum_{l=1}^{(i-1)} n_l + j}, \tag{9.2}$$

which implies,

$$\widehat{\mathbf{x}}(t) = \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \phi_k^{\mathbf{x}}; \quad N_{\mathbf{x}} = \sum_{l=1}^{N} n_l. \tag{9.3}$$

Eqn. (9.3) and Eqn. (3.7) are in the same form. Similarly, we can parameterize $\mathbf{u}(t)$ and $\mathbf{s}(t)$ as PU approximations and write them in the same form as in Eqn. (3.9) and Eqn. (3.10). Thus, a nonlinear programming problem can be formulated based on Chapter V Section B. To numerically evaluate the integral expressions in Eqn. (5.8) and Eqn. (5.11), we present two alternatives. First approach is to use a large number of points to form a uniform grid in [0,1], and evaluate the integral expressions using trapezoid rule. In the second approach, we may adopt a numerical quadrature scheme where quadrature nodes are filled between the nodes used for the approximation. Since there are three types of approximations, $\mathbb{V}_{\mathbf{x}}$, $\mathbb{V}_{\mathbf{u}}$ and $\mathbb{V}_{\mathbf{s}}$, nodes used in each approximation are superimposed onto the interval [0,1]. The quadrature nodes are then filled in between the points of the superimposed grid. This results in the higher quadrature nodes density in the areas of low regularity. We use the Legendre-Gauss (LG) scheme to place the quadrature nodes and to evaluate the corresponding weights. The arrangement of quadrature nodes on a typical grid is shown in Figure (47). Let $\{t_m\}_{m=1}^{N_q}$ be the quadrature nodes with corresponding weights w_m. The NLP is formulated based on Section III-A as following. Determine $\{\alpha_k \in \mathbb{R}^n\}_{k=1}^{N_{\mathbf{x}}}, \{\beta_k \in \mathbb{R}^m\}_{k=1}^{N_{\mathbf{u}}}, \{\varsigma_k \in \mathbb{R}^q\}_{k=1}^{N_{\mathbf{s}}}$, $\nu \in \mathbb{R}^n$ and time instance τ_f, that minimize the cost,

$$\widehat{J} = \widehat{\Psi}(\alpha_k, \phi_k^{\mathbf{x}}(1), \tau_f), \tag{9.4}$$

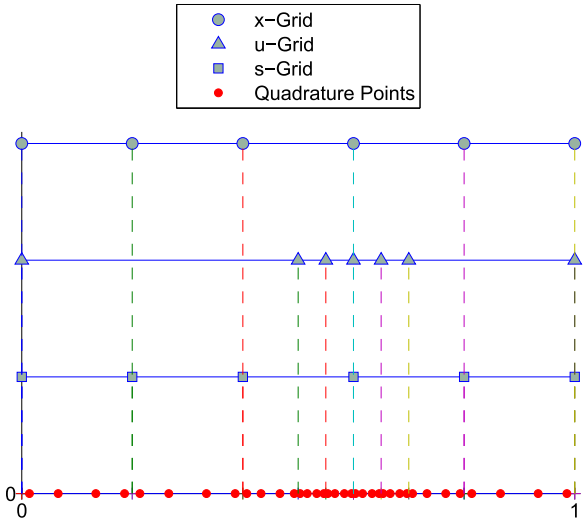


Fig. 47. Arrangement of quadrature nodes for a nonuniform grid in domain [0,1].

subject to the constraints,

$$\sum_{m=1}^{N_q}[\tau_f \widehat{\mathbf{f}}_{\mathbf{x}}^T(t_m)\phi_j^{\mathbf{x}}(t_m) - \mathbf{I}\dot{\phi}_j^{\mathbf{x}}(t_m)][\tau_f \widehat{\mathbf{f}}(t_m) - \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \dot{\phi}_k^{\mathbf{x}}(t_m)]w_m + \frac{1}{2}\nu\phi_j^{\mathbf{x}}(0) = 0;$$

$$j = 1, .., N_{\mathbf{x}}, \tag{9.5}$$

$$\sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \phi_k^{\mathbf{x}}(0) - \mathbf{x}_0 = 0, \quad \widehat{\psi}(\alpha_k, \phi_k^{\mathbf{x}}(1), \tau_f) = 0, \tag{9.6}$$

$$\sum_{m=1}^{N_q}[\widehat{\mathbf{h}}(t_m) + \sum_{k=1}^{N_{\mathbf{s}}} \varsigma_k \phi_k^{\mathbf{s}}(t_m) \circ \sum_{l=1}^{N_{\mathbf{s}}} \varsigma_l \phi_l^{\mathbf{s}}(t_m)]\phi_j^{\mathbf{s}}(t_m)w_m = 0; \qquad j = 1, .., N_{\mathbf{s}}, \tag{9.7}$$

were $\widehat{\mathbf{f}}_{\mathbf{x}}^T(t_m) = \widehat{\mathbf{f}}_{\mathbf{x}}^T(\alpha_k, \beta_k, \phi_k^{\mathbf{x}}(t_m), \phi_k^{\mathbf{u}}(t_m))$, and $\widehat{\mathbf{h}}(t_m) = \widehat{\mathbf{h}}(\alpha_k, \beta_k, \phi_k^{\mathbf{x}}(t_m), \phi_k^{\mathbf{u}}(t_m))$. The nonlinear programming problem defined here can be solved using any of the available optimization software like 'fmincon' in MATLAB, SNOPT [12], or NPSOL [13].

B. A Posteriori Error Estimation

To obtain high accuracy solutions, it is computationally inefficient to employ a high order approximation uniformly over the whole domain as it results in over killing the problem in the areas of low variability. Since the true solution is not known, it is desirable to estimate the error in the computed solution, and have an adaptive strategy to improve the local approximability of the function space selectively in the regions having large errors.

This section presents an *a posteriori* error estimation method for optimal control problems, which takes into account the errors in both feasibility and optimality of the solution. This method is based on the variational analysis of the trajectory residuals of EL equations. Let $\mathbf{y}(t)$ represents all the trajectory variables associated with $\mathcal{M}_\lambda$. Then at any time instant t, define the error $\mathbf{e}(t)$ to be the variations of all the trajectory variables from the true solution. So that,

$$\mathbf{y}(t) = \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{u}(t) \\ \lambda(t) \\ \xi(t) \\ \mathbf{s}(t) \\ \dot{\mathbf{x}}(t) \\ \dot{\lambda}(t) \end{bmatrix}, \quad \mathbf{e}(t) = \delta\mathbf{y}(t) = \begin{bmatrix} \delta\mathbf{x}(t) \\ \delta\mathbf{u}(t) \\ \delta\lambda(t) \\ \delta\xi(t) \\ \delta\mathbf{s}(t) \\ \delta\dot{\mathbf{x}}(t) \\ \delta\dot{\lambda}(t) \end{bmatrix}. \tag{9.8}$$

The residuals associated with all the trajectory constraints in $\mathcal{M}_\lambda$ can be written as,

$$R(\mathbf{y}(t)) = \begin{bmatrix} \dot{\mathbf{x}}(t) - \tau_f \mathbf{f}(\mathbf{x},\mathbf{u}) \\ \tau_f \mathbf{f}_{\mathbf{u}}^T(\mathbf{x},\mathbf{u})\lambda(t) + \mathbf{h}_{\mathbf{u}}^T(\mathbf{x},\mathbf{u})\xi(t) \\ \dot{\lambda}(t) + \tau_f \mathbf{f}_{\mathbf{x}}^T(\mathbf{x},\mathbf{u})\lambda(t) + \mathbf{h}_{\mathbf{x}}^T(\mathbf{x},\mathbf{u})\xi(t) \\ \mathbf{h}(\mathbf{x},\mathbf{u}) + \mathbf{s}(t) \circ \mathbf{s}(t) \\ 2\xi(t) \circ \mathbf{s}(t) \end{bmatrix}.$$

Let $\mathbf{y}^*(t)$ be the true solution of the optimal control problem, and $\widehat{\mathbf{y}}(t) = \mathbf{y}^*(t) + \delta\mathbf{y}(t)$. Taking the first variation of $R(\mathbf{y}(t))$ we get,

$$R(\mathbf{y}^*) = R(\widehat{\mathbf{y}} - \delta\mathbf{y}) = R(\widehat{\mathbf{y}}) - \left.\frac{\partial R(\mathbf{y})}{\partial \mathbf{y}}\right|_{\mathbf{y}=\widehat{\mathbf{y}}} \delta\mathbf{y}. \tag{9.9}$$

Since $R(\mathbf{y}^*) = 0$, we get,

$$R(\widehat{\mathbf{y}}) = \left.\frac{\partial R(\mathbf{y})}{\partial \mathbf{y}}\right|_{\mathbf{y}=\widehat{\mathbf{y}}} \delta\mathbf{y},$$

$$R^T(\widehat{\mathbf{y}})R(\widehat{\mathbf{y}}) = \mathbf{e}^T(t) R_{\mathbf{y}}^T(\widehat{\mathbf{y}}) R_{\mathbf{y}}(\widehat{\mathbf{y}}) \mathbf{e}(t).$$

From the theory of spectral decomposition,

$$\frac{R^T(\widehat{\mathbf{y}})R(\widehat{\mathbf{y}})}{\mathbf{e}^T(t)\mathbf{e}(t)} \leq \lambda_{max}(t)(R_{\mathbf{y}}^T(\widehat{\mathbf{y}})R_{\mathbf{y}}(\widehat{\mathbf{y}})) = \sigma_{max}^2(t)(R_{\mathbf{y}}(\widehat{\mathbf{y}})), \tag{9.10}$$

where $\sigma_{max}^2(t)$ is the maximum singular value of the matrix $R_{\mathbf{y}}(\widehat{\mathbf{y}})$ evaluated at time t. Re-arranging Eqn. (9.10) we get,

$$\mathbf{e}^T(t)\mathbf{e}(t) \geq \frac{R^T(\widehat{\mathbf{y}})R(\widehat{\mathbf{y}})}{\sigma_{max}^2(R_{\mathbf{y}}(\widehat{\mathbf{y}}))} = \|\underline{\mathbf{e}}(t)\|_2. \tag{9.11}$$

The quantity in the right hand side of Eqn. (9.11), denoted as $\|\underline{\mathbf{e}}(t)\|_2$, gives us a lower bound on the error norm $\|\mathbf{e}(t)\|_2$. Although $\|\underline{\mathbf{e}}(t)\|_2$ does not represent a rigorous estimate of the true error in the solution at time t, in the present treatment we adopt a heuristic approach by assuming that the true error lies at its lower bound. So that

$\|\underline{\mathbf{e}}(t)\|_2$ can be used as a measure for adaptive grid refinement. Numerical results demonstrate that the heuristic approach used here performs quite satisfactorily for the example problems presented in this chapter.

C. h-Adaptive Local Refinement Algorithm

Depending upon the type of approximation used, particle based or element based, an h-adaptive algorithm locally increases the particle density or element density in the regions where refinement is required. To identify these regions, the global domain [0,1] is partitioned into a set of subdomains $\{\mathcal{S}_k\}_{k=1}^{N}$ such that,

$$\mathcal{S}_k := [T_{k-1}, T_k]; \quad 0 = T_0 < T_1 < .. < T_N = 1.$$

Note that when element based approximations are used, such as B-Splines, $\mathcal{S}_k$'s represent the element mesh itself. When using a particle based approximation, $\mathcal{S}_k$'s can be constructed by taking T_k's as particle locations.

Next, we define a refinement function $\mathcal{R} : \{\mathcal{S}_k\} \rightarrow \mathbb{R}$ such that,

$$\mathcal{R}_{\mathcal{S}_k} = \Delta_k \int_{\mathcal{S}_k} \|\underline{\mathbf{e}}(t)\|_2 dt, \tag{9.12}$$

where $\Delta_i := (T_i - T_{i-1})$. Based on the value of refinement function, a set of rules are defined to refine the element mesh or to insert new particles. We bisect r number of particular subdomains corresponding to the r largest values of refinement function. Let $\mathcal{R}^* = \{\downarrow \mathcal{R}\}_{[1,2,..,r]}$ be the first r number of elements in the set formed by arranging $\mathcal{R}_{\mathcal{S}_k}$ in descending order. Then the subdomains which are bisected are found as,

$$\mathcal{S}^* = \{\mathcal{S}_k | \mathcal{R}_{\mathcal{S}_k} \in \mathcal{R}^*\}, \tag{9.13}$$

The overall algorithm can be summarized in the following steps:

1. Compute the solution using current approximation.
2. Evaluate the refinement function for each subdomain $\mathcal{S}_k$ using Eqn. (9.12).
3. Using Eqn. (9.13), select r number of subdomains for bisection. Construct approximation based on the new set of particles or elements.
4. If $R^T(\hat{y})R(\hat{y}) < \epsilon$, where ϵ is a pre-selected error bound, exit. Otherwise, go to step 1.

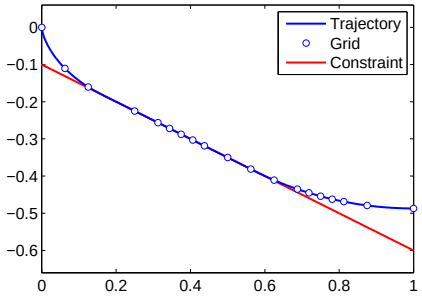
D. Numerical Examples

In this section, a number of example problems are solved to demonstrate the working of our method outlined in the previous sections. Problems are selected from the literature which have discontinuities or corners in their solutions. The solutions are obtained using SNOPT as the optimizer in MATLAB environment. A direct transcription MATLAB-toolbox named OPTRAGEN-3 has been developed. It takes user-friendly inputs and generates all the required numerical setup for solving the NLP.

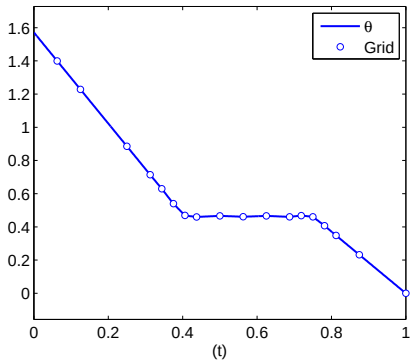
1. Example 1: Brachistochrone Problem

$$
\begin{aligned}
\text{Minimize: } & J = t_f \\
\text{Subject to: } & \dot{x} = V\cos(\theta) \\
& \dot{y} = V\sin(\theta) \\
& \dot{V} = 10\sin(\theta) \\
& x(0) = 0, \quad y(0) = 0, \quad V(0) = 0, \\
& x(t_f) = 1, \quad y - 0.5x - 0.1 \leq 0.
\end{aligned}
$$

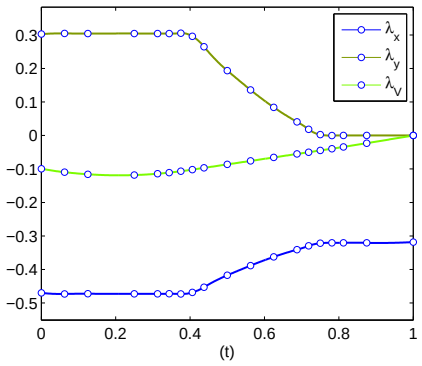
This is a problem with a first-order state variable inequality constraint. The analytical solution for this problem can be found in Ref. [10]. We solve this problem by approximating the state variables as B-Splines with 4^{th} order piece-wise polynomials having 3^{rd} order smoothness. The control variable $\theta(t)$ is approximated as a B-Spline with 3^{rd} order piece-wise polynomials having 2^{nd} order smoothness. The algorithm starts with a uniform grid of 4 elements and performs 15 refinement iterations. The results are shown in Figure (48). We compare the obtained results with the analytical solution. The $\mathcal{L}_2$-norm of errors in all the trajectory variables are listed in Table II. We see that adaptive algorithm rightly captures the corner regions in $\theta(t)$, and



(a) Trajectory in (x-y) plane.



(b) Control solution $\theta(t)$.



(c) Costates λ_x,λ_y and λ_V.

Fig. 48. Results for Brachistochrone problem.

the final grid density is relatively higher in those regions. Also, we obtain excellent costate estimates. The value of jumps in the costates λ_x and λ_y are 0.152344 and -0.304315 respectively which match with the true solution to the 3^{rd} decimal place. This problem demonstrates the applicability of the our algorithm in the presence of state inequality constraints.

Table II. Results for Brachistochrone problem

Quantity of interest	Value
$\mathcal{L}_2$ error in $x(t)$	7.01e-008
$\mathcal{L}_2$ error in $y(t)$	1.97e-007
$\mathcal{L}_2$ error in $\theta(t)$	2.25e-006
EqvI, EqvII	$\begin{bmatrix} 2.74\text{e-}003 & 2.10\text{e-}003 \\ -1.74\text{e-}003 & 2.62\text{e-}005 \\ 5.87\text{e-}004 & 2.63\text{e-}005 \end{bmatrix}$
$\|\mathcal{H}(t)\|_2$	3.33e-007
Optimal cost	0.580058

2. Example 2: Robot Path Planning

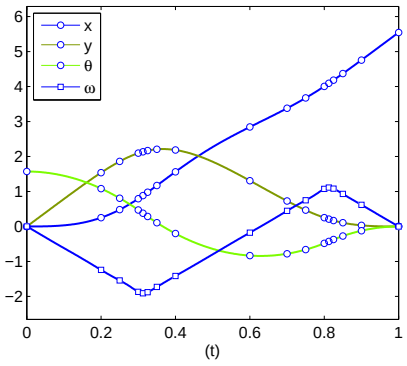
$$
\begin{aligned}
\text{Minimize: } & J = t_f \\
\text{Subject to: } & \dot{x} = 2\cos(\theta) \\
& \dot{y} = 2\sin(\theta) \\
& \dot{\theta} = \omega \\
& \dot{\omega} = u \\
& x(0) = 0, \quad y(0) = 0, \quad \theta(0) = \frac{\pi}{2}, \quad \omega(0) = 0, \\
& y(t_f) = 0, \quad \theta(t_f) = 0, \quad \omega(t_f) = 0, \\
& |u| \leq \frac{\pi}{2}.
\end{aligned}
$$

This problem represents a ground robot moving in the x-y plane with a constant velocity of 2 m/s. The applied control is in the form of angular acceleration and is bounded with maximum possible magnitude of $\frac{\pi}{2}$ rad/sec^2. The initial heading angle is $\frac{\pi}{2}$ and the goal is to align the robot along x-axis in minimum time. The solution to this

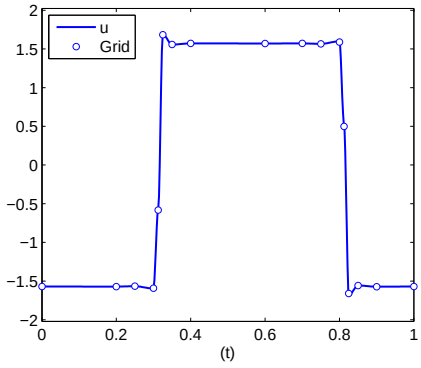
Table III. Results for robot path planning problem

Quantity of interest	Value
EqvI, EqvII	$\begin{bmatrix} 2.26\text{e-}003 & -4.16\text{e-}006 \\ -1.56\text{e-}002 & -1.17\text{e-}002 \\ -3.11\text{e-}002 & 3.24\text{e-}002 \\ -3.44\text{e-}002 & -1.78\text{e-}002 \end{bmatrix}$
$\|\mathcal{H}(t)\|_2$	2.47e-003
Optimal cost	3.94593

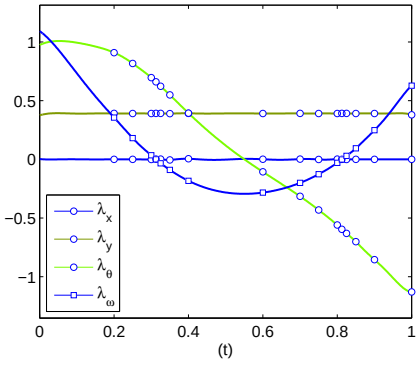
problem has bang-bang structure. We solve this problem by approximating the states as B-Splines with 5^{th} order piece-wise polynomials having 4^{th} order smoothness. The control u is approximated using the 1^{st} order GLOMAP weight functions as defined in Eqn. (2.83). The algorithm starts with a uniform grid of 5 elements and performs 12 refinement iterations. The trajectories are plotted in Figure (49). We see that the adaptive algorithm makes the grid points to concentrate near the discontinuity in u, while x_4 has a corner at the same time location. Also, the switching structure in u is well captured by the GLOMAP functions. The results are presented in Table III.



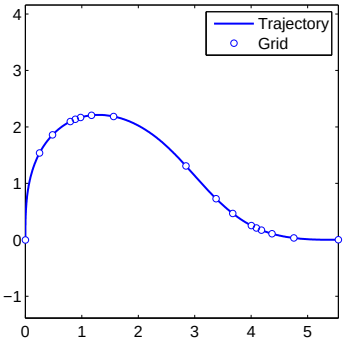
(a) States x,y,θ and ω.



(b) Control u.



(c) Costates λ_x,λ_y, λ_θ and λ_ω.



(d) Robot path in (x-y) plane.

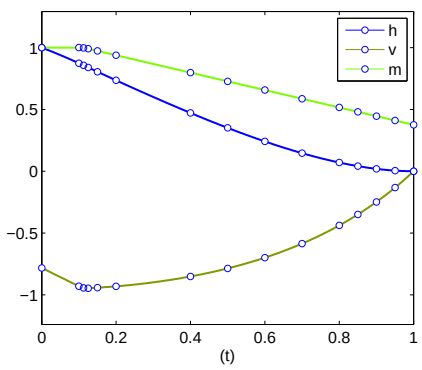
Fig. 49. Results for robot path planning problem.

3. Example 3: Moonlanding Problem

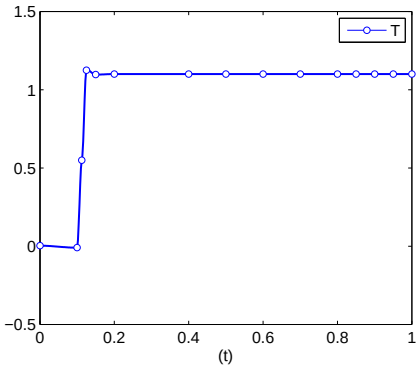
We consider the moonlanding problem as defined in Ref. [63],

$$\begin{aligned}
\text{Minimize: } & J = -m(t_f) \\
\text{Subject to: } & \dot{h} = v \\
& \dot{v} = -1 + \frac{T}{m} \\
& \dot{m} = -\frac{T}{2.3} \\
& h(0) = 1, \quad v(0) = -0.783, \quad m(0) = 1, \\
& h(t_f) = 0, \quad v(t_f) = 0, \quad 0 \leq T \leq 1.1
\end{aligned}$$

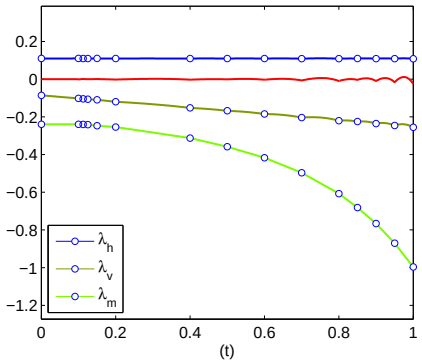
Here the state variables h, v and m are altitude, velocity, and mass respectively. T is the thrust magnitude. The final time t_f is free. We solve this problem by approximating the state trajectories as B-Splines with 3^{rd} order piece-wise polynomials having 2^{nd} order smoothness. Control T is approximated by using 1^{st} order GLOMAP functions. The algorithm starts with a uniform grid of 4 elements and performs 10 refinement iterations. The solution for this problem has a switching in control which is well captured by our algorithm. The states exhibit corners at the same time location as seen in Figure (50(a)).



(a) States h, v and m.



(b) Control T.



(c) Costates λ_h,λ_v, and λ_m.

Fig. 50. Results for moonlanding problem.

Our solution matches well with the one given in Ref. [63], where the optimal cost $m(t_f) = 0.3751$ agrees to 4 decimal places. The results are summarized in Table IV and shown in Figure (50).

4. Example 4: Maximum Radius Orbit Transfer

This problem has been discussed in Refs.[10, 32]. The objective is to find the control that maximizes the final orbital radius of a rocket starting form a given initial orbit. The state variables are the orbital radius r, the true anomaly θ, the radial component of velocity u, and the tangential component of velocity v. The control variable is the

Table IV. Results for moon-landing problem

Quantity of interest	Value
EqvI, EqvII	$\begin{bmatrix} 1.19\text{e-}004 & -1.69\text{e-}003 \\ 1.11\text{e-}004 & -8.15\text{e-}003 \\ 1.80\text{e-}004 & 3.67\text{e-}003 \end{bmatrix}$
$\|\mathcal{H}(t)\|_2$	1.10e-005
Optimal cost	0.375122

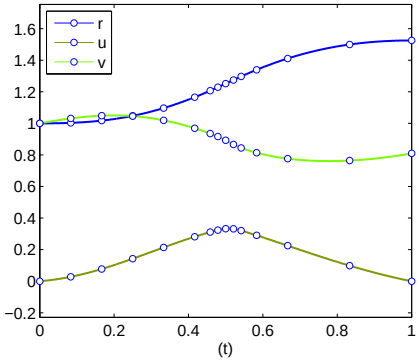
thrust steering angle β measured from the local horizontal. The transfer time t_f is fixed. The optimal control problem is,

$$\begin{aligned} \text{Minimize: } & J = -r(t_f) \\ \text{Subject to: } & \dot{r} = u \\ & \dot{\theta} = \frac{v}{r} \\ & \dot{u} = \frac{v^2}{r} - \frac{1}{r^2} + \frac{T}{m}\sin(\beta) \\ & \dot{v} = -\frac{uv}{r} + \frac{T}{m}\cos(\beta) \\ & \dot{m} = -0.0749 \end{aligned}$$

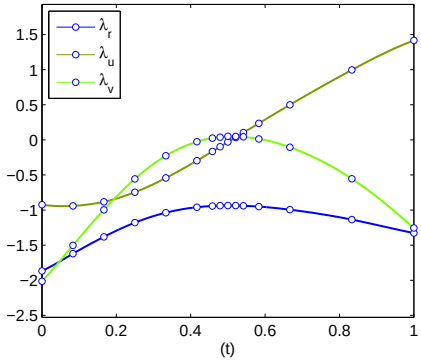
$$\begin{aligned} & r(0) = 1.1, \quad \theta(0) = 0, \quad u(0) = 0, \\ & v(0) = 1/\sqrt{1.1}, \quad m(0) = 1, \\ & u(t_f) = 0, \quad v(t_f) = \sqrt{1/r(t_f)}, \\ & 0 \le T \le 1.1 \end{aligned}$$

The thrust magnitude $T = 0.1405$ and $t_f = 3.32$.

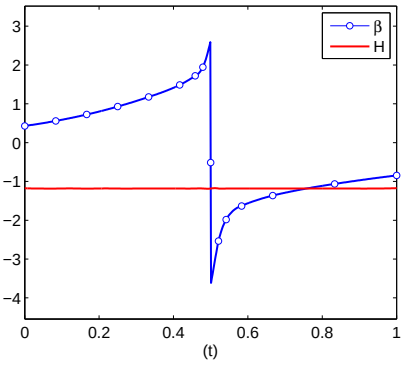
We solve this problem by approximating the state trajectories as B-Splines with 5^{th} order piece-wise polynomials having 4^{th} order smoothness. Control T is approximated by using 3^{rd} order B-Splines with 0^{th} order smoothness. The results are shown in Figure (51). Our solution matches well with the results given in Ref. [32]. The discontinuity in β is captured adaptively as the grid is refined. We start the refinement algorithm with 6 intervals and perform 10 iterations to obtain the present results. The optimal cost $r(t_f) = 1.525$ obtained by us matches with Ref. [32].



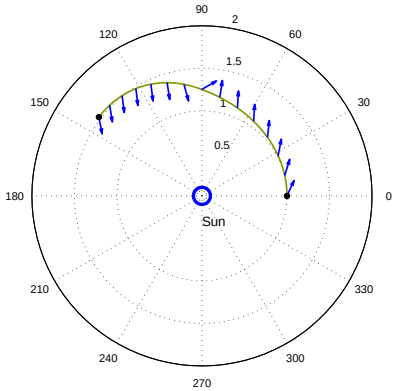
(a) States r, u, and v.



(b) Costates λ_r,λ_u and λ_v.



(c) Control β and Hamiltonian $\mathcal{H}$.



(d) Rocket trajectory in $(r-\theta)$ plane with thrust directions.

Fig. 51. Results for maximum radius orbit transfer problem.

The results for Example 4 are summarized in Table V. This example demonstrates the applicability of the present method for real-life applications.

Table V. Results for maximum range orbit transfer problem

Quantity of interest	Value
EqvI, EqvII	$\begin{bmatrix} 1.10\text{e-}002 & 8.29\text{e-}003 \\ -1.23\text{e-}004 & 5.99\text{e-}004 \\ 5.60\text{e-}003 & -7.79\text{e-}003 \\ 1.17\text{e-}002 & 8.95\text{e-}003 \\ -3.38\text{e-}003 & -9.36\text{e-}005 \end{bmatrix}$
$\|\dot{\mathcal{H}}(t)\|_2$	4.21e-002
Optimal cost	1.5251

E. Conclusions

This chapter proposed an h-adaptive algorithm for solving optimal control problems by using local approximating functions like B-Splines and PU based approximations. An a posteriori error estimation procedure was developed, and was used for adaptive grid refinement. Through a number of numerical examples, the efficiency of this approach was demonstrated.

CHAPTER X

TIME-SCALING METHOD FOR NON-SMOOTH PROBLEMS WITH MULTIPLE PHASES

Many real life optimal control problems have solutions with discontinuities and corners. Such problems typically arise when the control is constrained and appears linearly in both the state dynamics and the cost function, which happens frequently in robotics and aerospace applications. The discontinuous control may also be intrinsic to the problem formulation, for example in the case where the actuators are of "on/off" type. When the control is discontinuous, some of the states may possess corners where state derivatives are not continuous. Direct optimization based numerical methods are quite efficient for solving smooth optimal control problems, but perform poorly with the problems having non-smooth solutions. This is because the locations of discontinuities are not known before hand. In this paper, our focus is on developing a direct optimization based methodology to solve non-smooth optimal control problems.

In this chapter, a direct optimization algorithm is developed to solve problems with discontinuous control based on the least square method for optimal control (LSM_{oc}) [64]. LSM_{oc} is flexible with respect to the choice of approximating functions, and provides costate estimation and optimality verification results. To accommodate the discontinuities and corners, we divide the time domain into a number of subintervals. Each subinterval defines a control phase. Depending upon the problem in hand, a control type is assigned to each phase. For example, for a problem having bang-bang solution, the control type is constant at either its maximum or minimum value. To deal with the unknown switching times, we map the control phases on a computational domain with equal intervals. The state dynamics is appropriately scaled in

each interval. The scaled problem is then discretized using B-Splines and transcribed to an NLP using the LSM_{oc}. The NLP is solved and the solution is mapped back to the original time domain.

There are several advantages of using the present approach. First, being a direct method, it is robust with respect to the deviation of the initial guess from the true solution. Second, the NLP formulation is based on a weighted residual formulation, which results in a smaller problem size compared to collocation based methods. Further, using B-Splines, the state continuity conditions at switching times can be imposed by construction itself. This eliminates the need of extra knotting conditions or phase boundary conditions in the NLP. Finally, the LSM_{oc} allows us to obtain costate estimates from the Karush Kuhn Tucker multipliers of the NLP, which are crucial for optimality verification of the obtained solution. In this chapter, we consider an optimal control problem in Bolza form. The objective is to determine the state-control pair $\{\mathbf{X}(\tau) \in \mathbb{R}^n, \mathbf{U}(\tau) \in \mathbb{R}^m; \tau \in [0, \tau_f]\}$ and time instance τ_f, that minimize the cost,

$$J = \int_0^{\tau_f} L(\mathbf{X}(\tau), \mathbf{U}(\tau)) d\tau + \Psi(\mathbf{X}(0), \mathbf{X}(\tau_f), 0, \tau_f), \tag{10.1}$$

subject to the state dynamics,

$$\frac{d\mathbf{X}(}{d\tau}\tau) = F(\mathbf{X}(\tau), \mathbf{U}(\tau)), \tag{10.2}$$

end-point state equality constraints,

$$\mathbf{X}(0) = \mathbf{x}_0, \quad \psi(\mathbf{X}(0), \mathbf{X}(\tau_f), 0, \tau_f) = 0, \tag{10.3}$$

and control limits,

$$u^{min} \leq \mathbf{U}(\tau) \leq u^{max}, \tag{10.4}$$

were $L : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$, $\Psi : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$, $F : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $\psi : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^p$ are continuously differentiable with respect to their arguments. It is assumed that the optimal solution to the above problem exists and the constraint qualifications required to apply the first-order optimality conditions are met.

A. Time-Scaling Methodology

1. Control Specifications

At any given time instance τ, the control $\mathbf{U}(\tau)$ either lies on the constraints boundary $\{u^{min}, u^{max}\}$ or has its value in the interval (u^{min}, u^{max}). For some problems, the control may be "turned-off", which means that $\mathbf{U}(\tau)$ may be zero. Thus, we can define four distinct control types: (1) constant at zero, (2) constant at the minimum value, (3) constant at the maximum value and (4) varying in a bounded interval.

Assuming that the control maintains a particular type for a finite duration, we introduce the concept of a control phase. A control phase is defined as a finite interval of time over which the control is continuous and maintains its type. From the discussion above, there are four types of control phases, one for each control type.

A number of control phases can be joined in a sequence to represent the control trajectory. As stated earlier, the control is continuous over a particular phase, however, it can be discontinuous at the junction point of adjacent phases, termed as switching times. The switching times alow us to place the discontinuities in the control trajectory. Since the locations of discontinuities are not known, switching times are the unknowns of the problem.

Depending upon the problem in hand, the control type in each phase is assigned based on some preliminary analysis or an intelligent guess. For some problems for example, it is known that the optimal control is of "bang-bang" or "bang-zero-bang"

type. This information is used to decide the number of phases and the control type in each phase.

Let $\{\tau_j\}_{j=1}^{N}$ be the unknown switching times such that,

$$0 = \tau_0 \leq \tau_1 \leq ... \leq \tau_N = \tau_f, \tag{10.5}$$

with $\Omega_j := [\tau_{j-1}, \tau_j]$. Then the j^{th} phase of i^{th} control variable is defined as,

$$\mathbf{U}_{i,j}(\tau) = \mathbf{U}_i(\tau)|_{\tau \in \Omega_j}; \quad i = 1, .., m, \tag{10.6}$$

with the associated phase domain Ω_j and phase length $T_j = (\tau_j - \tau_{j-1})$. The control type in each phase is pre-assigned, so that $\mathbf{U}_{i,j}(\tau)$ takes one of the following form:

1. $\mathbf{U}_{i,j}(\tau) = 0$, denoted as "0",
2. $\mathbf{U}_{i,j}(\tau) = u_i^{min}$, denoted as "$-$",
3. $\mathbf{U}_{i,j}(\tau) = u_i^{max}$, denoted as "$+$",
4. $u_i^{min} \leq \mathbf{U}_{i,j}(\tau) \leq u_i^{max}$, denoted as "$\pm$" or "free",

for $i = 1, .., m$ and $j = 1, .., N$.

For a given N, the control structure is defined as a sequence of N control types in order. For example, $\mathbf{U}_i : (-, +, \pm, 0)$ represents the control structure (minimum,maximum,free,zero).

2. Control Sequencing and Modified Bolza Problem $\mathcal{B}^c$

The control variables of the original optimal control problem are assigned a specific structure. The controls are represented as a sequence of a fixed number of phases, and a control type is assigned for each phase. By doing so, the control value in some of the phases is fixed and thus is no longer an optimization variable. However, the

phase lengths defined by the switching times become an additional set of unknowns introduced into the problem. In this section, we first define a systematic procedure to incorporate a control structure into the problem. Subsequently, we transform the original optimal control problem with free final time and unknown switching times, to a problem on a fixed computational domain with switching times incorporated as parameters of optimization in the state dynamics via a scaling factor.

To incorporate a pre-defined control structure into problem $\mathcal{B}$, we replace the original control variables $\mathbf{U}(\tau)$ with a new set $\mathbf{U}^c(\tau)$ defined as,

$$\mathbf{U}^c(\tau) = C^1(\tau) + C^2(\tau)\mathbf{U}(\tau), \tag{10.7}$$

where $C^1(\tau), C^2(\tau) \in \mathbb{R}^m$ are introduced as two auxiliary input variables. The values of $C^1(\tau)$ and $C^2(\tau)$ in each phase determine the control type of $\mathbf{U}^c(\tau)$ in that particular phase. For example, the i^{th} control variable $\mathbf{U}^c{}_i(\tau)$ can be fixed to its maximum value in the j^{th} phase by selecting $C^1_{i,j}(\tau) = u_i^{max}$ and $C^2_{i,j}(\tau) = 0$. Similarly, we can enforce any control type in a given phase by selecting the values of $C^1(\tau)$ and $C^2(\tau)$ as per Table VI.

Table VI. Auxiliary inputs to implement a given control structure

$C^1_{i,j}(\tau)$	$C^2_{i,j}(\tau)$	$\mathbf{U}^c{}_{i,j}(\tau)$	Control type
0	0	0	0
u_i^{min}	0	u_i^{min}	$-$
u_i^{max}	0	u_i^{max}	$+$
0	1	free	$\pm$

The concept of auxiliary inputs and the working of Table VI can be better understood in Figure (52), where an example case of four control phases is shown. The control structure $\mathbf{U}^c{}_i(\tau) : (-, +, \pm, 0)$ is derived using (10.7) by selecting the values of $C^1(\tau)$ and $C^2(\tau)$ from Table VI. Next, we replace the control variable $\mathbf{U}(\tau)$ in

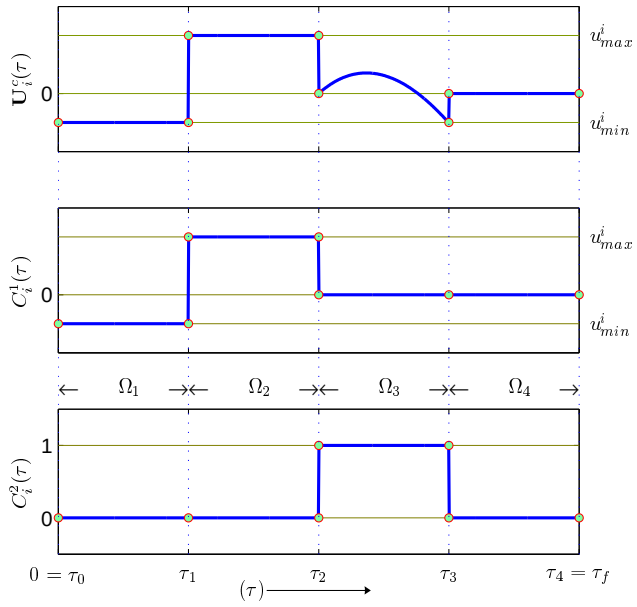


Fig. 52. Control sequencing, $\mathbf{U}^c{}_i : (-, +, \pm, 0)$, with corresponding auxiliary inputs.

problem $\mathcal{B}$ by $\mathbf{U}^c(\tau)$ and re-define the associated function expressions. Using Eqns. (10.1), (10.2) and (10.7) we write,

$$L(\mathbf{X}(\tau), \mathbf{U}(\tau)) = L(\mathbf{X}(\tau), \mathbf{U}_c(\tau)) = \mathcal{L}(\mathbf{X}(\tau), \mathbf{U}(\tau), C^1(\tau), C^2(\tau)), \tag{10.8}$$

$$F(\mathbf{X}(\tau), \mathbf{U}(\tau)) = F(\mathbf{X}(\tau), \mathbf{U}_c(\tau)) = \mathcal{F}(\mathbf{X}(\tau), \mathbf{U}(\tau), C^1(\tau), C^2(\tau)). \tag{10.9}$$

The modified optimal control problem $\mathcal{B}^c$ is to find $\{\mathbf{X}(\tau), \mathbf{U}(\tau); \tau \in [0, \tau_f]\}$ and $\{\tau_j\}_{j=1}^N$ which minimize the cost,

$$J = \int_0^{\tau_f} \mathcal{L}(\mathbf{X}(\tau), \mathbf{U}(\tau), C^1(\tau), C^2(\tau)) d\tau + \Psi(\mathbf{X}(0), \mathbf{X}(\tau_f), 0, \tau_f), \tag{10.10}$$

subject to,

$$\frac{d\mathbf{X}(}{d\tau}\tau) = \mathcal{F}(\mathbf{X}(\tau), \mathbf{U}(\tau), C^1(\tau), C^2(\tau)), \tag{10.11}$$

$$\mathbf{X}(0) = \mathbf{x}_0, \quad \psi(\mathbf{X}(0), \mathbf{X}(\tau_f), 0, \tau_f) = 0, \tag{10.12}$$

$$u^{min} \leq \mathbf{U}(\tau) \leq u^{max}, \tag{10.13}$$

where $C^1(\tau)$ and $C^2(\tau)$ are constant inputs.

3. Time Scaling and Mapping to Bolza Problem $\mathcal{B}^N$

To deal with the unknown switching times $\{\tau_j\}_{j=1}^N$ in problem $\mathcal{B}^c$, a time scaling technique is outlined where the time domain of each phase is mapped to a computational domain of fixed length. The state dynamics is appropriately scaled by a factor which contains the phase length as an unknown parameter.

Consider a computational domain $t \in [0, 1]$ and divide it into N number of equal subintervals $\Delta_j|_{j=1}^N := [\frac{(j-1)}{N}, \frac{(j)}{N}]$. For a given phase j with domain Ω_j and phase length $T_j = \tau_j - \tau_{j-1}$, define a mapping from Ω_j to Δ_j via the following scaling relationship,

$$\frac{\tau - \tau_{j-1}}{T_j} = \frac{t - \frac{(j-1)}{N}}{\frac{1}{N}}; \quad \tau \in \Omega_j, t \in \Delta_j. \tag{10.14}$$

The mapping process is depicted in Figure (53). By differentiating (10.14) with respect to t we get,

$$\left.\frac{d\tau}{dt}\right|_{t\in\Delta_j} = NT_j. \tag{10.15}$$

The derivative $\frac{d\tau}{dt}$ in (10.15) is constant over each phase, and can be written in a

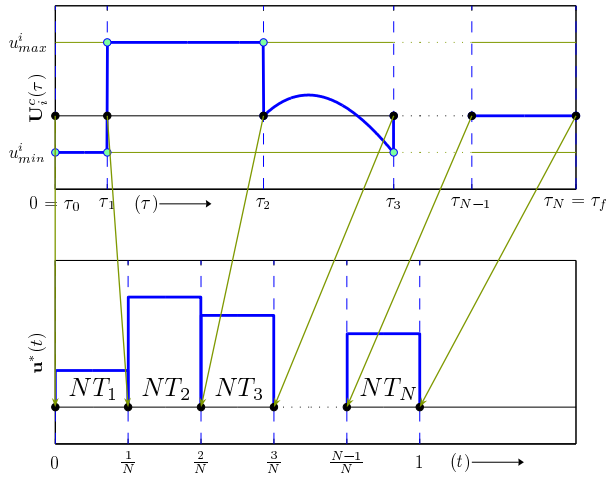


Fig. 53. Time-scaling and mapping of domain (τ) to domain (t).

trajectory form as following,

$$\frac{d\tau}{dt} = \sum_{j=1}^{N} NT_j \sqcap_j (t); \quad t \in [0,1]. \tag{10.16}$$

where $\sqcap_j(t)$ is a rectangular function defined as,

$$\sqcap_j(t) = \begin{cases} 1 & \text{if } t \in \Delta_j \\ 0 & \text{if } t \notin \Delta_j \end{cases}. \tag{10.17}$$

By integrating (10.16) and using $\tau(0) = 0$, we can represent τ as a function of t as following,

$$\tau(t) = \int_0^t \sum_{j=1}^{N} NT_j \sqcap_j (z) dz, \tag{10.18}$$

also,

$$\tau_f = \tau(1) = \int_0^1 \sum_{j=1}^{N} NT_j \sqcap_j (z) dz = \sum_{j=1}^{N} T_j. \tag{10.19}$$

Next, map the state, control and auxiliary input trajectories to domain t as,

$$\mathbf{x}(t) = \mathbf{X}(\tau(t)), \quad \mathbf{u}(t) = \mathbf{U}(\tau(t)) \tag{10.20}$$

$$\mathbf{c}^1(t) = C^1(\tau(t)) \quad \mathbf{c}^2(t) = C^2(\tau(t)). \tag{10.21}$$

Using (10.16), we introduce $(\mathbf{x}^*(t), \mathbf{u}^*(t))$ as an additional state-control pair satisfying,

$$\mathbf{x}^*(t) = \tau(t), \quad \dot{\mathbf{x}}^*(t) = \mathbf{u}^*(t) = \sum_{j=1}^{N} NT_j \sqcap_j (t). \tag{10.22}$$

For the state derivatives we have,

$$\frac{d\mathbf{X}(\tau)}{d\tau} = \frac{1}{\left(\frac{d\tau}{dt}\right)}\dot{\mathbf{x}}(t) = \frac{1}{\mathbf{u}^*(t)}\dot{\mathbf{x}}(t) \tag{10.23}$$

Using (10.2) and (10.23), the state dynamics can be written as,

$$\dot{\mathbf{x}}(t) = \mathbf{u}^*(t)\mathcal{F}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{c}^1(t), \mathbf{c}^2(t)). \tag{10.24}$$

Similarly we map all the function expressions in problem $\mathcal{B}^N$ to the domain t and define the mapped optimal control problem $\mathcal{B}^N$ as following. Find $\{\mathbf{x}(t), \mathbf{x}^*(t), \mathbf{u}(t), \mathbf{u}^*(t)\}$ that minimize,

$$J = \int_0^1 \mathbf{u}^*(t)\mathcal{L}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{c}^1(t), \mathbf{c}^2(t))dt + \Psi(\mathbf{x}(1), \mathbf{x}^*(1)), \tag{10.25}$$

subject to,

$$\begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{x}}^*(t) \end{bmatrix} = \begin{bmatrix} \mathbf{u}^*(t)\mathcal{F}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{c}^1(t), \mathbf{c}^2(t)) \\ \mathbf{u}^*(t) \end{bmatrix}, \tag{10.26}$$

$$\mathbf{x}(0) = \mathbf{x}_0, \quad \mathbf{x}^*(0) = 0, \quad \psi(\mathbf{x}(1), \mathbf{x}^*(1)) = 0, \tag{10.27}$$

$$u_{min} \leq \mathbf{u}(t) \leq u_{max}. \tag{10.28}$$

B. Direct Transcription Process

A direct method to solve an optimal control problem typically consists of two steps. First, state and control trajectories are approximated as a linear combinations of *a priori* selected basis functions with the corresponding coefficients as unknowns. Second, the cost function and the state dynamics is appropriately transformed into a set of algebraic equations in terms of the unknown coefficients. This transcribes an optimal control problem to a nonlinear programming problem.

To select appropriate approximating functions for $\{\mathbf{x}(t), \mathbf{x}^*(t), \mathbf{u}(t), \mathbf{u}^*(t)\}$, we make the following observations:

1. $\mathbf{u}^*(t)$ is a piecewise constant trajectory with only C^0 continuity at switching points $t_j := \frac{j}{N}, j = 1..N$.

2. $\dot{\mathbf{x}}(t) = \mathbf{u}^*(t)$ is C^0 continuous at switching times which implies that $\mathbf{x}(t)$ is only C^1 continuous at $\{t_j\}_{j=1}^N$.

3. $\mathbf{x}^*(t)$ is piecewise linear and C^1 continuous at $\{t_j\}_{j=1}^N$.

Therefore the approximation scheme used to parameterize the state and control trajectories must only be C^1 continuous at the switching times and should have higher smoothness order in between. This makes the use of B-Splines a natural choice to parameterize $\{\mathbf{x}(t), \mathbf{x}^*(t), \mathbf{u}(t)\}$, as the B-Splines can satisfy the continuity requirements by construction.

A B-Spline function defined on the interval $[0, 1]$ is composed of segments of polynomials that are stitched at predefined break points, satisfying a given degree of smoothness. The number of continuous derivatives across the breakpoints defines the order of smoothness. An order of smoothness s_i at a breakpoint t_i implies that the curve is C^{s_i-1} times continuously differentiable at t_i. Given the number of subintervals

(N), the order of each polynomial segment (r) and the order of smoothness (s) at the breakpoints, a B-Spline curve $y(t)$ is represented in the basis form as,

$$y(t) = \sum_{k=1}^{N_c} \alpha_k B^{k,r}(t),$$

where α_k are the free parameters and $N_c = N(r-s)+s$ is the number of free parameters or the degrees of freedom of $y(t)$. In the present scenario, the break points are $\{\frac{i}{N}\}_{i=0}^{N}$ and the smoothness order is 1, *i.e.* $s_i|_{i=0}^{N} = 1$. For some polynomial orders $r_{\mathbf{x}}$ and $r_{\mathbf{u}}$, we approximate $\{\mathbf{x}(t), \mathbf{u}(t)\}$ as,

$$\mathbf{x}(t) \approx \widehat{\mathbf{x}}(t) = \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k B^{k,r_{\mathbf{x}}}(t), \tag{10.29}$$

$$\mathbf{u}(t) \approx \widehat{\mathbf{u}}(t) = \sum_{k=1}^{N_{\mathbf{u}}} \beta_k B^{k,r_{\mathbf{u}}}(t), \tag{10.30}$$

where $\alpha_k \in \mathbb{R}^n, \beta_k \in \mathbb{R}^m$, $N_{\mathbf{x}} = N(r_{\mathbf{x}}-1)+1$ and $N_{\mathbf{u}} = N(r_{\mathbf{u}}-1)+1$. Since $\mathbf{x}^*(t)$ is piecewise linear,

$$\mathbf{x}^*(t) \approx \widehat{\mathbf{x}}^*(t) = \sum_{k=1}^{N+1} \alpha_k^* B^{k,2}(t), \qquad \alpha_k^* \in \mathbb{R}. \tag{10.31}$$

Using Eqns. (10.29), (10.30), (10.31) and (10.22), we have $\alpha_k|_{k=1}^{N_{\mathbf{x}}}$, $\beta_k|_{k=1}^{N_{\mathbf{u}}}$, $\alpha_k^*|_{k=1}^{N+1}$ and $T_k|_{k=1}^{N}$ as unknowns to be determined.

1. Least Square Method for Optimal Control

The next step in the transcription of problem $\mathcal{B}^N$ to a nonlinear programming problem is to transform the integral cost and the state dynamics into a set of algebraic cost and constraints. We use least square method for direct optimal control as described Chapter V. In this method, the state dynamics is approximated as a weighted integral formulation derived from the least square method to solve initial value problems. Using Eqns. (10.29), (10.30) and (10.31), we denote the approximate state dynamics

as,

$$\widehat{\mathcal{F}}(\alpha_k, \beta_k, T_k, B^{k,r_{\mathbf{x}}}, B^{k,r_{\mathbf{u}}}) = \sum_{k=1}^{N} NT_k \sqcap_k \mathcal{F}(\widehat{\mathbf{x}}, \widehat{\mathbf{u}}, \mathbf{c}^1, \mathbf{c}^2).$$

Using similar notation for all other functionals the transcribed problem is to determine $\alpha_k|_{k=1}^{N_{\mathbf{x}}}$, $\beta_k|_{k=1}^{N_{\mathbf{u}}}$, $\alpha_k^*|_{k=1}^{N+1}$, $T_k|_{k=1}^{N}$, $\nu \in \mathbb{R}^n$ and $\nu^* \in \mathbb{R}$ that minimize the cost,

$$\widehat{J} = \int_0^1 \sum_{i=1}^{N} NT_i \sqcap_i \widehat{\mathcal{L}}(\alpha_k, \beta_k, B^{k,r_{\mathbf{x}}}, B^{k,r_{\mathbf{u}}})dt + \widehat{\Psi}(\alpha_k, \alpha_k^*, B^{k,r_{\mathbf{x}}}(1), B^{k,2}(1)), \quad (10.32)$$

subject to the constraints,

$$\int_0^1 (\widehat{\mathcal{F}}_{\mathbf{x}}^T B^{j,r_{\mathbf{x}}} - \mathbf{I}\dot{B}^{j,r_{\mathbf{x}}})(\widehat{\mathcal{F}} - \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \dot{B}^{k,r_{\mathbf{x}}})dt + \frac{1}{2}\nu B^{j,r_{\mathbf{x}}}(0) = 0 \quad (10.33)$$

$$\int_0^1 \dot{B}^{p,2}(\sum_{k=1}^{N} NT_k \sqcap_k - \sum_{k=1}^{N+1} \alpha_k^* \dot{B}^{k,2})dt - \frac{1}{2}\nu^* B^{p,2}(0) = 0 \quad (10.34)$$

$$\sum_{k=1}^{N_{\mathbf{x}}} \alpha_k B^{k,r_{\mathbf{x}}}(0) - \mathbf{x}_0 = 0, \quad (10.35)$$

$$\sum_{k=1}^{N+1} \alpha_k^* B^{k,2}(0) = 0, \quad (10.36)$$

$$\widehat{\psi}(\alpha_k, B^{k,r_{\mathbf{x}}}(1), \alpha_k^*, B^{k,2}(1)) = 0 \quad (10.37)$$

where $j = 1, .., N_{\mathbf{x}}$ and $p = 1, .., N + 1$. The subscript argument denotes the partial derivative, i.e. $\mathcal{F}_{\mathbf{x}} = \frac{\partial \mathcal{F}}{\partial \mathbf{x}}$. The integral expressions in Eqns. (10.33) and (10.37) need to be evaluated numerically by using some quadrature scheme, which will complete the transcription of problem $\mathcal{B}^N$ to a nonlinear programming problem.

2. Equivalence Conditions and Costate Estimates

The solution of problem $\mathcal{M}_\phi$ is based on satisfying a set of first order optimality conditions for a nonlinear program, also known as Karush-Kuhn-Tucker (KKT) conditions. To derive the KKT conditions, the augmented cost function for Problem $\mathcal{M}_\phi$ is formed by adjoining the original cost function with the constraint equations. So

that,

$$
\begin{aligned}
J' = & \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j^T \left[\int_0^1 (\widehat{\mathcal{F}}_{\mathbf{x}}^T B^{j,r_{\mathbf{x}}} - \mathbf{I}\dot{B}^{j,r_{\mathbf{x}}})(\widehat{\mathcal{F}} - \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \dot{B}^{k,r_{\mathbf{x}}}) dt + \frac{1}{2}\nu B^{j,r_{\mathbf{x}}}(0) \right] \\
& + \sum_{p=1}^{N+1} \gamma_p^{*T} \left[\int_0^1 \dot{B}^{p,2} (\sum_{k=1}^{N} NT_k \sqcap_k - \sum_{k=1}^{N+1} \alpha_k^* \dot{B}^{k,2}) dt - \frac{1}{2}\nu^* B^{p,2}(0) \right] \\
& + \mu^T (\sum_{k=1}^{N_{\mathbf{x}}} \alpha_k B^{k,r_{\mathbf{x}}}(0) - \mathbf{x}_0) + \mu^{*T} \sum_{k=1}^{N+1} \alpha_k^* B^{k,2}(0) + \widehat{\Psi} + \eta^T \widehat{\psi}, \qquad (10.38)
\end{aligned}
$$

where $\gamma_j \in \mathbb{R}^n$, $\mu \in \mathbb{R}^n$, $\eta \in \mathbb{R}^p$, $\gamma_p^* \in \mathbb{R}$ and $\mu* \in \mathbb{R}$ are the KKT multipliers associated with the constraints (10.33)-(10.37). The KKT first-order necessary conditions are then obtained by setting the derivatives of J' with respect to the unknowns $\{\alpha_i, \beta_i, \varsigma_i, \nu, \gamma_i, \mu, \eta, \zeta_i, \tau_f\}$ equal to zero. By comparing the KKT conditions with the discretized EL equations, we can derive a set of equivalence conditions under which the costates can be estimated from the KKT multipliers. The detailed proof can be found in Ref. [64], and is skipped here for brevity. The equivalence conditions for the LSM_{oc} are,

$$
(\sum_{j=1}^{N_{\mathbf{x}}} \gamma_j \widehat{\mathcal{F}}_{\mathbf{x}} B^{j,r_{\mathbf{x}}} - \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j \dot{B}^{j,r_{\mathbf{x}}}) \Big|_{t=0} = -\mu, \qquad (10.39)
$$

$$
(\sum_{j=1}^{N_{\mathbf{x}}} \gamma_j \widehat{\mathcal{F}}_{\mathbf{x}} B^{j,r_{\mathbf{x}}} - \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j \dot{B}^{j,r_{\mathbf{x}}}) \Big|_{t=1} = \widehat{\Psi}_{\mathbf{x}(1)} + \widehat{\psi}_{\mathbf{x}(1)}^T \eta \qquad (10.40)
$$

$$
(\sum_{p=1}^{N+1} \gamma_p^* \dot{B}^{p,2}) \Big|_{t=0} = \mu^*, \qquad (10.41)
$$

$$
(\sum_{p=1}^{N+1} \gamma_p^* \dot{B}^{p,2}) \Big|_{t=1} = -(\widehat{\Psi}_{\mathbf{x}^*(1)} + \widehat{\psi}_{\mathbf{x}^*(1)}^T \eta). \qquad (10.42)
$$

When the equivalence conditions (10.39) and (10.40) are satisfied, the estimates of the costates $\widehat{\lambda}(t)$ and the lagrange multipliers $(\widehat{\xi}(t), \upsilon, \kappa)$ can be estimated from the

KKT multipliers as following,

$$\widehat{\lambda}(t) = \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j \widehat{\mathcal{F}}_{\mathbf{x}} B^{j,r_{\mathbf{x}}}(t) - \sum_{j=1}^{N_{\mathbf{x}}} \gamma_j \dot{B}^{j,r_{\mathbf{x}}}(t), \tag{10.43}$$

$$\widehat{\lambda}^*(t) = \sum_{p=1}^{N+1} \gamma_p^* \dot{B}^{p,2}. \tag{10.44}$$

Form the estimates of the costates, the Hamiltonian is evaluated as,

$$\widehat{\mathcal{H}}(t) = \sum_{k=1}^{N} NT_k \sqcap_k (\widehat{\mathcal{L}} + \widehat{\lambda}^T \widehat{\mathcal{F}} + \widehat{\lambda}^*). \tag{10.45}$$

3. Numerical Integration

The numerical solution of the problem defined in Section 1 requires the integral expressions in (10.32)-(10.34) be evaluated numerically. This is accomplished by using a numerical quadrature scheme. In this section, we describe the quadrature scheme and derive the nonlinear programming problem to be solved.

In the computational domain (t), we define N_q number of LGL points $\{b_{ij}\}_{j=1}^{N_q}$ with corresponding quadrature weights $\{w_{ij}\}$ suitably mapped over each domain $\{\Delta_i\}_{i=1}^{N}$(see Figure (54)). Then, the integral of a function $f(t)$ over interval Δ_i can be approximated as,

$$\int_{\Delta_i} f(t)dt \approx \sum_{j=0}^{N_q} w_{ij} f(b_{ij}). \tag{10.46}$$

Next, we formulate a nonlinear programming problem $\mathcal{B}_\phi^N$ based on Section 1 and using (10.46).

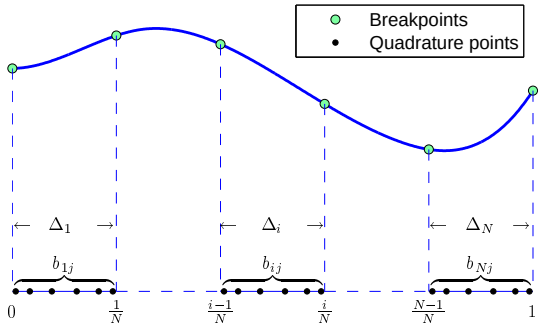


Fig. 54. The arrangement of breakpoints and quadrature points in domain $t \in [0,1]$. $\{t_i\}_{i=0}^{N}$ are the breakpoints. $\{b_{ij}\}_{j=1}^{N_q}$ are the quadrature points for domain $\Delta_i = [t_{i-1}, t_i]$.

4. Nonlinear Programming Problem $\mathcal{B}_\phi^N$

The problem $\mathcal{B}_\phi^N$ is to determine $\alpha_k|_{k=1}^{N_{\mathbf{x}}}$, $\beta_k|_{k=1}^{N_{\mathbf{u}}}$, $\alpha_k^*|_{k=1}^{N+1}$, $T_k|_{k=1}^{N}$, ν and ν^* that minimize,

$$J = \sum_{i=1}^{N}\sum_{j=1}^{N_q} w_{ij} N T_i \sqcap_i (b_{ij}) \widehat{\mathcal{L}}(\alpha_k, \beta_k, B_{ij}^{k,r_{\mathbf{x}}}, B_{ij}^{k,r_{\mathbf{u}}}) + \widehat{\Psi}(\alpha_k, \alpha_k^*, B^{k,r_{\mathbf{x}}}(1), B^{k,2}(1)),$$

subject to the constraints,

$$\sum_{i=1}^{N}\sum_{j=1}^{N_q} w_{ij}[(\widehat{\mathcal{F}}_{\mathbf{x}}^T B^{l,r_{\mathbf{x}}})_{ij} - \mathbf{I}\dot{B}_{ij}^{l,r_{\mathbf{x}}}][\widehat{\mathcal{F}}_{ij} - \sum_{k=1}^{N_{\mathbf{x}}} \alpha_k \dot{B}_{ij}^{k,r_{\mathbf{x}}}] + \frac{1}{2}\nu B_0^{l,r_{\mathbf{x}}} = 0, \tag{10.47}$$

$$\sum_{i=1}^{N}\sum_{j=1}^{N_q} w_{ij}\dot{B}_{ij}^{p,2}[\frac{T_i}{N}\sqcap_i (b_{ij}) - \sum_{k=1}^{N+1} \alpha_k^* \dot{B}_{ij}^{k,2}]dt - \frac{1}{2}\nu^* B^{p,2}(0) = 0 \tag{10.48}$$

$$\sum_{k=1}^{N_{\mathbf{x}}} \alpha_k B_0^{k,r_{\mathbf{x}}} - \mathbf{x}_0 = 0, \tag{10.49}$$

$$\sum_{k=1}^{N+1} \alpha_k^* B_0^{k,2} = 0, \tag{10.50}$$

$$\widehat{\psi} = 0, \tag{10.51}$$

where $l = 1, 2, .., N_{\mathbf{x}}$ and $p = 1, .., N + 1$. Here $(\cdot)_{ij}$ denotes the evaluation of the underlying expression at time b_{ij}. Similarly, $B_0^{k,r}, B_1^{k,r}$ denote $B^{k,r}(0), B^{k,r}(1)$ respectively.

5. Numerical Solution

The nonlinear programming problem defined in the previous section can be solved using any of the available optimization software like 'fmincon' in MATLAB, SNOPT [12], or NPSOL [13]. For the example problems solved in this paper, we use SNOPT as the optimizer in MATLAB environment. A direct transcription MATLAB-toolbox named OPTRAGEN-3 has been developed. It takes user-friendly inputs and generates all the required numerical setup for solving the NLP.

C. Application Examples

In this section, we solve real-life problems using the methodology presented in the previous sections. We select three problems, one each from the following fields: aerospace, robotics and motion planning. The results are compared with the solutions found in the literature.

1. Example 1: Orbit Rendezvous Problem

In this example problem, a spacecraft is to rendezvous an asteroid in a fixed time. The spacecraft is initially in a circular orbit, and in a given time it has to match the position and velocity of a target asteroid traveling in another circular orbit. This problem has been solved by Bai.Turner.2009 et. al. in Ref [62], and we use the same problem setup here. The spacecraft dynamics is modeled as a planar motion of a point mass acted upon by the gravitational force from the Sun. The spacecraft has

a mass m, and its position is defined as a solar-centric polar coordinates (r, θ). r is the distance of the spacecraft from the Sun and θ is the phase angle with respect to some inertial axis. The radial and tangential velocities are denoted by u and v respectively. The angle between the thrust direction and the local tangent is β. The dynamic equations for the spacecraft are,

$$\dot{r} = u, \tag{10.52}$$

$$\dot{\theta} = \frac{v}{r} \tag{10.53}$$

$$\dot{u} = \frac{v^2}{r} - \frac{\mu}{r^2} + \frac{T}{m}\sin(\beta), \tag{10.54}$$

$$\dot{v} = -\frac{uv}{r} + \frac{T}{m}\cos(\beta) \tag{10.55}$$

$$\dot{m} = -\frac{T}{g_0 I_{sp}}. \tag{10.56}$$

The thrust magnitude is bounded as,

$$0 \leq T \leq T_{max}. \tag{10.57}$$

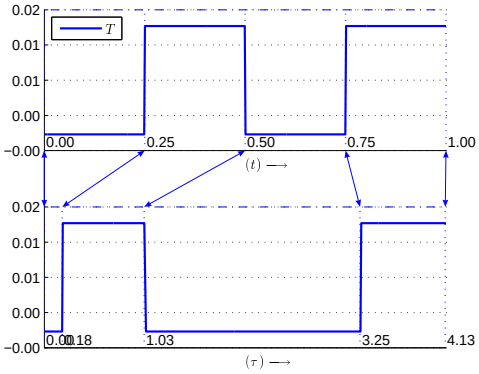
The problem is non-dimensionalized in distance by $1AU = 1.495978706910000 \times 10^{11}$m, in time by $1TU = 5.022642890912782 \times 10^{6}$sec and in mass by the initial spacecraft mass of 1500kg. The scaled values of relevant parameters are,

$$T_{max} = 0.01517685201253, \quad g_0 I_{sp} = 0.98775. \tag{10.58}$$

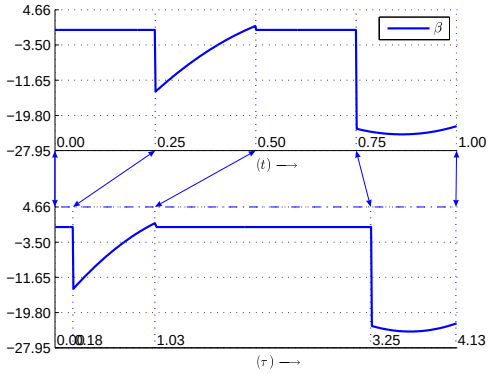
The optimal control problem is to find the history of T and β which drive the spacecraft form its initial state to the final, while maximizing the final mass of the spacecraft. The transfer time if $t_f = 4.1285$. The initial and final conditions are specified as,

r(0)	1	r(t_f)	1.05242919219003
$\theta(0)$	0	$\theta(t_f)$	3.99191781862267
u(0)	0	u(t_f)	0
v(0)	1	v(t_f)	0.97477314754443
m(0)	1	m(t_f)	maximum

The thrust magnitude T appears linearly in the state dynamics. Therefore, the optimal thrust profile is of "bang-bang" type. To solve this problem using the time-scaling technique, we divide the control profile into four phases having the thrust sequence $T : (-, +, -, +)$. The thrust direction β is free in all phases. Note that if we choose more than four phases, the optimization process will automatically "collapse" the extra phases within some numerical tolerance. The states are approximated as 6^{th} degree polynomials while thrust direction β is assumed to be a quadratic in each phase. 40 LGL nodes are used for numerical integration in each phase. The control histories are shown in Figure (55) and state trajectories in Figure (56).

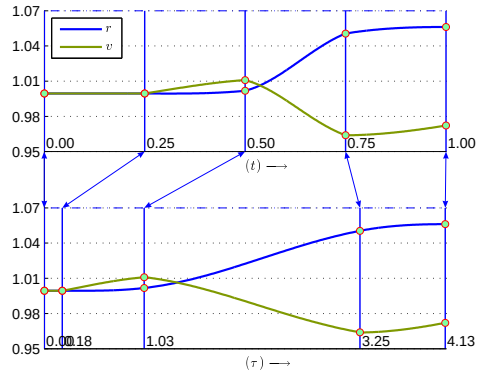


(a) Thrust magnitude.

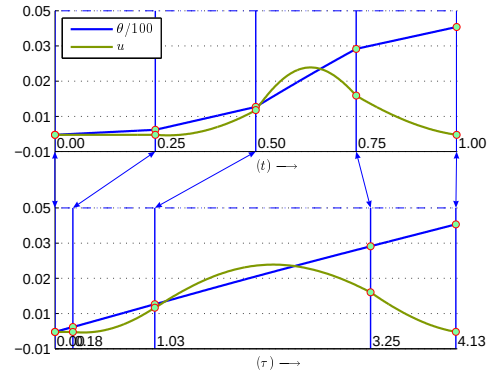


(b) Thrust direction in degrees.

Fig. 55. Optimal control histories for orbit rendezvous problem.



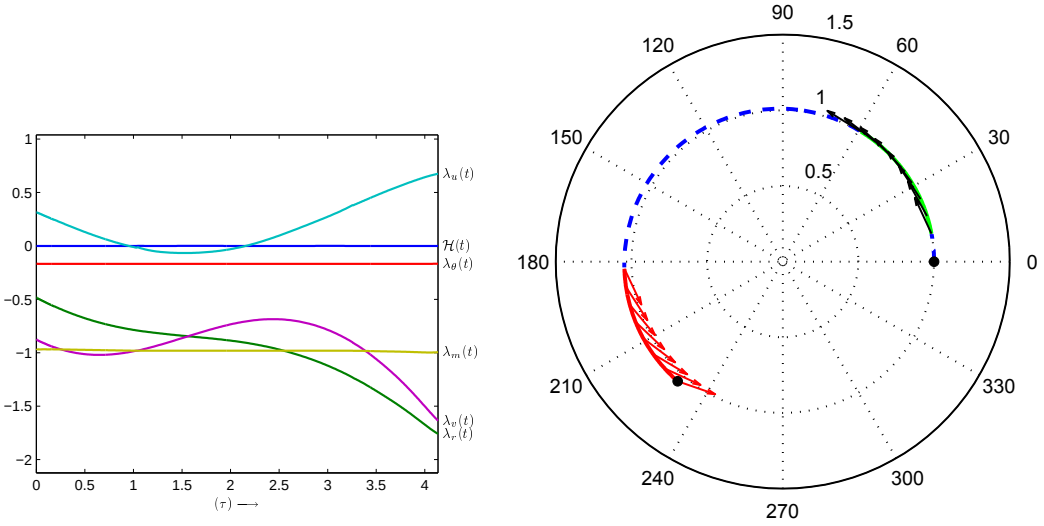
(a) State trajectories r and v.



(b) State trajectories θ and u.

Fig. 56. State trajectories of orbit rendezvous problem.

The spacecraft trajectory in $(r-\theta)$ plane is shown in Figure (57(b)). Figure (57(a)) shows the costates. The solution is obtained in t domain and is mapped to the original problem domain τ. The results are shown in both t and τ domain.



(a) (b)

Fig. 57. (a) Costates and Hamiltonian for rendezvous problem(b) Spacecraft trajectory.

Comparing the results with Ref. [62], we find that the switching times $(\tau_1, \tau_2, \tau_3, \tau_4)$ and the optimal cost $m(t_f)$ in our solution match with the reported values to the 4^{th} decimal place. Considering that the solution in Ref. [62] has been obtained by using an indirect approach, the present results are quite promising. We summarize the results in Table VII.

Table VII. Input-output data for rendezvous problem

Input Data	
No. of phases	4
Switching structure	$T : (-, +, -, +)$
	$\beta : (\pm, \pm, \pm, \pm)$
Polynomial order	$States:\ 6, \quad \beta:\ 3$
Results	
Switching times	$\begin{bmatrix} 0.184408 \\ 1.02836 \\ 3.25066 \\ 4.12865 \end{bmatrix}$
EqvI, EqvII	$\begin{bmatrix} 2.49\text{e-}003 & 9.07\text{e-}003 \\ 8.62\text{e-}002 & 8.62\text{e-}002 \\ -1.59\text{e-}003 & -3.77\text{e-}003 \\ 4.52\text{e-}003 & 8.35\text{e-}003 \\ 4.98\text{e-}003 & 5.14\text{e-}003 \end{bmatrix}$
$\|\mathcal{H}(t)\|_2$	1.65e-008
Optimal cost	-0.973545

2. Example 2: Caltech MVWT Vehicle Trajectory Optimization

The Caltech Multi-Vehicle Wireless Testbed (MVWT) is a platform for testing control methodologies for multiple vehicle coordination and formation stabilization Ref. [65]. The testbed consists of eight mobile vehicles. The MVWT vehicle rests on three low-friction, omni-directional casters and is powered by two high-performance ducted fans. Each fan is capable of producing up to 4.5 N of continuous thrust. The vehicle is underactuated and exhibits nonlinear second-order dynamics.

We solve a minimum-time trajectory optimization problem for the MTWV vehicle using the prescribed time-scaling technique. We assume that only the on/off type of control is available for the fans, and the motor transients can be ignored to approximate a "bang-bang" type of control structure. In this setting, the control structure is imposed by the problem definition itself and is not an outcome of applying the optimality conditions. In the (x-y) plane, the vehicle is initially at rest and is aligned with the x-axis. The problem is to find the optimal thrusting sequence of two fans so as to align the vehicle with y-axis in minimum time. The vehicle dynamics and associated data is taken from Ref. [65].

The problem is to minimize,

$$J = t_f + W[1 - v(t_f)]^2, \tag{10.59}$$

subject to the state dynamics,

$$\dot{x} = u, \qquad \dot{u} = -\frac{\eta}{m}u + \frac{(F_l + F_r)}{m}\cos(\theta), \tag{10.60}$$

$$\dot{y} = v, \qquad \dot{v} = -\frac{\eta}{m}v + \frac{(F_l + F_r)}{m}\sin(\theta), \tag{10.61}$$

$$\dot{\theta} = \omega, \qquad \dot{\omega} = -\frac{\psi}{J}\omega + \frac{(F_r - F_l)}{J}d, \tag{10.62}$$

where $W = 100$, rotational inertia $J = 0.050$ kg-m^2, the coefficient of viscous friction

$\eta = 5.5$ kg/s and the coefficient of rotational friction $\psi = 0.084$ kg m^2/s. The initial and final conditions are specified as,

$x(0)$	0	$x(t_f)$	0
$y(0)$	0	$y(t_f)$	–
$\theta(0)$	0	$\theta(t_f)$	$\pi/2$
$u(0)$	0	$u(t_f)$	0
$v(0)$	0	$v(t_f)$	–
$\omega(0)$	0	$\omega(t_f)$	0

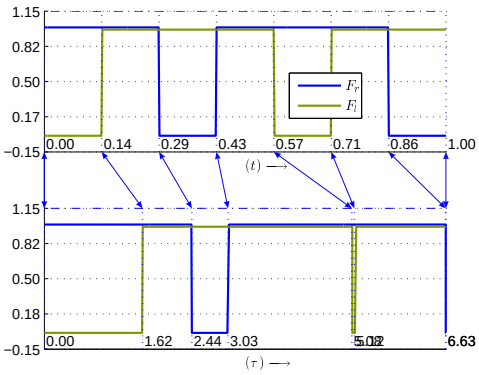
.

The thrust inputs are on/off type with thrust magnitude of 1N,

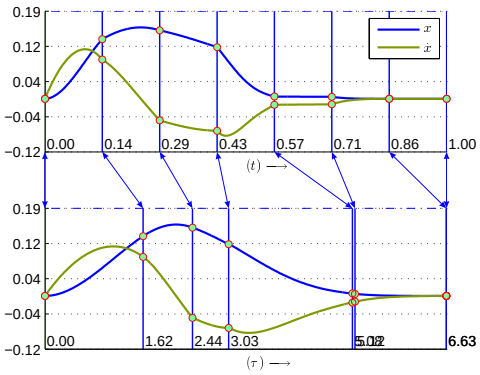
$$F_l, F_r \in \{0, 1 \text{ N}\} \tag{10.63}$$

Two control inputs makes this problem more challenging as all possible control combinations need to considered for defining a control sequence. To get an idea of the optimal switching structure, we first solve the problem using a standard direct method, like the least square method described in Ref. [64], with a coarse approximation. The obtained solution is not very accurate, yet provides insight into the underlying control structure. The regions where the control structure is ambiguous, we take all possible combinations. As we shall see in the results, any redundant control phases are eliminated in the optimization process. For the current problem, we start with 7 control phases and approximate the states as 5^{th} order polynomials. The problem is discretized using 40 LGL nodes in each phase.

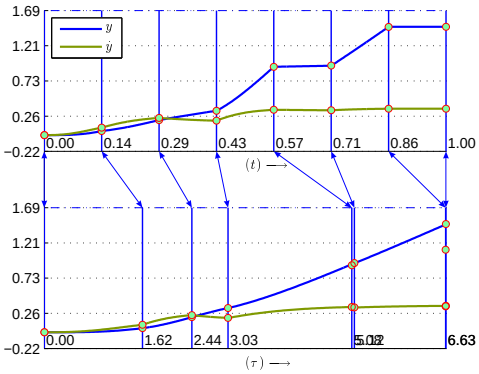
The results are shown in Figures (58) and (59). We see that the 7th control phase is redundant and gets "collapsed" in the final solution.



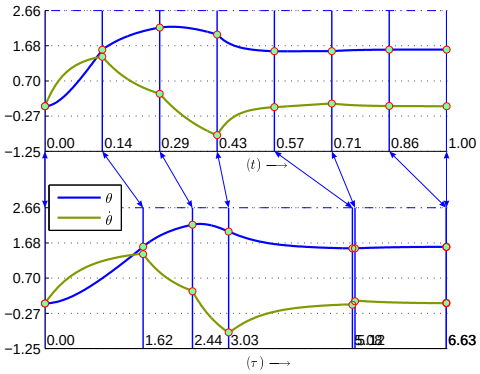
(a) Control trajectories F_l and F_r.



(b) State trajectories x and u.

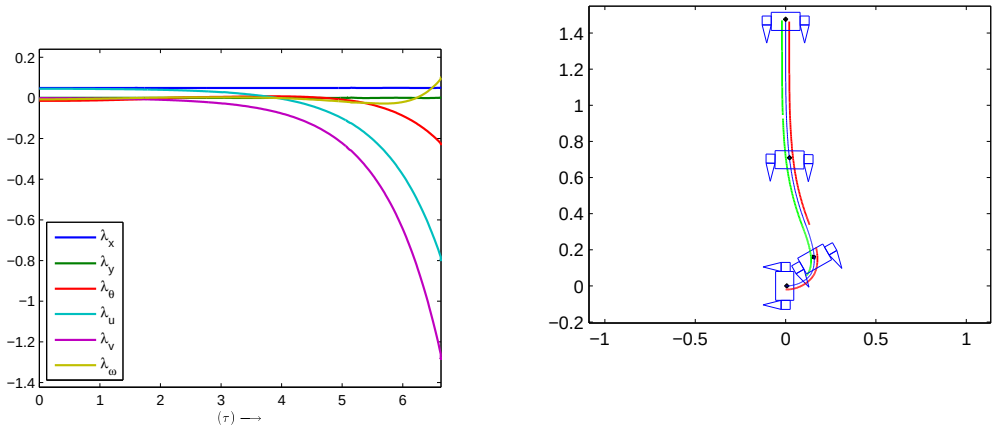


(c) State trajectories y and v.



(d) State trajectories θ and ω.

Fig. 58. Control and state trajectories of MVWT vehicle.



(a) (b)

Fig. 59. (a) Costates for the MVWT vehicle problem. (b) Path of MVWT vehicle in (x-y) plane. The central line is the robot path. The solid lines on its right and left side depict the "on" state of the right and left fan respectively.

Since there are no results available in the literature for this problem, the metrics of optimality are the equivalence conditions ($\approx \mathcal{O}^{-4}$) and the $\mathcal{L}_2$-norm of the Hamiltonian ($\approx \mathcal{O}^{-4}$). The values are summarized in Table VIII.

Table VIII. Input-output data for MVWT vehicle problem

Input Data	
No. of phases	7
Switching structure	$F_r : (+,+,-,+,+,+,-)$
	$F_l : (-,+,+,+,-,+,+)$
Polynomial order	$States: \ 5 \quad Controls : fixed$
Results	
Switching times	$\begin{bmatrix} 1.62076 \\ 2.43693 \\ 3.0336 \\ 5.07631 \\ 5.11894 \\ 6.6259 \\ 6.62839 \end{bmatrix}$
EqvI, EqvII	$\begin{bmatrix} 3.20\text{e-}005 & -1.12\text{e-}004 \\ -8.84\text{e-}006 & -2.29\text{e-}007 \\ 1.18\text{e-}004 & 5.22\text{e-}004 \\ -9.61\text{e-}005 & 1.83\text{e-}003 \\ 7.78\text{e-}005 & 2.94\text{e-}003 \\ 9.48\text{e-}005 & -2.32\text{e-}004 \end{bmatrix}$
$\|\mathcal{H}(t)\|_2$	1.13e-006
Optimal cost	0.478894

3. Example 3: Assembly Robot Motion Planning

An industrial robot performing assembly tasks typically exhibits point to point motion. Time optimality of robot motion is crucial for productivity. Therefore, there has been a considerable interest in the literature for studying and solving time optimal motions of assembly robots.

In the present example, we use the time-scaling method to compute the time optimal trajectories of a horizontal two-link robot. The numerical data is of IBM 7535 B 04 robot taken from Ref. [66]. The angular rotation of the inner link is θ and ϕ is the rotation angle of the outer link. The rotation angles θ, ϕ and the corresponding angular velocities define the robot states as,

$$x_1 = \theta, \quad x_2 = \dot{\theta}, \quad x_3 = \phi, \quad x_4 = \dot{\phi}. \tag{10.64}$$

The links are driven by torques M_θ and M_ϕ, which serve as the control variables,

$$u_1 = M_\theta, \quad u_2 = M_\phi. \tag{10.65}$$

The controls are bounded as,

$$-25 \text{ Nm} \leq u_1 \leq 25 \text{ Nm}, \quad -9 \text{ Nm} \leq u_2 \leq 9 \text{ Nm}. \tag{10.66}$$

The nonlinear state equations are,

$$\dot{x_1} = x_2, \tag{10.67}$$

$$\dot{x_2} = \frac{\begin{array}{c}[J_7\{u_1 - u_2 + J_6(x_2 + x_4)^2 \sin(x_3)\} \\ - J_6\{u_2 - J_6 x_2^2 \sin(x_3)\} \cos(x_3)]\end{array}}{[J_7 J_5 - J_6^2 \cos^2(x_3)]}, \tag{10.68}$$

$$\dot{x_3} = x_4, \tag{10.69}$$

$$\dot{x_4} = \frac{\begin{array}{c}[(J_5 + J_6\cos(x_3))\{u_2 - J_6 x_2^2 \sin(x_3)\} - \sin(x_3) \\ .(J_7 + J_6\cos(x_3))\{u_1 - u_2 + J_6(x_2 + x_4)^2\}]\end{array}}{[J_7 J_5 - J_6^2\cos^2(x_3)]}. \tag{10.70}$$

where,

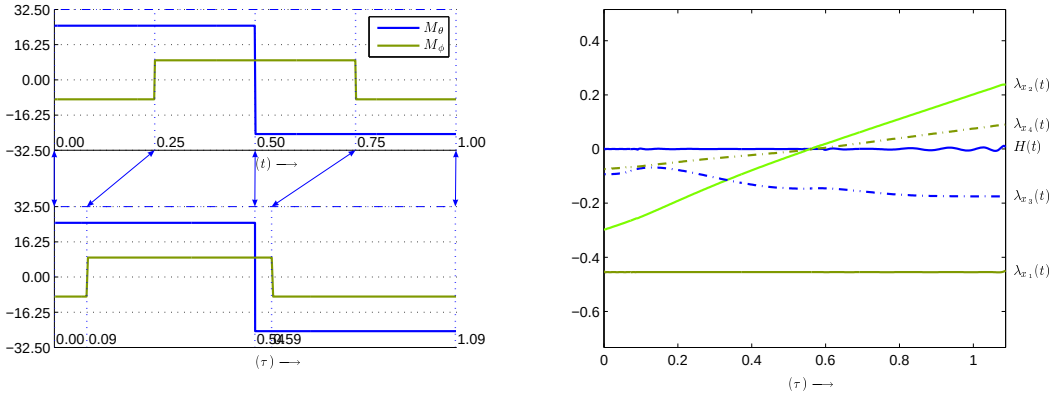
$$J_1 = 1.600 \text{ kg m}^2 \quad J_2 = 0.430 \text{ kg m}^2, \quad J_3 = 0.010 \text{ kg m}^2,$$
$$J_4 = 0.805 \text{ kg m}^2, \quad J_5 = 4.960 \text{ kg m}^2, \quad J_6 = 1.350 \text{ kg m}^2,$$
$$J_7 = 0.815 \text{ kg m}^2.$$

Starting at time $t = 0$, the optimal control problem is to find the control inputs u_1 and u_2 which drive the robot from its initial state to a prescribed final state while minimizing the final time t_f. The initial and final conditions are specified as,

$x_1(0)$	0	$x_1(t_f)$	0.95
$x_2(0)$	0	$x_2(t_f)$	0
$x_3(0)$	0	$x_3(t_f)$	0
$x_4(0)$	0	$x_4(t_f)$	0

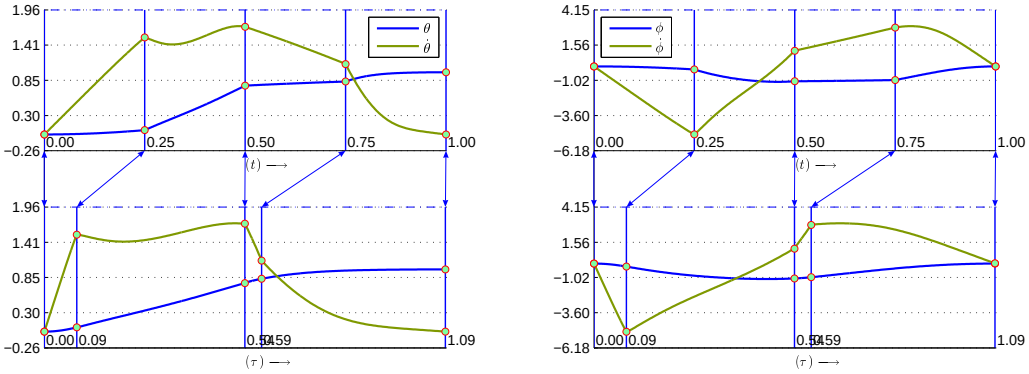
.

The control structure of this problem has been studied in a greater detail in Ref [66]. We solve this problem using 4 control phases. The states are approximated as 14^{th} degree polynomials in each phase. We use 40 LGL nodes in each phase. The results are shown in Figures (60), (61) and (62).



(a) (b)

Fig. 60. (a) Optimal torques M_θ and M_ϕ for the robot. (b) Costates.



(a) States θ and $\dot{\theta}$. (b) State ϕ and $\dot{\phi}$.

Fig. 61. State trajectories for the robot motion planning problem.

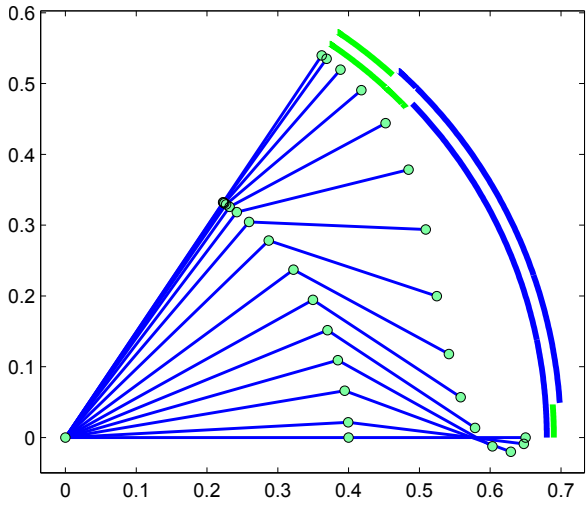


Fig. 62. Stroboscopic picture of the robot motion. The solid inner arc represents the "+" (blue) and "−" (green) state of M_θ. The solid outer arc show the "+" (blue) and "−" (green) state of M_ϕ

Comparing the results in Table IX with Ref. [66], we find that the switching times and the final time in our solution match with the reported values to the 3^{rd} decimal place.

Table IX. Input-output data for robot problem

Input Data	
No. of phases	4
Switching structure	$M_\theta : (+,+,-,-)$
	$M_\phi : (-,+,+,-)$
Polynomial order	$States:\ 14 \quad Controls : fixed$
Results	
Switching times	$\begin{bmatrix} 0.0877954 \\ 0.542835 \\ 0.588056 \\ 1.0857 \end{bmatrix}$
EqvI, EqvII	$\begin{bmatrix} 5.61\text{e-}003 & 5.63\text{e-}003 \\ 3.68\text{e-}003 & -3.40\text{e-}003 \\ 1.15\text{e-}003 & 2.10\text{e-}003 \\ 9.15\text{e-}004 & -2.36\text{e-}004 \end{bmatrix}$
$\|\mathcal{H}(t)\|_2$	2.69e-006
Optimal cost	1.0857

D. Conclusions

A time-scaling technique is presented for solving optimal control problems having discontinuous control solutions. The problem is divided into a number of phases and their lengths are treated as parameters of optimization. The least square method for optimal control is used as the underlying direct transcription method. Application problems are solved in MATLAB using SNOPT as the optimizer. The example problems demonstrate that the method performs well in solving the unknown switching times and control histories for real-life problems.

CHAPTER XI

CONCLUSIONS

This dissertation presented three novel methods for direct transcription of optimal control problems. The method of Hilbert space projection (MHSP), the least square method for optimal control (LSM_{oc}) and the generalized moment method for optimal control (GMM_{oc}) were derived in a unifying framework based on the weighted residual approximation of the state dynamics. These methods are flexible with respect to the choice of basis functions as both local and global approximating functions can be employed. Optimality analysis for all proposed methods was carried out and conditions were specified under which the costate variables can be estimated.

The weighted residual formulation of an optimal control problem provides a generic framework of analysis, under which three existing pseudospectral methods were formulated and analyzed. It was shown that Legendre, Radau and Gauss pseudospectral methods can be derived from WRM by judiciously choosing test and trial functions along with an associated numerical quadrature scheme. Further, spectral versions of LSM_{oc} and GMM_{oc} were derived by using global interpolating polynomials as approximating functions.

Numerical results were presented to demonstrate the accuracy and convergence of all three methods. It was seen that GMM_{oc} transcription performs very well for both local and global basis functions. The convergence rates of spectral GMM_{oc} are comparable to the existing pseudospectral methods like Gauss or Radau pseudospectral methods. s-GMM_{oc} has advantage over these methods because it does not suffer from any boundary defects.

Based on the variational analysis of first-order optimality conditions for the optimal control problem, an posteriori error estimation procedure was developed. Using

these error estimates, an h-adaptive scheme was outlined for the implementation of $\mathrm{LSM}_{\mathrm{oc}}$ in an adaptive manner. Several real-life examples were solved to show the efficacy of the h-adaptive algorithm.

A time-scaling technique was described to handle problems with discontinuous control, multiple phases or known control structure. A number of real-life examples were solved to demonstrate the applicability of this technique.

REFERENCES

[1] A. E. Bryson, "Optimal control-1950 to 1985," *IEEE Control Systems Magazine*, vol. 16, no. 3, pp. 26–33, 1996.

[2] H. H. Goldstine, *A History of the Calculus of Variations from the 17th through the 19th Century*. New York: Springer-Verlag, 1980.

[3] H. J. Sussmann and J. C. Willems, "300 years of optimal control: From the brachystochrone to the maximum principle," *IEEE Control Systems Magazine*, vol. 17, no. 3, pp. 32–44, 1997.

[4] R. E. Bellman, *Dynamic Programming*. Princeton, NJ: Princeton University Press, 1957.

[5] R. Gamkrelidze, "Discovery of the maximum principle," *Journal of Dynamical and Control Systems*, vol. 5, no. 4, pp. 437–451, 1999.

[6] J. T. Betts, "Survey of numerical methods for trajectory optimization," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 2, pp. 193–207, 1998.

[7] D. E. Kirk, *Optimal Control Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1970.

[8] L. S. Pontryagin, V. Boltyanskii, R. Gamkrelidze, and E. Mischenko, *The Mathematical Theory of Optimal Processes*. New York: Wiley-Interscience, 1962.

[9] F. L. Lewis and V. L. Syrmos, *Optimal Control*. New York: Wiley-Interscience, 1995.

[10] A. E. Bryson and Y. Ho, *Applied Optimal Control*. New York: Taylor & Francis, 1975.

[11] J. T. Betts, *Practical Methods for Optimal Control Using Nonlinear Programming*. Philadelphia: SIAM, 2001.

[12] P. E. Gill, W. Murray, and M. A. Saunders, "SNOPT: An SQP algorithm for large-scale constrained optimization," *SIAM Journal on Optimization*, vol. 12, no. 4, pp. 979–006, 1997.

[13] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright, "User's guide for NPSOL (version 4.0): A Fortran package for nonlinear programming," Stanford University, Stanford, CA, Tech. Rep. ADA169115, 1986.

[14] W. G. Vlases, S. W. Paris, R. M. Lajoie, M. J. Martens, and C. R. Hargraves, "Optimal trajectories by implicit simulation," Boeing Aerospace and Electronics, Tech. Rep. WRDC-TR-90-3056, 1990.

[15] J. T. Betts and W. P. Huffman, "Sparse optimal control software - socs," Mathematics and Engineering Analysis Library, Tech. Rep. MEA-LR-085, 1997.

[16] O. V. Stryk, *Users Guide for DIRCOL 2.1: A Direct Collocation Method for the Numerical Solution of Optimal Control Problems*, Technische Universität Darmstadt, Darmstadt, Germany, 1999.

[17] M. Milam, K. Mushambi, and M. Murray, "A new computational approach to real-time trajectory generation for constrained mechanical systems," in *Proc. of the 39th IEEE Conference on Decision and Control*, Sydney, 2000, pp. 845–851.

[18] I. M. Ross and F. Fahroo, "Users manual for dido 2002: A matlab application package for dynamic optimization," Naval Postgraduate School, Monterey, CA, Tech. Rep. AA-02-002, 2002.

[19] P. Williams, *Users Guide to DIRECT Version 1.16*, RMIT University, Melbourne, Australia, 2005.

[20] R. Bhattacharya, "Optragen: A matlab toolbox for optimal trajectory generation," in *Proc. of the 45th IEEE Conference on Decision and Control*, San Diego, CA, 2006, pp. 6832–6836.

[21] A. V. Rao, D. A. Benson, C. Darby, M. A. Patterson, C. Francolin, I. Sanders, and G. T. Huntington, "Gpops, a matlab software for solving multiple-phase optimal control problems using the gauss pseudospectral method," *ACM Transactions on Mathematical Software (TOMS)*, vol. 37, no. 2, pp. 1–39, 2010.

[22] P. Williams, "A comparison of differentiation and integration based direct transcription methods," *Advances in the Astronautical Sciences*, vol. 120, pp. 389–408, 2005.

[23] M. A. Kazemi and M. Miri, "Numerical solution of optimal control problems," in *Proc. of IEEE SoutheastCon Conference*, Charlotte, NC, 1993.

[24] G. N. Elnagar and M. A. Kazemi, "Pseudospectral chebyshev optimal control of constrained nonlinear dynamical systems," *Computational Optimization and Applications*, vol. 11, no. 2, pp. 195–217, 1998.

[25] G. Elnagar, M. A. Kazemi, and M. Razzaghi, "Pseudospectral legendre method for discretizing optimal control problems," *IEEE Transactions on Automatic Control*, vol. 40, no. 10, pp. 1793–1796, 1995.

[26] P. Williams, "A quadrature discretization method for solving optimal control problems," *Advances in the Astronautical Sciences*, vol. 119, pp. 703–721, 2005.

[27] D. A. Benson, G. T. Huntington, T. P. Thorvaldsen, and A. V. Rao, "Direct trajectory optimization and costate estimation via an orthogonal collocation method," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 6, pp. 1435–40, 2006.

[28] Q. Gong, I. M. Ross, W. Kang, and F. Fahroo, "Connections between the covector mapping theorem and convergence of pseudospectral methods for optimal control," *Comput. Optim. Appl.*, vol. 41, no. 3, pp. 307–335, 2008.

[29] W. W. Hager, "Runge-kutta methods in optimal control and the transformed adjoint system," *Numerische Mathematik (Germany)*, vol. 87, no. 2, pp. 247–82, 2000.

[30] V. O. Stryk and R. Bulirsch, "Direct and indirect methods for trajectory optimization," *Annals of Operations Research*, vol. 37, no. 1-4, pp. 357–73, 1992.

[31] P. Williams, "Hermite-legendre-gauss-lobatto direct transcription methods in trajectory optimization," *Advances in the Astronautical Sciences*, vol. 120, pp. 465–484, 2005.

[32] I. M. Ross and F. Fahroo, "Legendre pseudospectral approximations of optimal control problems," in *Lecture Notes in Control and Information Sciences*, vol. 295. New York: Springer-Verlag, 2003, pp. 327–342.

[33] Q. Gong, I. M. Ross, W. Kang, and F. Fahroo, "On the pseudospectral covector mapping theorem for nonlinear optimal control," in *Proc. of the 45th IEEE Conference on Decision and Control*, San Diego, CA, 2006, pp. 2679–2686.

[34] J. T. Betts and W. P. Huffman, "Mesh refinement in direct transcription methods for optimal control," *Optimal Control Applications and Methods*, vol. 19, no. 1,

pp. 1–21, 1998.

[35] T. Binder, L. Blank, W. Dahmen, and W. Marquardt, "Grid refinement in multiscale dynamic optimization," *RWTH Aachen, Tech. Rep. LPT-2000-11*, 2000.

[36] T. Binder, A. Cruse, C. A. C. Villar, and W. Marquardt, "Dynamic optimization using a wavelet based adaptive control vector parameterization strategy," *Computers & Chemical Engineering*, vol. 24, no. 2–7, pp. 1201 – 1207, 2000.

[37] S. Jain and P. Tsiotras, "Trajectory optimization using multiresolution techniques," *Journal of Guidance, Control, and Dynamics*, vol. 31, no. 5, pp. 1424–1436, 2008.

[38] I. M. Ross and F. Fahroo, "Pseudospectral knotting methods for solving nonsmooth optimal control problems," *Journal of Guidance, Control, and Dynamics*, vol. 27, no. 3, pp. 397–405, 2004.

[39] D. Bertsekas, *Nonlinear Programming*. Belmont, MA: Athena Scientific, 1995.

[40] P. Davis, *Interpolation and Approximation*. New York: Dover Publications, 1975.

[41] E. Kreyszig, *Introductory Functional Analysis with Applications*. New York: Wiley, 1989.

[42] C. D. Boor, *A Practical Guide to Splines*. New York: Springer, 1978.

[43] L. B. Lucy, "A numerical approach to the testing of the fission hypothesis," *The Astron. J.*, vol. 82, pp. 1013–1024, 1977.

[44] R. A. Gingold and J. J. Monaghan, "Smoothed particle hydrodynamics: Theory and application to non-spherical stars," *Monthly Notices Royal Astronomical Society*, vol. 181, pp. 375–389, 1977.

[45] W. K. Liu, S. Jun, and Y. F. Zhang, "Reproducing kernel particle methods," *International Journal for Numerical Methods in Engineering*, vol. 20, pp. 1081–1106, 1995.

[46] D. Shepard, "A two-dimensional interpolation function for irregularly spaced points," in *Proc. of the 23rd ACM National Conference*, New York, 1968, pp. 517–524.

[47] J. M. Melenk and I. Babuska, "The partition of unity finite element method: Basic theory and applications," *Computer Methods in Applied Mechanics and Engineering*, vol. 139, pp. 289–314, 1996.

[48] T. Strouboulis, K. Copps, and I. Babuska, "The generalized finite element method: An example of its implementation and illustration of its performance," *International Journal for Numerical Methods in Engineering*, vol. 47, no. 8, pp. 1401–1417, 2000.

[49] J. L. Junkins, G. W. Miller, and J. R. Jancaitis, "A weighting function approach to modeling of geodetic surfaces," *Journal of Geophysical Research*, vol. 78, no. 11, pp. 1794–1803, 1973.

[50] P. Singla, "Multi-resolution methods for high fidelity modeling and control allocation in large-scale dynamical systems," Ph.D. dissertation, Texas A&M University, College Station, TX, 2006.

[51] C. A. J. Fletcher, *Computational Galerkin Methods*. New York: Springer-Verlag, 1984.

[52] P. Williams, "Jacobi pseudospectral method for solving optimal control problems," *Journal of Guidance, Control, and Dynamics*, vol. 27, no. 2, pp. 293–297, 2004.

[53] H. T. Banks and F. Fakhroo, "Legendre-tau approximations for lqr feedback control of acoustic pressure fields," *Journal of Mathematical Systems, Estimation, and Control*, vol. 5, no. 2, pp. 1–34, 1995.

[54] J. Vlassenbroeck, "Chebyshev polynomial method for optimal control with state constraints," *Automatica*, vol. 24, no. 4, pp. 499–506, 1988.

[55] J. Vlassenbroeck and D. R. Van, "Chebyshev technique for solving nonlinear optimal control problems," *IEEE Transactions on Automatic Control*, vol. 33, no. 4, pp. 333–340, 1988.

[56] M. Razzaghi, "Fourier series direct method for variational problems," *International Journal of Control*, vol. 48, no. 3, pp. 887–895, 1988.

[57] H. Hua, "Numerical solution of optimal control problems," *Optimal Control Applications and Methods*, vol. 21, no. 5, pp. 233–241, 2000.

[58] G. T. Huntington, "Advancement and analysis of a gauss pseudospectral transcription for optimal control problems," Ph.D. dissertation, MIT, Cambridge, MA, 2007.

[59] D. Benson, "A gauss pseudospectral transcritopn for optimal control," Ph.D. dissertation, MIT, Cambridge, MA, 2005.

[60] F. Fahroo and I. M. Ross, "Costate estimation by a legendre pseudospectral method," *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 2, 2001.

[61] S. Kameswaran and L. Biegler, "Convergence rates for direct transcription of optimal control problems at radau points," *Computational Optimization and Applications*, vol. 41, no. 1, pp. 81–126, 2008.

[62] X. Bai, J. Turner, and J. Junkins, "Bang-bang control design by combing pseudospectral method with a novel homotopy algorithm," in *AIAA Guidance, Navigation, and Control Conference*, no. AIAA-2009-5955, Chicago, IL, 2009.

[63] P. Williams, "Guidance and control of tethered satellite systems using pseudospectral methods," *Advances in Astronautical Sciences*, vol. 119, pp. 1045–1064, 2005.

[64] B. Singh and R. Bhattacharya, "Direct optimal control and costate estimation using least square method," in *Proc. of 2010 American Control Conference*, Baltimore, MD, 2010, pp. 1556–1561.

[65] L. Cremean, W. B. Dunbar, D. V. Gogh, J. Hickey, E. Klavins, J. Meltzer, and R. M. Murray, "The caltech multi-vehicle wireless testbed," in *Proc. of the 41st IEEE Conference on Decision and Control*, Las Vegas, NV, 2002, pp. 86–88.

[66] H. P. Geering, L. Guzzella, S. A. R. Hepner, and C. H. Onder, "Time-optimal motions of robots in assembly tasks," *IEEE Transactions on Automatic Control*, vol. 31, no. 6, pp. 512–518, 1986.

VITA

Baljeet Singh was born in Punjab, India in 1979. He received his baccalaureate (B.Tech.) and Masters (M.Tech.) degree in mechanical engineering from Indian Institute of Technology, Bombay, India in July 2003. In December 2003, he was employed with J. Ray McDermott ME Inc. as a mechanical engineer. He joined the Mechanical Engineering Department of Texas A&M University for his graduate studies in the fall of 2005. In December 2006, he joined the Aerospace Engineering Department of Teaxs A&M University and started his Ph.D. work under the supervision of Dr. Raktim Bhattacharya. His Ph.D. dissertation is focused on developing novel numerical methods and techniques for solving complex optimal control problems.

He may be reached at baljeet8dm@gmail.com or by contacting Dr. Raktim Bhattacharya, Department of Aerospace Engineering, Texas A&M University, College Station, TX-77843.