

**DIRECT AND INDIRECT MULTIPLE SHOOTING METHODS FOR
NONLINEAR OPTIMAL CONTROL PROBLEMS**

MSC PROJECT

DAWIT ERFITO

MARCH 2019

HARAMAYA UNIVERSITY, HARAMAYA

**Direct and Indirect Multiple Shooting Methods for Non-Linear Optimal
Control Problems**

A Project Submitted to Department of Mathematics,

Postgraduate Program Directorate

HARAMAYA UNIVERSITY

In Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE IN MATHEMATICS

(OPTIMIZATION)

Dawit Erfito

March 2019

Haramaya University, Haramaya

DEDICATION

I dedicate this project manuscript to my beloved wife Zena Ermolo and brother Daniel Erfito for their sacrifice throughout my education and support for the success of my life.

STATEMENT OF THE AUTHOR

By my signature below, I declare that this project is my own work. I have followed all ethical and technical principles of scholarship in the preparation, and compilation of this project. Any scholarly matter that is included in the project has been given recognition through citation. This project is submitted in partial fulfillment of the requirements for Master of Science in Mathematics at Haramaya University. The project is deposited in the Haramaya University Library and is made available to borrowers under the rules of the Library. I solemnly declare that this project has not been submitted to any other institution anywhere for the award of any academic degree, diploma or certificate.

Brief quotations from this project may be made without special permission provided that accurate and complete acknowledgement of the source is made. Requests for permission for extended quotations from or reproduction of this project in whole or in part may be granted by the head of the department or Director of Postgraduate Program Directorate when in his or her judgment the proper use of the material is in the interest of scholarship. In all other instances, however, permission must be obtained from the author of the project.

Name: Dawit Erfito Hangacho

Date: _____

Department: Mathematics

Signature _____

BIOGRAPHICAL SKETCH

The author was born on September 15 in 1988 in SNNP Regional State, Hadiya Zone, Soro wereda, Jacho kebele from his mother Etenesh Anebo and father Erfito Hangacho. He attended his primary education at Jacho primary school. Then after, he joined Gimbicho secondary school and Wachemo preparatory school to attend his secondary education. He then joined Dilla University in 2008 and received Bachelor of Education degree in Mathematics in July, 2011. Then he has been employed in SNNP Regional State, Hadiya zone, Anlemo wereda, fonko secondary school in 2011 and worked for five years in the school, he joined postgraduate program at Haramaya University, College of Natural and Computational Sciences, Department of Mathematics in 2017 to pursue a program of study for MSc. degree in Mathematics with specialization in Optimization.

ACKNOWLEDGEMENT

First and for most, I would like to be grateful for the loving and kindness of the Almighty God in bestowing my health, strength, patience, and protection throughout my life. My profound sincere appreciation and thanks go to my course instructors and my project advisors **Dr. Getinet Alemayehu** and **Dr. Melisew Tefera** who had been kindly and extensively advised me by giving constructive comments and make improvement on this project.

I wish to extend my sincere thanks to all my family, who have been constant source of encouragement and support.

Finally, I would like to express my heartfelt thanks to Anlemo worda and Haramaya University Mathematics department who gave me encouragement and helped me during this postgraduate study.

ABBREVIATIONS

BVP	Boundary Value Problem
DM	Direct Method
DMS	Direct Multiple Shooting
IM	Indirect Method
IMS	Indirect Multiple Shooting
IVP	Initial Value Problem
MPBVP	Maximum Principle Boundary Value Problem
NLP	Non-Linear Problem
NNOC	Nearest Neighbor Optimal Control
OCP	Optimal Control Problem
ODE	Ordinary Differential Equation
PMP	Pontryagin Minimum/Maximum Principle
TPBVP	Two Point Boundary Value Problem

TABLE OF CONTENTS

BIOGRAPHICAL SKETCH	vi
ACKNOWLEDGEMENT	vii
ABBREVIATIONS	viii
LIST OF TABLES	xi
LIST OF FIGURES	xii
ABSTRACT	xiii
1. INTRODUCTION	1
1.1. Background of the Study	1
1.2. Statement of the Problem	8
1.3. Objective of the Study	9
2. REVIEW OF RELATED LITERATURE	10
2.1. Nonlinear Optimal Control Problem	10
2.2. Direct Methods	12
2.3. Indirect Methods	14
3. MATERIALS AND METHODS	18
4. PRELIMINARY	19
4.1. Optimal Control Problem	19
4.2. Shooting Methods	23
4.2.1. Direct Single Shooting Method	23
4.2.2. Direct Multiple Shooting Method	25
4.3. Errors	27
5. INDIRECT AND DIRECT METHODS	32
5.1. Direct Method	32

5.2. Indirect Method	43
5.2.1. Calculus of Variations	43
5.2.2. Variational Optimal Control	47
5.3. Comparisons of DMS and IMS for Optimal Control (OCP)	60
6. SUMMARY AND CONCLUSIONS	63
6.1. Summary	63
6.2. Conclusion	64
7. REFERENCES	65
8. APPENDICES	69
8.1. Matlab Code A	69
8.2. Matlab Code B	71
8.3. Matlab Code C	73
8.4. Matlab Code D	75

LIST OF TABLES

Table	Page
5.1. Numerical results for DMS in example 5.1.	38
5.2. Numerical results for DMS in example 5.2.	41
5.3. Numerical results for IMS in example 5.3.	55
5.4. Numerical results for IMS in example 5.4.	59
5.5. Comparisons of numerical results for DMS and IMS in example 5.1 and 5.3.	60
5.6. Comparisons of numerical results for DMS and IMS in example 5.2 and 5.4.	61

LIST OF FIGURES

Figure	Page
4.1. Illustration of the direct single shooting discretization applied to the OCP.	24
4.2. Illustration of the direct multiple shooting discretization applied to the OCP.	25
5.1. Numerical solutions for DMS in example 5.1.	38
5. 1. Numerical solutions for DMS in example 5.2.	41
5.3. Numerical solutions for IMS in example 5.3.	56
5.4. Numerical solutions for IMS in example 5.4.	60
5.5. Numerical solutions for DMS and IMS in example 5.1 and 5.3.	61

Direct and Indirect Multiple Shooting Methods for Non-Linear Optimal Control Problems

ABSTRACT

In this project, we discussed nonlinear optimal control problems. Nonlinear optimal control problems are challenging to solve due to the prevalence of local minimum that prevent convergence or optimality. This project was concerned with two numerical methods (indirect and direct methods). Indirect method based on Pontryagin's Maximum Principle optimize in an infinite dimensional function space and derive the necessary conditions for optimality using co-state variables and convert the optimal control problem into a two point boundary value problem, then solving the two point boundary value problem numerically. Direct methods were based on a discretization of the infinite dimensional optimal control problem into a finite dimensional nonlinear optimization problem. This study was described nearest neighbors optimal control. The presented concepts and methods were verified by means of different examples, where by theoretical results are numerically confirmed. We choose the test problems, so that the simple shooting method becomes unstable and a genuine multiple shooting techniques were required. By comparisons of both methods we obtained that indirect multiple shooting methods is more accurate than direct multiple shooting method. Most computations were performed using MATLAB codes. This project can be extending to solve nonlinear optimal control problems with inequality constraints using these two methods.

Key words: *Nonlinear optimal control problem, direct and indirect multiple shooting methods.*

1. INTRODUCTION

1.1. Background of the Study

Optimal control is an important subject in modern control theory. Apart from traditional areas such as aerospace engineering, robotics, and chemical engineering, optimal control theory has also been used with great success in areas as diverse as economics to biomedicine. Many optimal control problems do not have computable solutions, or they have solutions that may be obtained only with a great deal of computing effort. The main difficulty arises in seeking a closed form solution to the Hamilton-Jacobi-Bellman equations, or in solving the canonical Hamiltonian equations resulting from an application of the maximum principle (Pontryagin,1959).

In the classical development, the variational method of optimal control theory, which typically consists of the calculus of variations and Pontryagin's methods, can be used to derive a set of necessary conditions that must be satisfied by an optimal control law and its associated state-control equations (Pinch,1993). Necessary conditions of optimality lead to a generally nonlinear two-point boundary value problem (TPBVP) that must be solved to determine the explicit expression for the optimal control. Many authors have studied optimal control of nonlinear systems for decades. Tang (2005) applied a successive approximation approach to nonlinear systems. Shirazia (2012) used the modified variational iteration method for nonlinear quadratic optimal control problems.

Definition of some key terms in optimal control problems (Elsgolts,1962):

- ❖ Scalars: A problem may also contain unknown scalar variables that were solve for at the same time as the control variable and state variable.
- ❖ State variable: Is one of the set of variables that was used to describe the mathematical of dynamical system.
- ❖ Control variable: It is a quantity or condition that was measured and controlled the systems.

- ❖ Final state: Some of the states that are not eventually specified at final time through which is possible to modify the cost function.
- ❖ Parameters: A constant numbers that are used to define a problem.
- ❖ Functional: A variable quantity J is a functional dependent on a function $f(x)$, written as $J = J(f(x))$, if to each function $f(x)$, there corresponds a value J , i.e., we have a correspondence to the function $f(x)$ there corresponds a number J . Functional depends on several functions. Example: Let $x(t) = 2t^2 + 1$. Then $J(x(t)) = \int_0^1 x(t)dt = \int_0^1 (2t^2 + 1) dt$.
- ❖ Plant: For the purpose of optimization, we describe a physical plant by a set of nonlinear differential equations. For example a nonlinear systems characterized by $\dot{x}(t) = f(t, x(t), u(t))$.

Indirect methods are based on Pontryagin's maximum principle, adapting the sign convention for minimization problem. This principle's derivation employs calculus of variations techniques, of which comprehensive references are (Bliss,1946). The goal is to convert the optimal control problem into a two point boundary value problem through the statement of the necessary conditions that a profile shall satisfy to be an optimal solution. The process starts with the definition of an augmented performance index \tilde{J} , in a fashion similar to equality-constrained static optimization problems, where Lagrange's multipliers λ multiplying the dynamical constraints are summed to the objective function to form the augmented performance index:

$$\tilde{J} = \Phi + \int_{t_0}^{t_f} [L[t, x(t), u(t)] + \lambda^T(t)\{f[t, x(t), u(t)] - \dot{x}\}] \quad (1.1)$$

According to calculus of variation, the necessary condition for a stationary extremum is that the first-order variation $\delta\tilde{J}$ shall nullify at any instant of time for any constraint-allowed variation $\delta u(t)$. The problem Hamiltonian is defined as:

$$H = L[t, x(t), u(t)] + \lambda^T(t)f[t, x(t), u(t)] \quad (1.2)$$

when path constraints are present, the Hamiltonian shall be augmented with the constraints' violation weighted by associated dual variables. After mathematical manipulation Hartjes et al. (2014) for a detailed derivation, the necessary conditions

for a control profile $u^*(t)$ to be a stationary function of the performance index are represented by the following Euler-Lagrange equation:

$$\dot{x} = f(t, x(t), u(t)) \quad (1.3)$$

$$\dot{\lambda} = - \left[\frac{\partial H}{\partial x} \right]^T \quad (1.4)$$

$$0 = \left[\frac{\partial H}{\partial u} \right]^T \quad (1.5)$$

where the relations in (1.4) are labeled as adjoint equations and (1.5) as control equations. These differential equations, which a control profile has to necessarily satisfy to be a stationary solution, are coupled with a set of transversality conditions:

$$\begin{aligned} t_0 \text{ given or } H(t_0) &= 0 \\ t_f \text{ given or } H(t_f) &= - \left. \frac{\partial \Phi}{\partial t} \right|_{t_f} \\ x_0 \text{ given or } \lambda(t_0) &= 0 \\ x_f \text{ given or } \lambda(t_f) &= - \left. \frac{\partial \Phi}{\partial x} \right|_{t_f} \end{aligned} \quad (1.6)$$

hence, if any of the boundary conditions is a free parameter, either on time or state variables, the above conditions complete the minimum required number of known conditions at the initial or final time. In the matter of low-thrust, final conditions could be unspecified (but possibly still constrained) in many non-rendezvous trajectories such as orbit raising or decreasing, orbital escape problems, gravity assists and so on. Up to this point, the process defined the necessary conditions for a solution to be a stationary one. The Legendre-Clebsch condition about local convexity of the Hamiltonian shall be satisfied to ensure that the solution is an actual local minimum:

$$\left. \frac{\partial^2 H}{\partial u^2} \right|_{u^*} \geq 0 \quad (1.7)$$

The TPBVP defined by (1.3), coupled with the conditions (1.6) and (1.7), has no analytical closed-form solution for complex problems like low-thrust trajectories. Hence, numerical methods shall be employed. However, further information can be obtained by exploitation of the problem's first integrals. If the functions L and f defined in the System do not depend explicitly on the independent variable t , then the Hamiltonian is a first integral of the TPBVP along an optimal trajectory (Kemble, 2006). In general, if a first integral is found, the redundant information that it generates can be exploited to eliminate one adjoint equation, formally transforming the original TPBVP into another one of lower dimension, by following the procedure shown by (Hartjes et al., 2014).

Numerical solution of the optimal control problem for first approach corresponds to the indirect method. For this method the first step is the formulation of the appropriate two-point Boundary Value Problem (TPBVP) and the second step is solving the TPBVP numerically. The indirect approach for the optimal control problem is based on a generalization of the calculus of variations. Necessary conditions for an extremum are derived by considering the first variation of the performance index with constraints adjoined in the manner of Lagrange. Since the setting is infinite-dimensional, the familiar Lagrange multipliers are now functions of time, and are called co-states in analogy to the system state.

While in the finite-dimensional case the multipliers are computed from algebraic equations, the co-states obey a differential equation. The necessary conditions entail both the original differential equations of the underlying dynamical system and the associated adjoint differential equations of the co-states. The end result is a TPBVP which is made up of the state and co-states equations together with the initial and terminal conditions.

For indirect methods, multiple shooting works as follows:

- The time horizon is divided into several segments and at the beginning of each segment initial values for the state and the adjoint variables are guessed. Starting from those initial values, the state and adjoint differential equations are integrated to the end of the segment.
- Error equations are set up from initial conditions, from the mismatch between final values of the state and adjoint variables at the end of each segment and the initial values of the subsequent segment, and from final conditions. These error equations are steered to zero by an iterative Newton method, which varies the initial values.
- Bulirsch et al.(1991) it was focused on multiple shooting as solution method for MPBVPs containing arcs with singular controls and state constraints. Local feedback control laws in the linearized tangent space around an optimal solution are precalculated and used on line Pesch (1989) to increase the robustness against perturbations.
- Multiple shooting solves optimal control problems with high accuracy and near the optimum the method converges quadratically.
- Multiple shooting can be implemented straight forwardly. Disadvantages are that the domain of convergence is relatively small compared to other methods due to integrating either the state or adjoint differential equations into the instable direction. This and the fact that the adjoint variables have no physically intuitive equivalence cause the initialization to be difficult, especially in comparison to other approaches. Further, the sequence of constrained and unconstrained arcs has to be known a priori.

The numerical solution of optimal control problem of second approach corresponds to the direct method, which is based on the discretisation of state and/or control variables over time, so that a Nonlinear Problem (NLP) solver can be used. A direct method does not require the derivation of the necessary conditions needed by indirect methods. On the contrary, it aims to find a sequence of profiles which progressively reduce the non-augmented performance index J and the constraint's

violation. Direct methods require a parameterization of the control functional form over trajectory's arcs. This is generally achieved by two conceptually different methods (Kemble, 2006).

- A grid at different times where the control parameters are to be found and the values within an interval are computed through interpolation.
- A set of orthogonal basis of mathematical functions dependent on time. Usually Fourier series, Legendre polynomials or the Chebyshev ones.

The goal is then to determine the values of the specified free parameters, either control values at fixed times in the grid form or the coefficients of the series in the second case, able to minimize the objective index and to respect the constraints. In this passage, the number of free parameters is reduced from infinite degrees of freedom to a finite number of parameters, depending on the chosen parameterization. This passage could seem a limitation of the direct methods when compared to the indirect ones. However, as already stated in the previous section, a numerical procedure is necessary also for indirect methods when dealing with complex cases such as low-thrust trajectory optimization. These numerical methods require a so-called transcription to convert the infinite dimension optimal problem into a solvable finite-dimension one. Hence, what seemed a limitation of the direct methods is a required passage of any technique nonetheless.

A direct method's solution is generally not an optimal solution itself, i.e. not a local minimum of the performance index, but just an approximation as a consequence of the discretization or interpolation steps. Hence, the necessary conditions (1.3) and (1.5) can be used as an indicator of how close the solution to the real local optimum (Conway, 2010). We described the direct multiple shooting method Book and Plitt (1984) as an efficient tool for the discretization and parameterization of a broad class of OCPs. We consider the following general class of optimal control problems

$$\begin{aligned} \text{Min } J(u) &= \int_{t_0}^{t_f} l(t, x(t), u(t)) dt \\ \text{Subject to } \dot{x} &= f(t, x(t), u(t)) \quad \forall t \in [t_0, t_f] \end{aligned}$$

$$0 \leq C(t, x(t), u(t)) \quad \forall t \in [t_0, t_f]$$

$$0 \leq r(t_i, x(t_i)) \quad 0 \leq i \leq m$$

in which we strive to minimize objective function $l(\cdot)$ depending on the trajectory $x(\cdot)$ of a dynamic process described in terms of a system f of ordinary differential equations the time horizon $[t_0, t_f] \subset \mathbb{R}$, and governed by a control trajectory $u(\cdot)$ subject to optimization. The process trajectory $x(\cdot)$ and the control trajectory $u(\cdot)$ shall satisfy certain inequality path constraints c on the time horizon $[t_0, t_f]$, as well as equality point constraints r on a grid of $m + 1$ grid points on $[t_0, t_f]$, $t_0 < t_1 < \dots < t_{m-1} < t_m = t_f$, $m \in \mathbb{N}$, $m \geq 1$. The direct multiple shooting method is applied to discretize the control trajectory $u(\cdot)$ to make this infinite dimensional problem computationally accessible.

1.2. Statement of the Problem

Nonlinear optimal control problems are observed in many engineering applications. Despite decades of research, they are still difficult to solve globally with high confidence. Methods for solving optimal control problems mainly fall in to two categories: direct methods and indirect methods (Betts,1998). Direct methods convert the optimal control problem in to an optimization problem through parameterization. Indirect methods derive the necessary conditions for optimality using co-state variables, and convert the optimal control problem into a two point boundary value problem (TPBVP) (Bryson,1975). However, they are difficult to apply successfully because the TPBVP has a narrow convergence domain (Jiang, 2012). The main difficulty of indirect methods is the co-state variables lack physical meanings, so good starting values are difficult to provide. For problems with strong nonlinearity, the convergence domain is so narrow that a large number of initial guesses have to be tried to obtain convergence.

A nonlinear optimal problem would be given as:

$$\text{Minimize } J(u) = \int_{t_0}^{t_f} L(t, x(t), u(t))dt \quad (1.8)$$

$$\text{Subject to } \dot{x}(t) = f(t, x(t), u(t))$$

where $J(u)$ was objective function, t was stands for the independent variable, usually $t \in [t_0, t_f]$ was time, x and u are functions of time, L and f are continuous in (t, x, u) and had continuous partial derivatives with respect to x and u .

1.3. Objective of the Study

The general objective of this project was to solve nonlinear optimal control problem using indirect and direct multiple shooting methods.

The specific objectives of this study were:

- To discuss nonlinear optimal control problem.
- To describe direct and indirect multiple shooting methods.
- To solve nonlinear optimal control problems using direct and indirect multiple shooting methods.
- To explain how we can estimate the associated error.
- To compare indirect and direct multiple shooting methods.

2. REVIEW OF RELATED LITERATURE

2.1. Nonlinear Optimal Control Problem

Optimal control of nonlinear systems is one of the most challenging and difficult subjects control theory. It is well known that the nonlinear optimal control problem can be reduced to the Hamilton-Jacobi-Bellman partial differential equation Bryson and Ho (1975), but due to difficulties in its solution, this is not a practical approach. Instead, the search for nonlinear control schemes has generally been approached on less ambitious grounds than requiring the exact solution to the Hamilton-Jacobi-Bellman partial equation. In fact, even the problem of stabilizing a nonlinear system a challenging task.

Lyapunov theory, successful and widely used tool, is century old. Despite this, there still do not exist systematic methods for obtaining Lyapunov functions for general nonlinear systems. Nevertheless, the ideas put forth by Lyapunov early a century ago continue to be used and exploited extensively in the modern theory of control for nonlinear systems. One not ably successful use of the Lyapunov methodology is the concept of control Lyapunov function (CLF) Freeman and Primbs (1996) the idea of which is to rest choose a function which can be made in to a Lyapunov function for the closed loop system by choosing appropriate control actions. The knowledge of such a function is then used to design control laws. Once again, there do not exist systematic techniques for ding control Lyapunov function for general. Nonlinear systems, but this approach has been applied successfully too many classes of systems for which control Lyapunov function can be found.

Solving a general nonlinear optimal control problem (OCP) is a difficult challenge, despite powerful theoretical tools are available, e.g. the maximum principle and Hamilton-Jacobi-Bellman (HJB) optimality equation. The problem is even more difficult in the presence of state and/or control constraints. State constraints are particularly difficult to handle, and the interested reader is referred to Capuzzo-Dolcetta and Lions (1990) for a detailed account of HJB theory in the case of state

constraints. There exist many numerical methods to compute the solution of a given optimal control problem; for instance, multiple shooting techniques which solve two-point boundary value problems as described, or direct methods, which use, among others, descent or gradient-like algorithms.

To deal with optimal control problems with state constraints, some adapted versions of the maximum principle have been developed for a survey of this theory, but happen to be very hard to implement in general. On the other hand, the OCP can be written as an infinite-dimensional linear program (LP) over two spaces of measures. This is called the weak formulation of the OCP Vinter (1993) (stated in the more general context of differential inclusions). The two unknown measures are the state-action occupation measure up to the final time T , and the state occupation measure at time T . The optimal value of the resulting LP always provides a lower bound on the optimal value of the OCP, and under some convexity assumptions, both values coincide (Vinter, 1993)

2.2. Direct Methods

Direct methods are based on the transformation of the original optimal control problem into a nonlinear problem (NLP) by discretising the state and/or control history and then solving the resulting NLP problem. A variety of direct methods has been developed and applied for solving an optimal control problem. Gradient algorithms were proposed (Pontryagin, 1959). A state constrained optimal control problem using a gradient algorithm and applied it for some problems. Pinch (1993) reintroduced the direct transcription approach, by discretising the dynamic equations using a collocation method. A cubic polynomial is used to approximate the state variables and linear interpolation for the control variables. The collocation scheme was originally used by Tang (2005) to solve TPBVP. Shirazian and Effati (2012) introduced an approach based on the representation of the dynamical system in terms of differential inclusions.

Shooting methods can have difficulty when applied to open loop unstable systems, so by pre-compensating the system with Sontag's formula we removed this problem. In fact, when *riots* was applied to the open-loop system (which is unstable), we encountered numerous numerical difficulties. Hence, the facts that a control Lyapunov function also provides a stabilizing control law which can be used to pre stabilize. The dynamics before performing trajectory optimization is yet another example of the synergies available between control Lyapunov function and deciding horizon control. There resulting optimizations appeared to be very well conditioned for shooting techniques, and no further numerical problems were encountered.

Direct methods solve optimal control problems for purely continuous, nonlinear systems based on a discretization with respect to time. That means the infinite dimensional problem of finding optimal state and control trajectories is transformed into a finite dimensional optimization problem by evaluating state and control values only at a finite number of time samples. By varying the state and control values at those samples, the cost criterion is directly optimized. The optimization can be

performed by linear or nonlinear programming methods like sequential quadratic programming (SQP) (Nocedal and Wright, 1999).

The optimal control problem is solved by numerical method using a direct multiple shooting. Direct multiple shooting to trajectory optimization is generally based on the discretisation of control and/or state variables. The basic idea of the direct multiple shooting methods is to transform the original optimal control problem in to nonlinear programming problem by coupling the control parameterization with a multiple shooting discretisation of the state variables (stoer and Bulirsch, 2002). A comparison of collocation and shooting methods is given (Betts, 1998). Collocation methods usually require a finer discretization grid, so that the resulting optimization problem is larger.

Furthermore, direct multiple shooting is more suitable for stiff problems since error-controlled solvers can be used for the integration of the system dynamics, where as an adaptive grid refinement for error control is not straightforward in collocation (Sager, 2006). The finer grid in collocation methods offers slight advantages in considering state constraints. However, both methods can only enforce state constraints at nodes of the discretized time or if the sequence of constrained and unconstrained arcs of the system trajectory is known, multiple shooting can also maintain state constraints with higher precision.

In the 1980s, an alternative shooting approach was developed which is known as direct multiple shooting (DMS). The most influential publications in this context originated from Bock and his co-workers. DMS is based on a discretization of the solution variables leading to a finite-dimensional optimization problem and is therefore a ‘first-discretize-then-optimize’ approach. The finite-dimensional problem is solved by suitable methods for nonlinear programming problems (NLP). The solution to optimal control problems in the direct methods is to transform the original problem to a nonlinear programming problem (NLP) and directly solving the NLP. The underlying philosophy in direct methods is “discretize, then optimize” as the direct methods discretize the state and/or control of the continuous OCP and then transcribe it to a nonlinear optimization problem.

In consequence, solving the optimal control problem directly consists of two major subtasks: solving the differential equations of system dynamics, and solving the resulting NLP. Direct methods have recently been extensively researched and they can avoid the main drawbacks of indirect methods, however their disadvantage is that they produce approximate solutions (Buskens and Maurer, 2000). Furthermore, direct methods can solve problems that have inequality constraints more easily than indirect methods. The approach to solve the two main steps can be done either sequentially or simultaneously.

The single shooting method, for example, is a sequential method in which the dynamics are first integrated and then the resulting trajectory is evaluated to optimize the parameters. In simultaneous methods both the simulation and optimization task are solved simultaneously e.g. in collocation method the trajectory points themselves are optimization variables and upon convergence the points become continuous and satisfy the dynamics constraints. In simultaneous methods, the number of variables and constraints is large, however, this so-called infeasible path approach can use prior knowledge of the trajectory and handles unstable systems more effectively than single shooting (Diehl et al., 2006).

2.3. Indirect Methods

An indirect method is known as inverse dynamic optimization (Seywald and Guid, 1994). It is regarded as an inverse control method since points on the state-variable trajectories are chosen so that the path is optimal. The control variable trajectories are found by numerical differentiation of the state trajectories. This method involves discretization and uses nonlinear programming. The method is limited since accuracy is lost as the number of time steps increases (Bryson, 1999). Bryson and Ho (1975) provide a thorough and comprehensive overview of necessary conditions for different types of unconstrained and constrained optimal control problems.

The optimal control problem is turned into a two point boundary value problem containing the same mathematical information as the original one by means of

necessary conditions of optimality. Then, the boundary value problem (BVP) is discretized by some numerical technique to get a solution. Numerical techniques for solving this two point boundary value problem can be classified as gradient methods Kelley (1960), indirect single shooting and indirect multiple shooting Stoer and Bulirsch (2002) and indirect collocation (Ascher et al., 1979). Indirect methods appeared first in a form of calculus of variations, which go back to the 17-th century, when Johan Bernoulli invited to the solution of a novel problem. The problem consisted in finding the time-optimal path of a particle in a gravitational field between any two points. Since then, the methodology was steadily developed theoretically and algorithmically.

For nonlinear optimal control problems solved by indirect methods, the optimal control might not be continuous. The system dynamical and corresponding Euler-Lagrange are usually difficult to integrate analytically so numerical integration has to be applied. Another problem is the strong nonlinearity of the problem, so which makes the convergence domain to be quite small. As a result, the optimal control for quite simple dynamical system is difficult to solve, so its application is limited. In Jiang et al. (2012) a homotopic approach is used so we can start from an easier problem and gradually approximate the original problem.

In Jiang et al. (2012) a distribution of initial co-state variables is proposed which more or less solve the problem that we do not know the bounds of co-state variables. Bulirsch et al. (1991) introduced shooting methods to the class of optimal control problems (OCP). His approach Marsden et al. (2007) is known as indirect multiple shooting (IMS). It is intuitive as it applies multiple shooting to the system of first order optimality conditions of such OCP. This optimality system constitutes the basis of most solution algorithms and is structurally similar to BVP.

The classical approach to solving optimal control problems is based on Pontryagin's maximum principle. The necessary conditions of optimality are used to transform the optimization problem to a multipoint boundary value problem that is solved, by multiple shooting, Bulirsch (1991). An optimal solution typically consists of several

arcs. On each arc we do have constraint-seeking or compromise-seeking controls. These controls are determined by the necessary conditions of optimality and have to be calculated analytically. The disadvantages of indirect methods are quite obvious.

The formulation of the boundary value problem in a numerically stable way requires a lot of knowhow and work. Furthermore already small changes in the value of a parameter or in the problem definition, e.g. an additional constraint, may change the switching structure completely. The switching times have to stay in the multiple shooting intervals, otherwise convergence of Newton's method is not ensured anymore. Only if the switching structure is guessed correctly in advance and does not change during the iterations of the multiple shooting algorithm, it is possible to transform the problem onto fixed switching times. Start values for all variables have to be delivered, which is often difficult especially for the adjoints. This is crucial, because one has to start inside the convergence region of Newton's method. In case of path constraints usually homotopies have to be applied to obtain such start values.

The main advantage of indirect methods is the high accuracy of the obtained solution, as the infinite dimensional problem has been solved and no approximations of the controls have been undertaken, in contrast to direct methods. As all degrees of freedom in the controls vanish, this approach seems appropriate for problems with a high number of control functions when compared to direct methods. If the number of states is large compared to the number of controls, direct methods are usually more efficient. In general, an interactive iterative process involving the solution of several multipoint boundary value problems is necessary to get a solution using indirect methods. Both, insight into the problem and specific numerical knowledge are typically required for this task. Consequently, nowadays indirect methods are most often applied when high accuracy of the solution is crucial and enough time for obtaining the solution is available, typically, initial guesses for the variables are generated by applying direct methods, (Bulirsch et al.,1991).

3. MATERIALS AND METHODS

This project work outlines the methods that were used to solve nonlinear optimal control problems, and materials used in the study. Sources in the web and libraries were used to collect all the pieces of information about nonlinear optimal control problems, indirect and direct multiple shooting methods. Specifically,

- ❖ Relevant journals and books consulted to gather information about nonlinear optimal control problem, indirect and direct multiple shooting methods
- ❖ The collected information analyzed and arranged.
- ❖ Important concepts, definitions, and theorems are discussed to make ideas clear.
- ❖ Indirect and direct multiple shooting methods were used to solve nonlinear optimal control problems.
- ❖ Appropriate examples were used to make comparison between indirect and direct multiple shooting methods for nonlinear optimal control problems.
- ❖ The error estimated to make comparison.
- ❖ MATLAB code employed to carry out all complicated computational tasks that arise in the solution methods.

4. PRELIMINARY

In this chapter, we deal with theoretical concepts, definitions and theorems which are important for the study of both direct and indirect multiple shooting methods to solve non-linear optimal control problems.

4.1. Optimal Control Problem

The origins of optimal control theory lie in Bernoulli's Brachistochrone problem from 1696 which asks to determine the curve along which a body travels from one fixed point to another in the shortest time taking into account the gravitational force. Solutions were found by Leibniz, Newton, Bernoulli himself and l'Hôpital. Particularly Leibniz and Newton's subsequent works on the field of variational analysis contributed to the early growth of optimal control theory. The main developments in this field started in the 1950th with the works of Pontryagin and collaborators and since then, a large research effort has been put into this problem class.

A continuous optimal control problem is described in terms of states which evolve in time from a given initial state according to certain laws of dynamics ranging from ordinary differential equations (ODEs), partial differential equations (PDEs), stochastic differential equations and discrete difference equations. We consider the dynamical system to be in the form of an ODE system. The evolution of the states is influenced according to some performance criterion by a control function taken from a set of admissible controls. Together with some constraints on the control and the state functions, the problem becomes a constrained nonlinear, infinite-dimensional optimization problem with the goal to determine the optimal control function and state trajectories.

Consider a problem of kind:

$$\text{Min } J(u) = \int_{t_0}^{t_f} f(t, x(t), u(t)) dt, \quad x \in C^1(t_0, t_f) \quad (4.1)$$

with $x(t_0) = x_0$. If we fix $x(t)$ we can introduce a new variable $z(t)$ such that

$$\dot{z}(t) = f(t, x(t), u(t)), \quad z(t_0) = 0.$$

In this way, we have

$$z(t_f) = z(t_f) - z(t_0) = \int_{t_0}^{t_f} \dot{z}(t) dt = \int_{t_0}^{t_f} f(t, x(t), u(t)) dt,$$

and the previous problems can be restated as a terminal control problem:

$$\text{Min } z(t_f) \quad \text{subject to } \dot{z}(t) = f(t, x(t), u(t)), \quad (4.2)$$

with initial conditions $x(t_0) = x_0$ and $z(t_0) = 0$. We have to find the control $u(t)$ at which $z(t_f)$ attains its minimal value. This formulation does not involve the operation of integration, and it is well known that the solution of a Cauchy problem for a system of ordinary differential equations (ODE) is less computationally expensive than the solution of the corresponding integral equations formulation. Although problem (4.2) is equivalent to problem (4.1), it has extended the class of problems that we can treat, since it is not a problem of calculus of variations. If we consider (4.2) from a vectorial point of view, we can assume $t \in \mathbb{R}^n, f(t, x, u) : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}$ and the ODE from \mathbb{R}^{2n+1} to \mathbb{R}^m . The problem is nondegenerate if $m < n$, otherwise the minimization is done over a discrete or an empty set of trajectories. For nondegenerate problems, the condition $m < n$ in the n -dimensional space of variables u , defines an $(n - m)$ dimensional manifold M , and in a neighbourhood of a point in M we can introduce coordinates $U \in \mathbb{R}^{n-m}$ and parametrize the manifold, the system of ODE depending on (t, x, u) . There are three types of optimal control problems, they differ apparently in the formulation of the functional to be optimized.

1. The first standard form for an OCP is given by the formulation of Mayer; the functional is not an integral but a function M that depends in general from the dependent variable t and the final point of the t -domain. Often this is useful to be intended as a problem of optimizing the final time, such as in the time-optimal OCPs. The objective function is called pay off function and is constrained by a set of differential equations; in general ODE, but often we encounter also differential algebraic equations (DAE). The standard formulation is:

$$\begin{aligned}
\min_{u \in U} J(u) &= M(t_f, x(t_f)) \\
\dot{x}(t) &= g(t, x(t), u(t)) \quad t \in (t_0, t_f) \\
x(t_0) &= x_0.
\end{aligned} \tag{4.3}$$

The set U represents a general class of function available for the control, for example the control can be C^1 or \hat{C}^1 , other possibilities are the piecewise constant functions. U can also contain limitations for the control, a classical case is $|u| \leq 1$.

2. The second standard form for an OCP is given by the formulation of Lagrange; we have the objective functional in (pure) integral form. The previous considerations for the Mayer problem still hold. The standard formulation is:

$$\begin{aligned}
\min_{u \in U} J(u) &= \int_{t_0}^{t_f} f(t, x(t), u(t)) dt, \\
\dot{x}(t) &= g(t, x(t), u(t)) \quad t \in (t_0, t_f) \\
x(t_0) &= x_0.
\end{aligned} \tag{4.4}$$

3. The third standard form for an OCP is given by the formulation of Bolza, which consider a linear combination of the problems of Mayer and Lagrange.

$$\begin{aligned}
\min_{u \in U} J(u) &= M(t_f, x(t_f)) + \int_{t_0}^{t_f} f(t, x, u) dt, \\
x'(t) &= g(t, x(t), u(t)), \quad t \in (t_0, t_f) \\
x(t_0) &= x_0.
\end{aligned} \tag{4.5}$$

As before, the considerations done for the Mayer problem hold.

Even if the Bolza problem looks more general than the other two, we show next that the three formulations are equivalent. It is clear that problems (4.3) and (4.4) are particular cases of (4.5), hence we have to show how (4.5) becomes (4.4) and how (4.4) becomes (4.3). To change from Bolza to Lagrange, add a new component to the $x \in \mathbb{R}^n$, $x_{n+1}(t) = M(t, x(t))$ According to this notation, the problem of Bolza becomes

$$\min_{u \in U} J(u) = \int_{t_0}^{t_f} f(t, x, u) + \dot{x}_{n+1}(t) dt \quad t \in [t_0, t_f]$$

$$\begin{pmatrix} \dot{x} \\ \dot{x}_{n+1} \end{pmatrix} = \begin{pmatrix} g(t, x, u) \\ \frac{d}{dt} M(t_0, x_0) \end{pmatrix}$$

$$\begin{pmatrix} x(t_0) \\ x_{n+1}(t_0) \end{pmatrix} = \begin{pmatrix} x_0 \\ M(t, x(t)) \end{pmatrix},$$

which is a problem of Lagrange.

To transform (4.4) into a Mayer (4.3) or from Lagrange to Mayer, consider a new variable x_{n+1} defined as $\dot{x}_{n+1}(t) = f(t, x, u)$, $x_{n+1}(t_0) = 0$, with the initial condition. Hence the problem of Lagrange becomes:

$$\min_{u \in U} J(u) = x_{n+1}(t_f); \quad t \in [t_0, t_f]$$

$$(\dot{x}, \dot{x}_{n+1}) = (g(t, x, u), f(t, x, u))$$

$$(x(t_0), x_{n+1}(t_0)) = (x_0, 0)$$

which is a problem of Mayer.

To change from Mayer to Lagrange, Consider a new variable x_{n+1} defined as $\dot{x}_{n+1}(t) = 0$ with the condition that $x_{n+1}(t) = \frac{M(t, x(t_f))}{t_f - t_0}$. Hence the problem of Mayer becomes:

$$\min_{u \in U} J(u) = \int_{t_0}^{t_f} x_{n+1}(t) dt \quad t \in [t_0, t_f]$$

$$(\dot{x}, \dot{x}_{n+1}) = (g(t, x, u), 0)$$

$$(x(t_0), x_{n+1}(t_0)) = \left(x_0, \frac{M(t, x(t_f))}{t_f - t_0} \right),$$

which is a problem of Lagrange. But Mayer and Bolza problems are not used in this project and thus only described briefly.

4.2. Shooting Methods

There are three main algorithms employed in the application of a direct method, the first is the shooting method (single and multiple) which results in small NLP problems; the second is the pseudospectral method (medium sized problem); the third is the collocation method, which is the most accurate at the price of a very large NLP. The main advantage of the direct methods is that NLPs are widely studied and a plethora of state of art solution algorithms is available. Moreover it is easier to treat inequality constraints because they have their natural equivalent form in the associated NLP problem. The principal disadvantage is that direct methods produce only suboptimal or approximate solutions. Nowadays they are very popular because they are easy to understand and apply (no calculus of variations needed), they are also robust.

4.2.1. Direct Single Shooting Method

In contrast to indirect approaches, direct methods are based on a discretization of the infinite-dimensional OCP into a finite-dimensional nonlinear optimization problem. In direct single shooting one chooses to regard the state trajectory as a dependent value of the control $u(\cdot)$ and to solve an initial value problem (IVP) using an ODE or DAE solver to compute it. To this end, a finite-dimensional discretization

$$q = (q_0, q_1, \dots, q_{m-1}) \in \mathbb{R}^{m \times n^q},$$

of the control trajectory $u(\cdot)$ on \mathcal{T} , $[t_0, t_m]$ is chosen for the solution of the IVP

$$\dot{x}(t) = f(t, x(t), u(t)), \quad t \in \mathcal{T}$$

$$x(t_0) = s_0,$$

e.g. by using piecewise constant functions ($n^q = n^u$)

$$u(t) = q_i, \quad t \in [t_i, t_{i+1}] \subset \mathcal{T}, \quad 0 \leq i \leq m.$$

from this approach a Nonlinear Program (NLP) in the $n_x + mn_q$ unknowns

$$V \stackrel{def}{=} (s_0, q_0, q_1, \dots, q_{m-1})$$

is obtained which can be solved using e.g. the sequential quadratic programming techniques presented. Control and path constraints frequently are discretized and enforced on the control discretization grid only, or may be included in the objective using penalty terms.

The direct single shooting method suffers from a number of drawbacks. As only the initial state and the control variables enter the NLP, initialization of the state trajectory using prior knowledge about the process is not possible. Further, for a chosen initialization of x_0 and the controls u , the IVP's solution need not even exist, e.g. due to a singularity in time. Even if it exists, it possibly cannot be computed numerically due to propagation of errors over the course of the integration. Typically, to guarantee existence of a numerical solution of highly nonlinear or unstable ODEs, an initial guess of these variables is required that is already very close to the true optimal solution. Such a guess may of course be hard to obtain.

The convergence speed of the NLP solver is effectively governed by the amount of nonlinearity present in the ODE system, and single shooting methods cannot improve upon this situation. Finally, even well-conditioned IVPs may induce unstable Boundary Value Problems (BVPs), and a small step in the initial value x_0 may result in a large step in the ODE's solution $x(\cdot)$ or induce large violations of constraints. This behavior is a challenge for derivative-based NLP methods. Still, direct single shooting is often used in practice as the idea is easy to grasp, the implementation is straightforward, and the resulting NLP has a small number of unknowns only.

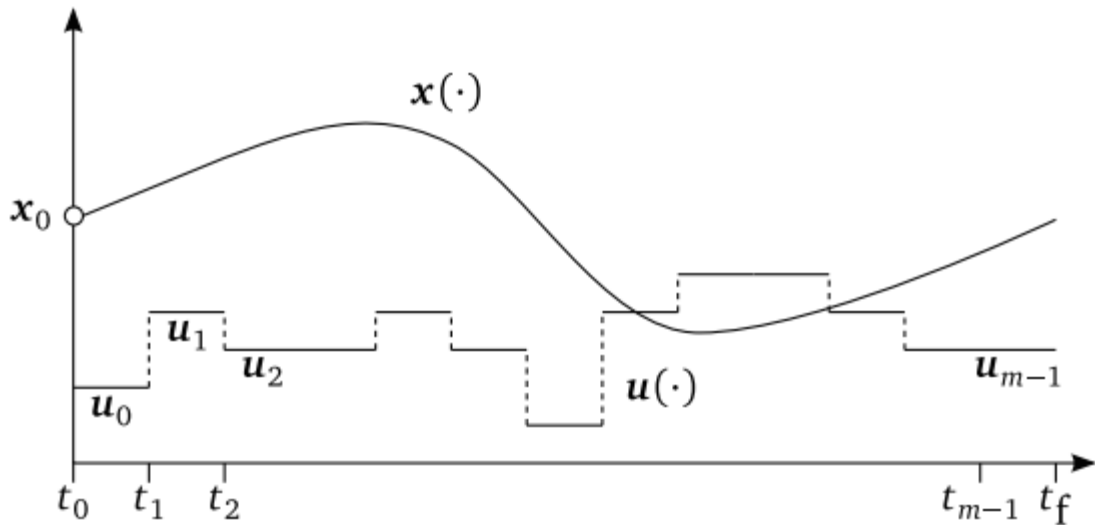


Figure 4.1. Illustration of the direct single shooting discretization applied to the optimal control problem.

4.2.2. Direct Multiple Shooting Method

We consider the numerical solution of the following class of OCPs that have been introduced in the previous sections together with various possible extensions:

$$\text{Minimize } J(u) = m(t_f, x(t_f)) + \int_{t_0}^{t_f} L(t, x(t), u(t)) dt \quad (4.6a)$$

$$\text{Subject to } \dot{x}(t) = f(t, x(t), u(t)), \quad t \in \mathcal{T} \quad (4.6b)$$

$$0 = r_i^{eq}(t_i, x(t_i)), \quad \{t_i\} 0 \leq i \leq n^r \subset \mathcal{T}, \quad (4.6c)$$

$$0 \leq r_i^{in}(t_i, x(t_i)), \quad \{t_i\} 0 \leq i \leq n^r \subset \mathcal{T} \quad (4.6d)$$

$$0 \leq c(t, x(t), u(t)), \quad t \in \mathcal{T}, \quad (4.6e)$$

In which we minimize a Bolza type objective function (4.1a) of a dynamic process $x(\cdot)$ defined on the horizon $\mathcal{T} \stackrel{\text{def}}{=} [t_0, t_f] \subset \mathbb{R}$ in terms of an ODE system with right hand side function f (4.1b). The process is controlled by a control trajectory $u(\cdot)$ subject to optimization. It shall satisfy certain inequality path constraints $c(\cdot)$ (4.1e) as well as equality and inequality point constraints

$r_i(\cdot)$ (4.1c, 4.1d) imposed on a constraint grid $\{t_i\} \subset \mathcal{T}$ of n^r points on \mathcal{T} . We expect all functions to be twice continuously differentiable with respect to the unknowns of problem (4.1).

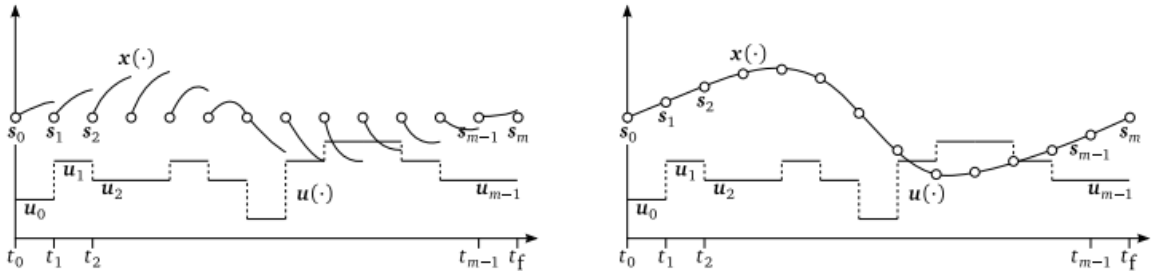


Figure 4.2. Illustration of the direct multiple shooting discretization applied to the optimal control problem.

Figure 4.2 on the left, all shooting nodes were initialized identically and the solution of the m IVPs violates the matching conditions. On the right, the matching conditions are fulfilled after convergence of the NLP solver. More generally, the IVPs (4.6a-4.6b) together with the matching conditions (4.6c) may be replaced by a sufficiently smooth but otherwise arbitrarily chosen mapping

$$0 = j_i(t_i, t_{i+1}, s_i, s_{i+1}, q_i, q_{i+1}), \quad 0 \leq i \leq m-1.$$

In particular, the Jacobians of the mappings j_i with respect to the subsequent state s_i or the control q_i need not necessarily be regular. This effectively increases the number of degrees of freedom of the discretized optimal control problem. In case of singularity of j_i with respect s_i , we allow for discontinuities of the state trajectory $x(\cdot)$, while singularity of j_i with respect to q_{i+1} is just a generalization of choosing a discontinuous control discretization. The resulting vector of $(m+1)n^x mn^q$ unknowns, assuming n_{ij}^q to be identical for all nodes and controls, is denoted by $V \stackrel{def}{=} [s_0 \ q_0 \ \cdots \ s_{m-1} \ q_{m-1} \ s_m]$.

4.3. Errors

Approximation is a central concept in almost all the uses of mathematics. The main purpose of numerical analysis and scientific computing is to develop efficient and accurate methods to compute approximations to quantities that are difficult or impossible to obtain by analytic means. Numerical results are affected by many types of errors. Some of them are:

Inherent error: Errors which are already present in the statement of the problem before its solution.

Rounding error: It is errors that arise from the process of rounding off the numbers during computation. It occurs whenever an irrational number, for example, is shortened (“rounded off”) to a fixed number of digits, or when a decimal fraction is converted to the binary form used in the computer. The limitation of floating-point numbers in a computer leads at times to a loss of information that, depending on the context, may or may not be important.

Truncation error: These are errors committed when a limiting process is truncated (broken off) before one has come to the limiting value. A truncation error occurs, for example, when an infinite series is broken off after a finite number of terms, or when a derivative is approximated with a difference quotient (although in this case the term **discretization error** is better). Another example is when a nonlinear function is approximated with a linear function. We use some examples to make the idea more precise. Let we expand $Y(x_{n+1})$ by using Taylor series about x_n as

$$Y(x_{n+1}) = Y(x_n) + Y'(x_n)h + \frac{h^2}{2}Y''(\varphi_n), \quad x_n < \varphi_n < x_{n+1}.$$

To obtain the Euler method, we drop the error term. The dropped term $\frac{h^2}{2}Y''(\varphi_n)$ is called the **truncation error** at x_{n+1} .

Absolute, Relative and Percentage errors: If x is the true value of a quantity and x_a is its approximate value, then the absolute error defined as $E_a = |x - x_a|$. The relative error E_r is defined by $E_r = \left| \frac{x - x_a}{x} \right|$, and the percentage error E_p is defined by

$$E_p = 100E_r = 100 \left| \frac{x - x_a}{x} \right|.$$

For example, let $x = e = 2.7182818 \dots$ and $x_a = \frac{19}{7} = 2.7142857 \dots$ then, $E_a = |x - x_a| = 0.003996 \dots$, $E_r = \left| \frac{x - x_a}{x} \right| = 0.00147 \dots$, and $E_p = 100E_r = 0.147$. In this project work we used only the absolute error, but the others were to describing only.

Note: Stopping (tolerance) criteria is the iteration procedure when the magnitudes of the of the differences between the two successive iterates of all the variables are smaller than a given accuracy or error tolerance or an bound ε , that is,

$$\left| x_i^{(w+1)} - x_i^{(w)} \right| \leq \varepsilon, \forall i \text{ and } w = 0, 1, 2, \dots$$

For example, if we require two decimal places of accuracy, then we iterate until $\left| x_i^{(w+1)} - x_i^{(w)} \right| < 0.006$, for all i . If we require three decimal places of accuracy, then we iterate until $\left| x_i^{(w+1)} - x_i^{(w)} \right| < 0.0006$, for all i .

Definition 4.2. An optimal control problem is constrained optimization problems with a dynamical system as constrained. Let us consider the following initial value problem (IVP).

$$\dot{x}(t) = f(x(t), u(t), t), \quad x(0) = x_0, \quad t \in [0, T] \quad (4.7)$$

where $x(t) \in \mathbb{R}^n$ are the states of the dynamical system and $u(t) \in U \subset \mathbb{R}^m$ controls. If the controls vector $u(t)$ known and sufficiently regular (e.g. piecewise continuous) and $f(x, u, t)$ is regular enough as a function of (x, u, t) (e.g. continuous in (x, u, t) , and Lipschitz in x) then IVP has an unique solution that depends on the control history $u(t)$. In more complex problems, the initial condition $x(0) = x_0$ may be replaced with a general boundary condition $b(x(0), x(T)) = 0$. Furthermore, path

constraints $c(x(t), u(t), t) \geq 0$ may also be present. However, here to simplify the explanation and comparison among the three families of solution methods for OCP. Let us now define a feasible control a control history $\bar{u}(t) \in U$, which determines a function $\bar{x}(t)$ (i.e. a history of states) for the dynamical system (4.7) that satisfies the set of boundary conditions, and when defined-the path constraints. The pair $(\bar{u}(t), \bar{x}(t))$ is called feasible manoeuvre.

Now, let's introduce a performance functional index:

$$J[u] = M(x(T)) + \int_0^T l(x(t), u(t), t) dt \quad (4.8)$$

this functional $J[u]$ evaluates to a scalar, where function $x(t)$ is the solution of the ODE (4.7) given a control manoeuvre \bar{u} . The term $M(\cdot)$ is called final cost or Mayer's term, while $l(\cdot)$ is named running cost or Lagrange's term. Under these definitions, the solution of an optimal control problem is a feasible manoeuvre, or pair of control history u and state x , which minimizes the performance index $J[u]$ when the dynamical system transits from the initial state to the final one satisfying the path constraints.

Definition 4.3. (Piecewise Continuous Functions). A real-valued function $u(t)$, $t_0 \leq t \leq t_f$, is said to be piecewise continuous, denoted $u \in \hat{C}[t_0, t_f]$, if there is a finite (irreducible) partition $t_0 = \theta_0 < \theta_1 < \dots < \theta_N < \theta_{N+1} = t_f$ such that u may be regarded as a function in $C[\theta_K, \theta_{K+1}]$ for each $K = 0, 1, \dots, N$. That is, the class $\hat{C}[t_0, t_f]^{n_u}$ of n_u -dimensional vector-valued analogue of $\hat{C}[t_0, t_f]$, consists of those controls u with components $u_j \in \hat{C}[t_0, t_f]$, $j = 1, \dots, N$. The discontinuities of one such control are by definition those of any of its components u_j . Note that piecewise continuous controls correspond to the assumption of inertia-less controllers, since the values of $u(t)$ may jump instantaneously when a discontinuity is met. This class of controls appears to be the most interesting for the practical applications of the theory, although existence of an optimal control is not guaranteed in general. The specification of the control region together with a class of controls leads naturally to the definition of an admissible control.

Definition 4.4. (Admissible Control). A piecewise continuous control $u(\cdot)$, defined on some time interval $t_0 \leq t \leq t_f$, with range in the control region U , $u(t) \in U, \forall t \in [t_0, t_f]$, is said to be an admissible control. We shall denote by $U[t_0, t_f]$ the class of admissible controls on $[t_0, t_f]$. It follows from definition 4.3 that every admissible control $u \in U[t_0, t_f]$ is bounded.

Theorem 4.1. (Sufficient Conditions). Consider the problem to minimize the functional

$$J(y) = \int_a^b F(x, y, y') dx$$

$$D := \{y \in C^1[a; b] : y(a) = y_a; y(b) = y_b\}.$$

Suppose that the Lagrangian $F(x, w, z)$ is continuously differentiable and [strictly] jointly convex in $(w; z)$. If y^* is stationary function for the Lagrangian F , then y^* is also a [strict] global minimizer for J on D .

Proof. The integrand $F(x, w, z)$ being continuously differentiable and jointly convex in $(w; z)$, we have for an arbitrary D -admissible trajectory y :

$$\begin{aligned} \int_a^b F[y] - F[y^*] dx &\geq \int_a^b (y - y^*)F_y[y^*] + (y' - y^{* \prime})F_{y'}[y^*] dx \quad (\text{by convexity}) \\ &= \int_a^b (y - y^*) \left[F_y[y^*] - \frac{d}{dx} F_{y'}[y^*] \right] dx + \left[(y - y^*)F_{y'}[y^*] \right]_a^b \end{aligned}$$

with the usual compressed notation. Clearly, the first term is equal to zero, for y^* is a solution to the Euler equation; and the second term is also equal to zero, since y is D -admissible, i.e., $y(a) = y^*(a) = y_a$ and $y(b) = y^*(b) = y_b$

Therefore, $\int_a^b F[y] dx \geq \int_a^b F[y^*] dx$ or $\int_a^b F(x, y, y') dx \geq \int_a^b F(x, y^*, y^{* \prime}) dx$.

Thus $J(y) \geq J(y^*)$ for each admissible trajectory. This complete the proof

Theorem 4.2. (Principle of Optimality).

Let $u^* \in \widehat{\mathcal{C}}[t_0, t_f]^{n_u}$ be an optimal control for the problem to

$$\text{minimize: } \int_{t_0}^{t_f} l(t, x(t), u(t)) dt, \quad (5.17)$$

$$\text{Subject to } \dot{x}(t) = f(t, x(t), u(t)), \quad x(t_0) = x_0, \quad (5.18)$$

and let $x^* \in \widehat{\mathcal{C}}[t_0, t_f]^{n_x}$ denote the corresponding optimal response. Then, for any $t_1 \in [t_0, t_f]$, the restriction $u^*, t_1 \leq t \leq t_f$, is an optimal control for the problem to

$$\text{minimize } J = \int_{t_0}^{t_f} l(t, x(t), u(t)) dt, \quad (5.19)$$

$$\text{Subject to } \dot{x}(t) = f(t, x(t), u(t)), \quad x(t_1) = x_1, \quad (5.20)$$

Proof. Let $v(x_0, t_0)$ denote the minimum values of the optimal control problem (5.17, 5.18). Clearly, $v(x_0, t_0) = \int_{t_0}^{t_f} l(t, x^*(t), u^*(t)) dt + \int_{t_1}^{t_f} l(t, x^*(t), u^*(t)) dt$.

By contradiction, suppose that the restriction is $u^*, t_1 \leq t \leq t_f$ not optimal for the problem (5.19, 5.20). Then, there exists a (feasible) control $\bar{u}, t_1 \leq t \leq t_f$ that imparts to the functional (5.18) the value

$$\int_{t_1}^{t_f} l(t, \bar{x}(t), \bar{u}(t)) dt \leq \int_{t_1}^{t_f} l(t, x^*(t), u^*(t)) dt.$$

Further, by joining $u^*, t_0 \leq t \leq t_1$ and $\bar{u}, t_1 \leq t \leq t_f$, one obtains a piecewise continuous control that is feasible and satisfies

$$\int_{t_0}^{t_1} l(t, x^*(t), u^*(t)) dt < \int_{t_1}^{t_f} l(t, \bar{x}(t), \bar{u}(t)) dt$$

hence contradicting the optimality of $u^*, t_0 \leq t \leq t_f$, for the problem (5.17, 5.18).

This complete the proof

5. INDIRECT AND DIRECT METHODS

5.1. Direct Method

A different approach to OCPs is given by the direct methods which follow the philosophy of first discretize, then optimize and are somehow the opposite of the indirect methods. Here the state and the control variables are approximated by polynomial interpolation; the target functional itself is approximated by a cost function. Hence the problem is discretized on a mesh, and the optimization variables become the unknowns of a general nonlinear programming problem (Marco Frego, 2014).

Definition 5.1. Discretization: Discretization is a process in which the time interval $t \in [t_0, t_f]$ is to be divided into an equal time segments, $t_0 \leq t_1 \leq t_2 \leq \dots \leq t_i = t_f$. As a result, depending on the discretization technique, the variables are sampled at each time point. Basically, there are two discretization techniques used in optimal control problem: Control-state discretization and control discretization (Abu Haya, 2011).

1. **Control-state discretization:** Apply this method to discretize both control variables $u(t)$ and state variables $x(t)$. As a result, the vector which contains a sequence of unknown state variables and control variables, will be produced.

2. **Control discretization:** This approach is to discretize the control variables $u(t)$ only. In order to get the state variables $x(t)$, it is necessary to integrate the system of state equations. This would produce state variables that are a function of control variables that are a function of the control variables. An advantage of this technique over Control-state discretization is that the resulted system is lower in dimension.

In order to obtain a computationally tractable representation of the infinite dimensional control trajectory $u(\cdot)$ we introduce a discretization of $u(\cdot)$ on the control horizon \mathcal{T} by partitioning \mathcal{T} into m not necessarily equidistant intervals $t_0 < t_1 < \dots < t_m = t_f$ that define the shooting grid $\{t_i\}$. On each interval $[t_i, t_{i+1}]$, $0 \leq i \leq$

$m - 1$, we choose base functions $b_{ij}: \mathcal{T} \times \mathbb{R}^{n_{ij}^q} \rightarrow \mathbb{R}$ for each component $1 \leq j \leq n^u$ of the control trajectory. To ensure reparability of the discretization the b_{ij} shall have local support, and are parameterized by a vector of finitely many control parameters $q_{ij} \in \mathbb{R}^{n_{ij}^q}$. Popular choices for the base functions include piecewise constant controls ($n_{ij}^q = 1$).

$$b_{ij}: [t_i, t_{i+1}] \times \mathbb{R}^{n_{ij}^q} \rightarrow \mathbb{R}, \quad (t, q_{ij}) \mapsto q_{ij},$$

piecewise linear controls ($n_q^{ij} = 2$)

$$b_{ij}: [t_i, t_{i+1}] \times \mathbb{R}^{n_{ij}^q} \rightarrow \mathbb{R}, \quad (t, q_{ij}) \mapsto \frac{t_{i+1}-t}{t_{i+1}-t_i} q_{ij_1} + \frac{t-t_i}{t_{i+1}-t_i} q_{ij_2},$$

or piecewise cubic spline controls ($n_q^{ij} = 4$)

$$b_{ij}: [t_i, t_{i+1}] \times \mathbb{R}^{n_{ij}^q} \rightarrow \mathbb{R}, \quad (t, q_{ij}) \mapsto \sum_{k=1}^4 q_{ij_k} \beta_k \left(\frac{t-t_i}{t_{i+1}-t_i} \right)^{k-1}$$

with appropriately chosen spline function coefficients β . Evidently, the type of discretization may be chosen differently for each of the n^u components of the control trajectory. For the ease of notation we introduce an additional discretized control $b_m(t_m, q_m)$ in the last point t_m of the shooting grid $t_m \{t_i\}$, which shall be implicitly fixed to the final control value of the previous shooting interval,

$$b_m(t_m, q_m) \stackrel{\text{def}}{=} b_{m-1}(t_m, q_{m-1})$$

For certain choices of the control discretization, e.g. for piecewise linear controls, continuity of the discretized control trajectory may be desired. This can be enforced by imposing additional control continuity conditions for the trajectory $u_j(t)$ in all points of the control discretization grid $\{t_i\}$,

$$0 = b_{ij}(t_{i+1}, q_i) - b_{i+1,j}(t_{i+1}, q_{i+1}), \quad 0 \leq i \leq m - 1.$$

The choice of the control discretization obviously affects the quality of the discretized OCP's solution approximation that of the infinite-dimensional one.

Definition 5.2. Parameterization: Parameterization technique is a process in which a functions or variables are approximated using known functions with unknown coefficient. Parameterization can be employed by one of the three forms:

1. Control Parameterization: In this technique, only the control variable $u(t)$ approximated by a finite length series of known function with unknown coefficient.
2. Control-State Parameterization: Using this technique, both control variables $u(t)$ and state variables $x(t)$ are approximated by a finite length series of known function with unknown coefficient.
3. State Parameterization: In this technique, only the state variable $x(t)$ are approximated by a finite length series of known functions with unknown coefficient. In addition to the control discretization, and notably different from single shooting described in section 4.2.1, we also introduce a parameterization of the state trajectory $x(\cdot)$ on the shooting grid $\{t_i\}$ that yields m IVPs with initial values $s_i \in \mathbb{R}^{n_x}$ on the intervals $[t_i, t_{i+1}]$ of the horizon \mathcal{T} ,

$$\begin{aligned} \dot{x}_1(t) &= f(t, t_i(t), b_i(t, q_i)), \quad t \in [t_i, t_{i+1}], \quad 0 \leq i \leq m-1, \\ \dot{x}_1(t) &= s_i. \end{aligned} \tag{5.1}$$

In order to ensure continuity of the obtained trajectory $x(\cdot)$, on the whole of \mathcal{T} , we introduce $m-1$ additional matching conditions

$$0 = x_i(t_{i+1}; t_i, s_i, q_i) - s_{i+1}, \quad 0 \leq i \leq m-1. \tag{5.2}$$

here, the expression $x_i(t_{i+1}; t_i, s_i, q_i)$ denotes the final value $x(t_{i+1})$ obtained as the solution of the IVP (5.1) on the interval $[t_i, t_{i+1}]$ when starting in the initial value $u(t) = b_i(t, q_i)$ and applying the control trajectory on $[t_i, t_{i+1}]$. The evaluation of the residual of constraint (5.2) thus requires the solution of an IVP by an appropriate numerical method.

In the direct approach, discretization happens first. There are two common frameworks available for the discretization procedure in the direct optimization

methods, complete parameterization (Biegler, 2007) and control parameterization (Goh and Teo, 1988). Either of them is applicable to convert the optimal control problem into a nonlinear programming problem which is solvable by a suitable optimization theoretic. However, there are also advantageous and disadvantageous in applying each one. The idea of complete parameterization is to discretize the whole variables of a control problem. As a result, the original problem is reduced to a problem of identifying the best parameters of the approximated variables. One advantage of this formulation is to eliminate the online need for the solution of an initial value problem; however this is at the cost of a very large scale optimization problem need to be solved. In this project work we used complete parameterization.

Definition 5.3. Krik (1970) Optimum of a function: A function $f(t)$ is said to have a relative optimum at the point t^* if there is a positive parameter ε such that for all points t in a domain D that satisfy $|t - t^*| < \varepsilon$, the increment of $f(t)$ has the same sign (positive or negative).

In other words, if

$$\Delta f = f(t) - f(t^*) \geq 0,$$

then, $f(t^*)$ is a relative local minimum. On the other hand, if

$$\Delta f = f(t) - f(t^*) \leq 0,$$

then, $f(t^*)$ is a relative local maximum.

Note: We begin with an example of finding the extrema of a function under a condition (or constraint). We solve this example with two methods, first by direct method and then by indirect method. Let us note that we consider this simple example only to illustrate some basic concepts associated with direct method (Stage and White, 1977).

Example 5.1. A manufacturer wants to maximize the volume of the material stored in a circular tank subject to the condition that the material used for the tank is limited

(constant). Thus, for a constant thickness of the material, the manufacturer wants to minimize the volume of the material used and hence part of the cost for the tank.

Solution: If a fixed metal thickness is assumed, this condition implies that the cross-sectional area of the tank material is constant. Let d and h be the diameter and the height of the circular tank. Then the volume contained by the tank is

$$V(d, h) = \pi d^2 h / 4 \quad (5.3)$$

$$A(d, h) = 2\pi d^2 / 4 + \pi dh = A_0. \quad (5.4)$$

Our intent is to maximize $V(d, h)$ keeping $A(d, h) = A_0$, where A_0 is a given constant. We discuss two methods: first one is called the direct method using simple calculus and the second one is called indirect method using the Lagrange multiplier.

In solving for the optimum value directly, we eliminate one of the variables, say h , from the volume relation (5.3) using the area relation (5.4). By doing so, the condition is embedded in the original function to be extremized. From (5.4),

$$h = \frac{A_0 - \pi d^2 / 2}{\pi d}. \quad (5.5)$$

Using the relation (4.5) for height in the relation (5.3) for volume

$$V(d) = A_0 d / 4 - \pi d^3 / 8. \quad (5.6)$$

Now, to find the extrema of this simple calculus problem, we differentiate (5.6) w.r.t. d and set it to zero to get

$$\frac{A_0}{4} - 3/8 \pi d^2 = 0 \quad (5.7)$$

Solving, we get the optimal value of d as

$$d^* = \sqrt{\frac{2A_0}{3\pi}}. \quad (5.8)$$

By demanding that as per the definition 5.3 for optimum of a function, the second derivative of V w.r.t. d in (5.6) be negative for maximum, we can easily see that the

positive value of the square root function corresponds to the maximum value of the function. Substituting the optimal value of the diameter (5.8) in the original cross-sectional area given by (5.4), and solving for the optimum h^* , we get

$$h^* = \sqrt{\frac{2A_0}{3\pi}}. \quad (5.9)$$

Thus, we see from (5.8) and (5.9) that the volume stored by a tank is maximized if the height of the tank is made equal to its diameter.

Example 5.2. (PROPT: BENCHMARK 51). Consider the problem:

$$\min J(u) = \int_0^{10} (x^2(t) + u^2(t)) dt \quad (5.10)$$

$$\text{s.t. } \dot{x}(t) = -x^3(t) + u(t), \quad (5.11)$$

$$x(0) = 1, \quad x(10) = \frac{3}{2} \quad (5.12)$$

Solution: Here, we eliminate $u(t)$ between the performance index (5.10) and the ODE (5.11) to get the functional as

Note that: It is not always possible to eliminate $u(t)$ from (5.10) and (5.11) especially for higher-order systems.

$$\begin{aligned} J &= \int_0^{10} \left[x^2(t) + (\dot{x}(t) + x^3(t))^2 \right] dt \\ &= \int_0^{10} [x^6(t) + x^2(t) + \dot{x}^2(t) + 2x^3(t)\dot{x}(t)] dt \end{aligned} \quad (5.13)$$

Now, we notice that functional (5.13) absorbed the condition (5.11) with in itself, and we need to consider it as a straight forward extermination of a functional as given earlier. Thus, applying the Euler-Lagrange equation

$$\left(\frac{\partial v}{\partial x} \right)_* - \frac{d}{dt} \left(\frac{\partial v}{\partial \dot{x}} \right)_* = 0$$

to the functional (5.13), where,

$$V = x^6(t) + x^2(t) + \dot{x}^2(t) + 2x^3(t)\dot{x}(t),$$

we get

$$6x^{*5}(t) + 2x^*(t) + 6x^{*2}(t)\dot{x}(t) - \frac{d}{dt}(2\dot{x}^*(t) + 2x^3(t)) = 0$$

simplifying the above

$$\ddot{x}^*(t) - 3x^{*2}(t)\dot{x}^*(t) - 3x^{*5}(t) + 3x^{*2}(t) - x^*(t) = 0$$

$$x(0) = 1$$

$$x(10) = 3/2,$$

where our conditions $x(0) = 1$ and $x(10) = 3/2$, are specified on the boundary of the interval of interest $t \in [0,10]$. (Though our solution will typically extend beyond this interval, the most common scenario in boundary value problems is the case in which we are only interested in values of the independent variable between the specified endpoints.) The first step in solving a second (or higher) order ordinary differential equation in MATLAB is to write the equation as a first order system with $x_1(t) = x(t)$ and $x_2(t) = \dot{x}(t)$, for which we have

$$\dot{x}_1(t) = x_2(t)$$

$$\dot{x}_2(t) = 3x_1^2 x_2(t)(t) + 3x_1^5(t) - 3x_1^2(t) + x_1(t).$$

we record this system in the M-file (MATLAB code) in appendix A.

Table 5.1. Numerical results in DMS for example 5.1.

t	$x(t)$	$u(t)$	$dx(t)/dt$
0.0000	1.0000	0.7864	-0.2136
1.0522	0.8872	0.6501	-0.0482
2.8480	0.8516	0.6117	-0.0059
4.7755	0.8469	0.6067	-0.0007
6.8899	0.8464	0.6064	-0.0000
8.3068	0.8477	0.6138	0.0064
9.3560	0.8916	1.0478	0.1562

10.0000	1.5315	8.3019	4.7098
---------	--------	--------	--------

In the table (5.1) exact boundary condition it was 1.5 and estimated boundary condition 1.5315 that was make DMS does not got which we wanted boundary condition point, because there difference was 0.0315 we have seen in the figure (5.1.)

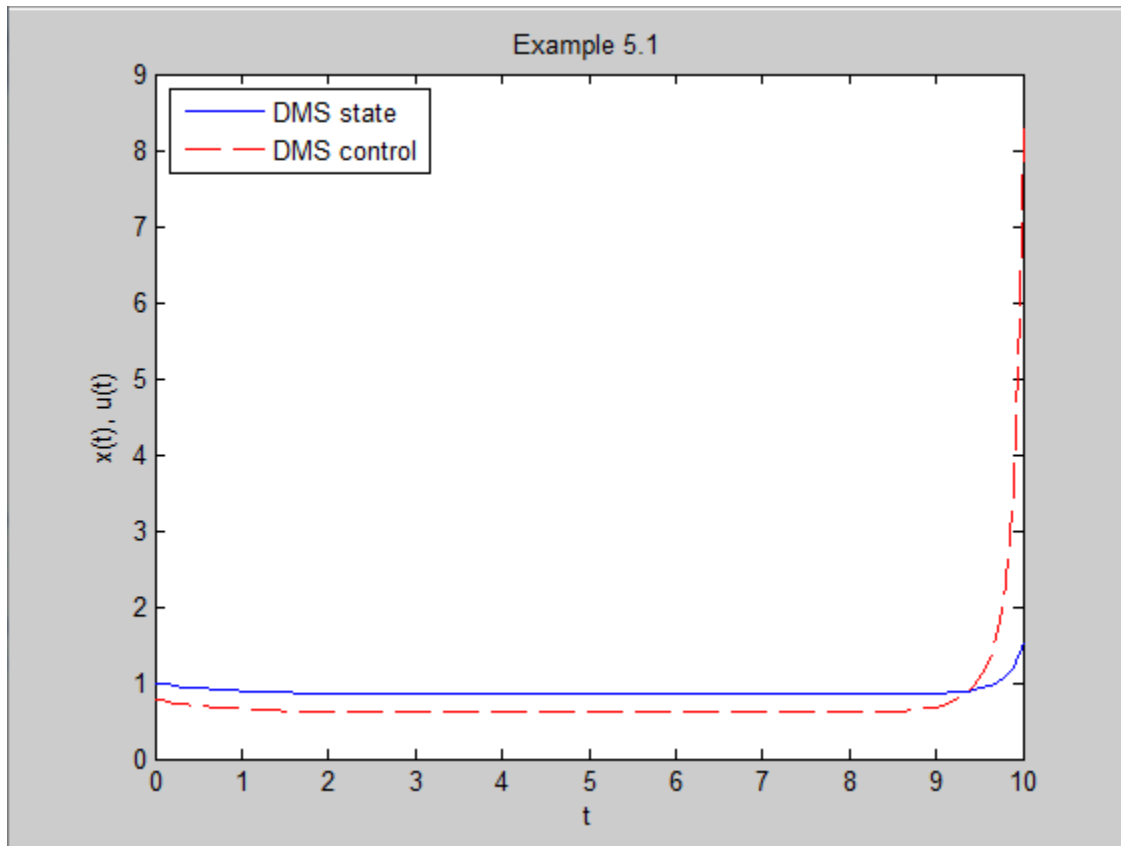


Figure 5.1. Graphs of numerical solutions for DMS in example 5.1

Example 5.2. [Semi-batch Isothermal Reactor]. Consider the optimal problem

$$\min J(u) = \int_0^3 x(t)^2 + u(t)^2 dt \quad (5.14)$$

$$\text{s.t. } \frac{dx}{dt} = (1+x)x + u, t \in [0,3], x(0) = x_0 = 0.05 \quad (5.15)$$

$$x(t) \geq -1, x(t) \leq 1$$

$$x(3) = 0, -1 \leq u(t) \leq 1 \quad (5.16)$$

Solution: Here, we eliminate $u(t)$ between the performances (5.14) and the ODE (5.15) to get the functional as:

$$\begin{aligned} J &= \int_0^3 [x(t)^2 + (\dot{x}(t) - x(t) - x(t)^2)^2] dt \\ &= \int_0^3 [x(t)^2 + \dot{x}(t)^2 - 2\dot{x}(t)x(t) - 2\dot{x}(t)x(t)^2 + x(t)^4 + 2x(t)^3 + 2x(t)^2] dt \end{aligned} \quad (5.17)$$

Now, we notice that functional (5.17) observed the condition (5.15) with in itself, and we need to consider it as a straight forward extermination of a functional as given earlier. Thus, applying the Euler-Lagrange equation

$$\left(\frac{\partial V}{\partial x}\right)_* - \frac{d}{dt} \left(\frac{\partial V}{\partial \dot{x}}\right)_* = 0$$

to the functional (5.17), where

$$V = \dot{x}^2(t) - 2\dot{x}(t)x(t) - 2\dot{x}(t)x^2(t) + x^4(t) + 2x^3(t) + 2x^2(t)$$

we get

$$\begin{aligned} -2\dot{x}(t) - 4\dot{x}(t)x(t) + 4x^3(t) + 6x^2(t) + 4x(t) - \frac{d}{dt} (2\dot{x}(t) - \\ 2x(t) - 2x^2(t)) = 0 \end{aligned} \quad (5.18)$$

Simplifying (5.18), we have

$$\ddot{x}(t) + 2\dot{x}(t)x(t) - 2x^3(t) - 3x^2(t) - 4x = 0$$

$$x(0) = 0.05$$

$$x(10) = 0,$$

where our conditions $x(0) = 0.05$ and $x(3) = 0$, are specified on the boundary of the interval of interest $t \in [0,3]$. (Though our solution will typically extend beyond this interval, the most common scenario in boundary value problems is the case in

which we are only interested in values of the independent variable between the specified endpoints.) The first step in solving a second (or higher) order ordinary differential equation in MATLAB is to write the equation as a first order system with $x_1(t) = x(t)$ and $x_2(t) = \dot{x}(t)$, for which we have

$$\dot{x}_1(t) = x_2(t)$$

$$\dot{x}_2(t) = -2x_2(t)x_1(t) + 2x_1^3(t) + 3x_1^2(t) + 4x_1.$$

we record this system in the MATLAB code appendix B

Table 5.2. Numerical results in DMS for example 5.2.

t	$x(t)$	$u(t)$	$dx(t)/dt$
0.0000	0.0500	-0.1554	-0.1029
0.3199	0.0260	-0.0795	-0.0528
0.8449	0.0090	-0.0272	-0.0181
1.3699	0.0031	-0.0094	-0.0063
1.9699	0.0009	-0.0028	-0.0019
2.4949	0.0003	-0.0007	-0.0007
3.0000	-0.0000	-0.0005	-0.0005

From this table (5.2) the approximation of state variable and control variable at time $t \in [0,3]$. Then we conclude from table (5.2) the approximation of control increases and state variable decreases when time interval increases.

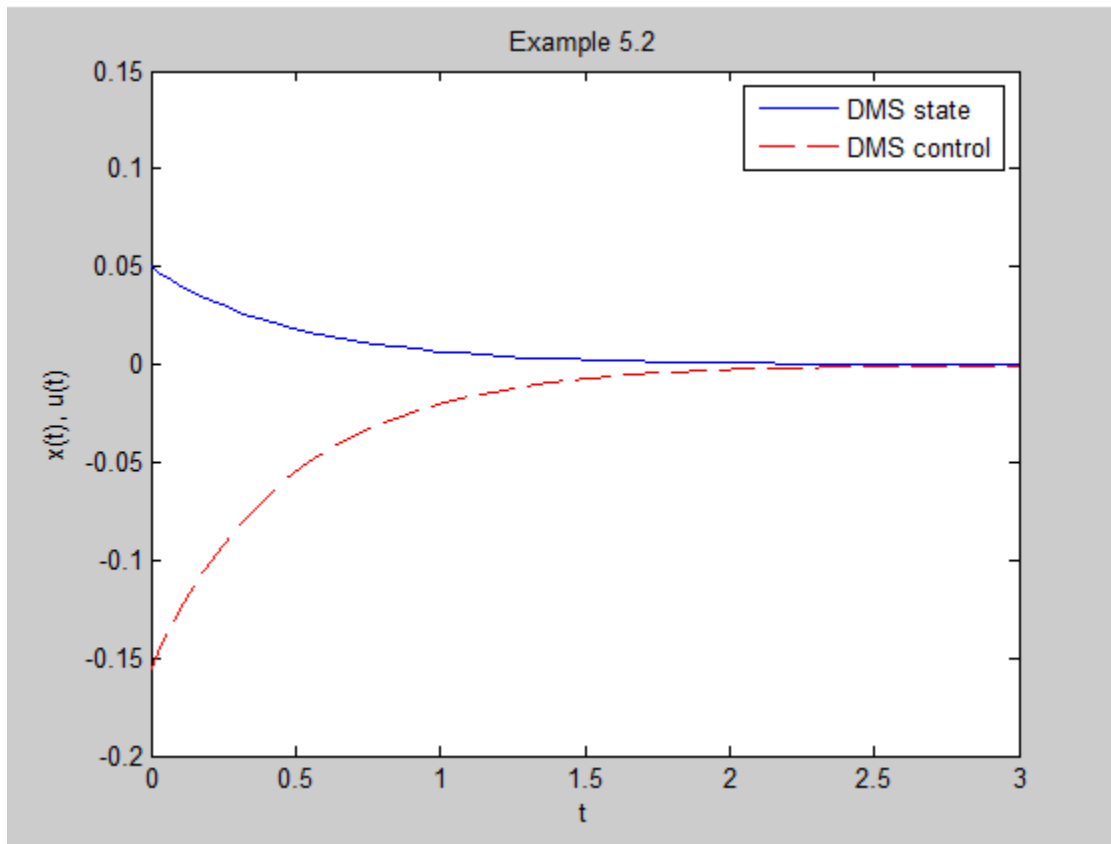


Figure 5.2. Graphs of numerical solutions for DMS in example 5.2.

5.2. Indirect Method

The indirect methods are based on the classic theory of calculus of variations and on the famous Pontryagin's Maximum (Minimum) Principle (PMP). Starting from the necessary first order optimality conditions they obtain a two-point (in general a multi-point) boundary value problem. It is derived from the first variation of the Lagrangian function associated to the optimal control problem. An equivalent derivation is possible taking derivatives of the Hamiltonian function. The boundary conditions of this BV problem are given by the initial/final condition given by the problem itself, other are yielded from the transversal condition of the adjoint variables. Of course, by the intrinsic nature of the optimal control problems, a closed form analytical solution is seldom obtained, but the indirect methods can produce it.

In presence of path constraint or inequalities it is difficult to apply the PMP to solve for an explicit formula for the control, this leads to state dependent switches. The claimed disadvantage of the indirect method is that the resulting BV problems are difficult to solve. This is not completely true, because today there are various techniques to solve systems of differential equations. It is also mandatory to analyse the computed solution, because it is only extremal but not necessary a minimum. This can be accomplished inspecting the problem (convexity, second variation, etc). The advantages are given by the underlying philosophy of first optimize, then discretize. Using indirect methods, it is possible to find the analytic solution to an OCP. However, for realistic and nonlinear problems an analytic solution becomes extremely hard to obtain.

5.2.1. Calculus of Variations

A functional $J: \Omega \rightarrow \mathbb{R}$ is a correspondence rule that maps a function of one or several variables in a certain class to a real number in \mathbb{R} . In an manner analogous to the ordinary calculus case, to find a stationary value of the functional, we require a perturbation on the function $y = f(x)$ to check whether this results in a zero rate of change of the functional. To this end, we introduce a perturbed function $\bar{f}(x)$

$$\bar{f}(x) = f(x) + \varepsilon\phi(x),$$

where $\phi(x)$ is an arbitrary function and ε is a small variable parameter tending towards zero. The difference between the original function $f(x)$ and perturbed function $\bar{f}(x)$ evaluated at x is referred to as the variation of the function $f(x)$ and is given by

$$\delta y = \bar{f}(x) - f(x) = \varepsilon\phi(x).$$

Definition 5.4. Let $\|\cdot\|$ denote the function norm. A functional $J: \Omega \rightarrow \mathbb{R}$ is said to have a local minimum at y^* if there is an $\varepsilon > 0$ such that for all functions $y \in \Omega$ that satisfy $\|y^* - y\| < \varepsilon$, the following holds

$$\Delta J = J(y) - J(y^*) \geq 0 \quad (5.19)$$

Conversely, y^* is a local maximum if

$$\Delta J = J(y) - J(y^*) \leq 0 \quad (5.20)$$

Furthermore, if the inequalities (5.19) and (5.20) hold for all y^* , the extremum is said to be global over (Naidu, 2002).

Theorem 5.1. Naidu (2002). The first order necessary condition for a functional $J(y)$ to have $J(y^*)$ as an extremum requires that the variation of the functional has to vanish at y^* for all $\delta \in \Omega$, i.e.

$$\delta J(y^*, \delta y) = 0$$

Proof. The increment of functional J at y^* is defined as:

$$\Delta J(y^*, \delta y) = J(y^* + \delta y) - J(y^*) = \delta J(y^*, \delta y) + gJ(y^*, \delta y) \cdot \|\delta y\|.$$

The $\delta J(y^*, \delta y)$ is the linear part of the increment and is referred to as the variation of $J(y^*, \delta y)$. The higher order terms are represented by $g(y^*, \delta y)$, which goes to zero faster than δy does as $\|\delta y\|$ approaches zero

$$\lim_{\|\delta y\| \rightarrow 0} g(y^*, \delta y) = 0$$

The sign of ΔJ therefore depends on the sign of δJ for sufficiently small $\|\delta y\| < \varepsilon$. Now, we assume that for some variation δy_0 , the variation of the functional is not zero,

$$\delta J(y^*, \delta y_0) \neq 0$$

For any $\alpha > 0$ such that $\|\alpha \delta y_0\| < \varepsilon$, the principle of linearity implies that

$$\delta J(y^*, \alpha \delta y_0) = \alpha \delta J(y^*, \delta y_0)$$

$$\delta J(y^*, -\alpha \delta y_0) = -\alpha \delta J(y^*, \delta y_0)$$

As mentioned earlier, for $\|\alpha \delta y_0\| < \varepsilon$, the signs of the increment and variation are the same. Therefore, $\Delta J(y^*, \alpha \delta y_0)$ can be made to have either sign. However, by definition 5.4, for y^* to be an extremal, ΔJ has to have the same sign in the neighbourhood of y^* . This is a contradiction and proves the theorem.

Lemma 5.1. Liberzon (2011) δ -process has commutative properties. More specially, the derivative of the variation is equal to the variation of the derivative:

$$\frac{d}{dx} \delta y = \delta \frac{d}{dx} y$$

and the variation of a definite integral is equal to the definite integral of the variation:

$$\delta \int_a^b f(x) dx = \int_a^b \delta f(x) dx.$$

Definition 5.5. Euler-Lagrange Equation: Was one of the earliest examples of a dynamics problem solved using variational techniques was the brachistochrone problem formulated by Johann Bernoulli (Liberzon, 2011). In problems of this kind, a function $f(x)$ needs to be found such that the definite integral J :

$$J(y) = \int_a^b L(y(x), y'(x), x) dx \quad (5.21)$$

shall be stationary subject to the boundary conditions given by

$$y(a) = y_a, = y(b) = y_b \quad (5.22)$$

A candidate function $y(x)$ that results in the stationary value of $J(y)$ should, by Theorem 5.1, result in the vanishing of the first variation of J denoted by δJ .

$$\delta J = \int_a^b L(y(x), y'(x), x) dx = 0. \quad (5.23)$$

Since only variations on the function y are meaningful as x is only a 'dummy' variable, we consider the variation of x to be zero.

$$\delta x = 0.$$

Furthermore, the variations at the boundary conditions (5.22) are zero as the function y is fixed at these points

$$\delta y|_{x=a} = 0$$

$$\delta y|_{x=b} = 0$$

Using Lemma 5.1, we can write (5.23) as

$$\delta J = \int_a^b \delta L(y(x), y'(x), x) dx$$

and using the chain rule

$$\delta J = \int_a^b \left(\frac{\partial L}{\partial y} \delta y + \frac{\partial L}{\partial y'} \delta y' \right) dx.$$

Integration by parts can be applied to the term involving $\delta y'$ to obtain

$$\int_a^b \frac{\partial L}{\partial y'} \delta y' dx = \frac{\partial L}{\partial y'} \delta y|_a^b - \int_a^b \frac{d}{dx} \frac{\partial L}{\partial y'} \delta y dx \quad (5.24)$$

$$\delta J = \int_a^b \left(\frac{\partial L}{\partial y} - \frac{d}{dx} \frac{\partial L}{\partial y'} \right) \delta y dx = 0 \quad (5.25)$$

Lemma 5.2. Gelfand and Silverman (2000). If function $h(x)$ is continuous in $[a, b]$, and if

$$\int_a^b h(x) \delta y(x) dx = 0$$

for every function δy that is continuous in $[a, b]$ such that $\delta y(a) = 0$ and $\delta y(b) = 0$, then h must be zero everywhere in $[a, b]$.

Proof. Suppose $h(x)$ is positive for some $x \in [a, b]$. Since $h(x)$ is continuous in $[a, b]$, there is some subinterval $[x_1, x_2]$ contained in $[a, b]$, where $h(x)$ is also positive. If we choose $y(x)$, which is arbitrary, to be

$$\delta y(x) = \begin{cases} (x - x_1)(x_2 - x) & x \in [x_1, x_2] \\ 0 & \text{otherwise} \end{cases}$$

then $\delta y(x)$ is continuous on $[a, b]$ and clearly positive on $[x_1, x_2]$. This implies that

$$\int_a^b h(x) \delta y(x) dx = \int_{x_1}^{x_2} h(x) (x - x_1)(x_2 - x) dx > 0,$$

which is a contradiction and proves the lemma

Invoking Lemma 5.2 on (5.25) we arrive at the celebrated Euler-Lagrange differential equation

$$\frac{\partial L}{\partial y} - \frac{d}{dx} \frac{\partial L}{\partial y'} = 0,$$

which provides the necessary and sufficient condition for the definite integral (5.21) to take a stationary value.

5.2.2. Variational Optimal Control

In a general optimal control problem we would like to find the control decision vector $u(t)$ that results in the system with state dynamics (Liberzon, 2011).

$$\dot{x}(t) = f(t, x(t), u(t)), \quad (5.26)$$

taking the minimum value of the cost functional J given by

$$J(u) = \Phi(x(t_f), t_f) + \int_{t_0}^{t_f} L(x(t), u(t), t) dt \quad (5.27)$$

here, t_0 and t_f are the initial final times, L is the Lagrange cost-to-go function and Φ is the terminal cost. General optimal control problems are typically subject to path

constraints, limits on the values of the control and also boundary conditions. For now, we omit these complications for the sake of clarity and consider a system with fixed initial states $x(t_0) = x_{t_0}$ and initial time t_0 . However, we let the final time t_f and terminal state $x(t_f)$ be free. The cost functional J (5.27) can be augmented with the constraint (5.26) using the Lagrangian method to make the problem unconstrained. To this end, we introduce the augmented cost functional J_a

$$J_a = \Phi(x(t_f), t_f) + \int_{t_0}^{t_f} L(t, x(t), u(t)) + \lambda^T(t)[f(t, x(t), u(t)) - \dot{x}(t)] dt$$

where λ is the adjoint associated with the state dynamics, and we define the Hamiltonian H to be

$$H(t, x(t), u(t), \lambda(t)) = L(t, x(t), u(t)) + \lambda^T(t)f(t, x(t), u(t)).$$

Since the first order condition for optimality requires that the variation of the functional has to be zero on an extremal, we start by expressing the variations of each term.

$$\delta J_a(u) = \delta \Phi + \delta \int_{t_0}^{t_f} (H - \lambda^T \dot{x}) dt. \quad (5.28)$$

The variation on the terminal cost term can be expressed as

$$\delta \Phi = \frac{\partial \Phi}{\partial x(t_f)} \delta x(t_f) + \frac{d\Phi}{dt_f} \delta t_f, \quad (5.29)$$

since only $x(t_f)$ and t_f are free and $x(t_0)$ and t_0 are assumed as fixed in this case.

Using the chain rule, the coefficient of δt_f (5.29) can be written as

$$\frac{d\Phi}{dt_f} = \frac{\partial \Phi}{\partial x(t_f)} \frac{dx(t)}{dt} \Big|_{t_f} + \frac{\partial \Phi}{\partial t_f} = \frac{\partial \Phi}{\partial x(t_f)} \dot{x}(t_f) + \frac{\partial \Phi}{\partial t_f}.$$

The variation in the final value of x comes from a combination of variations caused by t_f and $x(t_f)$ changes, hence

$$\delta x_f = \delta x(t_f) + \dot{x}(t_f) \delta(t_f). \quad (5.30)$$

Substituting (2.84) into (2.82) and (2.83) gives

$$\delta\Phi = \frac{\partial\Phi}{\partial x(t_f)} \delta x(t_f) + \frac{\partial\Phi}{\partial t_f} \delta t_f. \quad (5.31)$$

Applying Lemma 5.1 to the second term in (5.28) and noting that t_f is not fixed,

$$\delta \int_{t_0}^{t_f} (H - \lambda^T \dot{x}) dt = \int_{t_0}^{t_f} (\delta H - \delta \lambda^T \dot{x} - \lambda^T \delta \dot{x}) dt + (H - \lambda^T \dot{x})|_{t_f} \delta t_f \quad (5.32)$$

Variation in Hamiltonian H is

$$\delta H = \frac{\partial H}{\partial x} \delta x + \frac{\partial H}{\partial \lambda} \delta \lambda + \frac{\partial H}{\partial u} \delta u. \quad (5.33)$$

In a similar fashion to (5.24) in the derivation of the Euler-Lagrange equation, the term $-\lambda^T \delta \dot{x}$ in (5.24) can be integrated by parts to obtain

$$\int_{t_0}^{t_f} -\lambda^T \delta \dot{x} dt = -\lambda^T(t_f) \delta x(t_f) + \int_{t_0}^{t_f} \dot{\lambda}^T \delta x dt. \quad (5.34)$$

Noticing that $\delta \lambda^T \dot{x} = \dot{x}^T \delta \lambda$ and substituting (5.33) and (5.34) into (5.32) we obtain

$$\begin{aligned} \int_{t_0}^{t_f} (H - \lambda^T \dot{x}) dt &= \int_{t_0}^{t_f} \left(\frac{\partial H}{\partial x} \delta x + \frac{\partial H}{\partial \lambda} \delta \lambda + \frac{\partial H}{\partial u} \delta u - \dot{x}^T \delta \lambda + \dot{\lambda}^T \delta x \right) dt \\ &\quad + [H(t_f) - \lambda^T(t_f) \dot{x}(t_f)] \delta t_f - \lambda^T(t_f) \delta x(t_f). \end{aligned} \quad (5.35)$$

Using (5.30) in (5.35) and finally regrouping all the derived terms from (5.31) and (5.35) in (5.28) we arrive at

$$\begin{aligned} \delta J_a(u) &= \int_{t_0}^{t_f} \left[\left(\frac{\partial H}{\partial x} + \dot{\lambda}^T \right) \delta x + \left(\frac{\partial H}{\partial \lambda} - \dot{x}^T \right) \delta \lambda + \frac{\partial H}{\partial u} \delta u \right] dt \\ &\quad + \left(H(t_f) + \frac{\partial \Phi}{\partial t_f} \right) \delta t_f + \left(\frac{\partial \Phi}{\partial x(t_f)} - \lambda^T(t_f) \right) \delta x_f. \end{aligned}$$

Setting $J_a(u) = 0$ and invoking Lemma 5.2, we can obtain the well-known optimality necessary conditions

$$\dot{x}^{*T}(t) = \frac{\partial H}{\partial \lambda}(x^*(t), u^*(t), \lambda^*(t), t) \quad (5.36)$$

$$\dot{\lambda}^{*T}(t) = -\frac{\partial H}{\partial x}(x^*(t), u^*(t), \lambda^*(t), t) \quad (5.37)$$

$$0 = \frac{\partial H}{\partial u}(x^*(t), u^*(t), \lambda^*(t), t) \quad (5.38)$$

for $t_0 \leq t \leq t_f$ with transversality conditions

$$H(t_f, x^*(t_f), u^*(t_f), \lambda^*(t_f)) = -\frac{\partial \Phi}{\partial t_f}(t_f, x^*(t_f)) \quad (5.39)$$

$$\lambda^{*T}(t_f) = \frac{\partial \Phi}{\partial x(t_f)}(t_f, x^*(t_f)) \quad (5.40)$$

If we denote the Hamiltonian on the optimal trajectory to be H^* , then its time derivative is given by

$$\begin{aligned} \frac{dH^*}{dt} &= \frac{\partial H^*}{\partial x} \dot{x}^* + \frac{\partial H^*}{\partial \lambda} \dot{\lambda}^* + \frac{\partial H^*}{\partial u} \dot{u}^* + \frac{\partial H^*}{\partial t} = -\dot{\lambda}^{*T} \dot{x}^* + \dot{x}^{*T} \dot{\lambda}^* + \frac{\partial H^*}{\partial t} \\ &= \frac{\partial H^*}{\partial t} \end{aligned}$$

where $H^* = H(t, x^*(t), u^*(t), \lambda^*(t))$. Furthermore, if the Hamiltonian is not explicitly a function of time t , then it is invariant and constant across the optimal trajectory,

$$H(t, x^*(t), u^*(t), \lambda^*(t)) = \text{constant};$$

and its value can be determined by (5.39). The problem considered here assumes that the initial time and states are given. However, extending the derivation to account for free t_0 and $x(t_0)$ is straightforward. This can be done by considering the variations that are caused by the variations of t_0 and $x(t_0)$ in the Lagrange and terminal costs. Furthermore, the boundary conditions constraint can be dealt with by augmenting the cost function and taking the variation of the augmented cost in a similar manner to the above treatment.

Definition 5.6. Pontryagin's Minimum Principle: Optimal control problems in practice typically have bounds on the controls. The total torque provided to the wheels of a vehicle is limited by the engine power, a space rocket can only have a limited thrust and actuators of an industrial robot are subject to saturation. When the control is

subject to a bound, the control variation δu might enter an inadmissible region and therefore equation (5.38) obtained by requiring $\delta J(u, \delta u)$ to vanish, can no longer provide the necessary condition for optimality of the control. Pontryagin and his students in 1956 formulated the necessary conditions for the optimal control in the presence of control bounds. If the set of all admissible controls is denoted by U , this principle states that, the optimal control u^* that results in the minimum value of cost functional J must satisfy

$$H(t, x^*(t), u^*(t), \lambda^*(t)) \leq H(t, x^*(t), u(t), \lambda^*(t)), \quad \forall u \in U \quad (5.41)$$

for all for $t_0 \leq t \leq t_f$. Therefore, the optimal control u^* is the one that causes the Hamiltonian to take its global minimum. The principle is also often presented as

$$u^*(t) = \arg \min H(t, x^*(t), u(t), \lambda^*(t)), \quad u \in U \quad (5.42)$$

In summary, if the u^* lies on the boundary of U , then (5.41) or (5.42), which are known as the weak form of the Minimum Principle, can be readily employed to find the optimal control. In contrast, if u lies inside of U , then the strong form of Minimum Principle $\nabla_u H = 0$, also given in (5.38) can be applied. Equations (5.36), (5.37) (5.39), (5.40) and (5.41) offer the necessary but not the sufficient conditions for optimality. In much of classical literature and Pontryagin's original work, the Hamiltonian is defined with an opposite sign and thus this result is also referred to as the Maximum Principle. A rigorous proof of the Minimum Principle can be found in (Liberzon, 2011). The first order necessary condition for the following optimal control problem.

Minimise the cost functional:

$$J(u) = \Phi(x(t_f), t_f, t_0, x(t_0)) + \int_{t_0}^{t_f} L(x(t), u(t), t) dt,$$

subject to the state dynamics

$$\dot{x}(t) = f(x(t), u(t), t)$$

the boundary conditions

$$\Phi(x(t_f), t_f, t_0, x(t_0)) = 0$$

are given by:

$$\dot{x}^* = \nabla_{\lambda}(x^*, u^*, \lambda^*)$$

$$\dot{\lambda}^* = -\nabla_x(x^*, u^*, \lambda^*)$$

$$u^* = \arg \min H(x^*, u, \lambda^*), \quad u \in U$$

$$\Phi(x(t_f), t_f, t_0, x(t_0)) = 0.$$

Definition 5.7. Optimality Conditions for ODE Systems: An ODE system is one whose dynamics are generally given in the form

$$\dot{x} = f(x, u, t) \tag{5.43}$$

where $x(t) \in X$ is the state variable defined in the space $X = \mathbb{R}^{N_x}$, $u(t) \in U$ is the control variable defined in the space $U = \mathbb{R}^{N_u}$, and the time variable t is defined in the interval $[t_0, t_f]$. The function describing the dynamics of the state x is assumed to be continuously differentiable with respect to the variables x and u . The initial condition for the state variable is given by the vector x_0 , such that $x(t_0) = x_0$.

An optimal control problem for an ODE system in the form (5.44) with a bounded control can be put in the form

$$(P): \quad \min \Phi(x(t_f)) + \int_{t_0}^{t_f} L(x, u, t) dt \tag{5.44}$$

$$s. t. \quad \dot{x} = f(x, u, t) \quad \forall t \in [t_0, t_f]$$

$$x(0) = x_0$$

$$u(t) \in U_B$$

The objective functional is defined by the dynamic cost function L and the final cost function Φ , which are both assumed to be continuously differentiable with respect to x and u . The set U_B is set defined by

$$U_B = \{u \in U | u_L \leq u \leq u_U\}$$

where u_L is the lower bound and u_U is the upper bound for the control actions. Without loss of generality, the variables t_0 and t_f are assumed to be fixed values, such that $t_0 < t_f$.

The necessary optimality conditions for the problem (p) are usually expressed with respect to the Hamiltonian function

$$H(x, \lambda, u, t) = L(x, u, t) + \lambda^T f(x, u, t)$$

where $\lambda(t) \in \mathbb{R}^{N_x}$ is the adjoint variable, also known as costate. From Sira-Ramirez and Agrawal(2004) the necessary conditions for (x^*, λ^*, u^*) to be optimal to p_{ODE} are

$$\dot{x}^* = \frac{\partial H}{\partial \lambda} = f(x^*, u^*, t) \quad (5.45a)$$

$$-\dot{\lambda}^* = \frac{\partial H}{\partial x} = \frac{\partial L}{\partial x} + \lambda^{*T} \frac{\partial f}{\partial x} \quad (5.45b)$$

$$u^* = \arg \min H(x^*(t), \lambda^*(t), u, t) \quad u \in U_B \quad (5.45c)$$

$$\lambda^* = \frac{\partial \Phi}{\partial x}(x^*(t_f), t_f), \quad x^*(t_0) = x_0 \quad (5.45d)$$

which forms a system of $2N_x + N_u$ variables that are defined by $2N_x$ differential equations and N_u algebraic equations. For the differential equations, final conditions are imposed on the costates and initial conditions on the system states. Since a mix of initial and final conditions is given, the problem is characterized as a Boundary Value Problem

Notice that the conditions (5.45) are only necessary for optimality of OCP (5.44). If the optimum control u^* lies in the interior of U_B for all t , then the minimization of u^* in (5.45c) satisfies the condition $\frac{\partial H}{\partial u} = 0$. However, there will be times when the optimal control will be at the boundary of U_B , in which case the condition $\frac{H_{ODE}}{\partial u} = 0$ may not be verified. Let us assume that the Hamiltonian H is convex with respect to u . If the solution of $\frac{\partial H}{\partial u} = 0$ induces a solution $\hat{u}(t)$ that lies in the interior of U_B , then

$u^*(t) = \hat{u}(t)$ at time t . Otherwise, if $\hat{u}(t)$ does not lie in the interior of U_B , then, in such cases, the optimum is obtained by applying the value of the violated bounds (Sira-Ramirez and Agrawal, 2004). For instance, if $\hat{u}(t) \in \mathbb{R}$ is above u_U , at some time $t = t_1$, then $u^*(t_1) = u_U$. Therefore, given that $\hat{u}(t)$ is the solution to $\frac{H_{ODE}}{\partial u} = 0$, (5.45c) can be written for a convex Hamiltonian as:

$$\hat{u}(t) = \begin{cases} u_U, & \text{if } u_U \leq \hat{u}, \\ \hat{u}, & \text{if } u_L < \hat{u} < u_U, \\ u_L, & \text{if } \hat{u} \leq u_L \end{cases}$$

Theorem 5.2. (First Order Necessary Conditions). Consider the problem to minimize the functional

$$J(u) = \int_{t_0}^{t_f} l(t, x(t), u(t)) dt,$$

$$\text{Subject to } \dot{x}(t) = f(t, x(t), u(t)), \quad x(t_0) = x_0, \quad (5.46)$$

for $u \in C[t_0, t_f]^{n_u}$ with fixed endpoints $t_0 < t_f$, where l and f are continuous in (t, x, u) and have continuous first partial derivatives with respect to x and u for all $(t, x, u) \in [t_0, t_f] \times R^{n_x} \times R^{n_u}$. Suppose that $u^* \in C[t_0, t_f]^{n_u}$ is a (local) minimizer for the problem, and let $x^* \in C[t_0, t_f]^{n_x}$ denote the corresponding response. Then, there is a vector function $\lambda^* \in C[t_0, t_f]^{n_x}$ such that the triple $[u^*, x^*, \lambda^*]$ satisfies the system

$$\dot{x}(t) = f(t, x(t), u(t)); \quad x(t_0) = x_0, \quad (5.47)$$

$$\dot{\lambda}(t) = -l_x(t, x(t), u(t)) - f_x(t, x(t), u(t))^T \lambda(t); \quad \lambda(t_f) = 0 \quad (5.48)$$

$$0 = l_u(t, x(t), u(t)) + f_u(t, x(t), u(t))^T \lambda(t). \quad (5.49)$$

for $t_0 \leq t \leq t_f$. These equations are known collectively as the Euler-Lagrange equations, and (5.48) is often referred to as the adjoint equation (or the costate equation).

Example 5.3. Consider the problem:

$$\min J(u) = \int_0^{10} (x^2(t) + u^2(t)) dt \quad (5.50)$$

$$\text{s.t. } \dot{x}(t) = -x^3(t) + u(t), \quad (5.51)$$

$$x(0) = 1, \quad x(10) = \frac{3}{2} \quad (5.52)$$

here, we use the ideas developed in the previous section on the extermination of functions conditions. Consider the optimization of the functional (5.50) with the boundary conditions (5.52) under the condition describing the ODE (5.51). First we rewrite the condition (5.51) as

$$f(t, x(t), u(t)) = \dot{x}(t) + x^3(t) - u(t) = 0 \quad (5.53)$$

Now, we form an augmented functional as:

$$\begin{aligned} J &= \int_0^1 [x^2(t) + u^2(t) + \lambda(t)\{\dot{x}(t) + x^3(t) - u(t)\}] dt \\ &= \int_0^1 l(t, x(t), u(t), \lambda(t)) dt \end{aligned} \quad (5.54)$$

where, $\lambda(t)$ is the Lagrange multiplier, and

$$l(x(t), u(t), \lambda(t)) = x^2(t) + u^2(t) + \lambda(t)\{\dot{x}(t) + x^3(t) - u(t)\} \quad (5.55)$$

is the Lagrangian.

Now, we apply the Euler-Lagrange equation to the previous Lagrangian to get

$$\left(\frac{\partial l}{\partial x}\right)_* - \frac{d}{dt} \left(\frac{\partial l}{\partial \dot{x}}\right)_* = 0 \rightarrow 2x^*(t) + 3\lambda^*(t)x^{*2}(t) - \dot{\lambda}^*(t) = 0 \quad (5.56)$$

$$\left(\frac{\partial l}{\partial u}\right)_* - \frac{d}{dt} \left(\frac{\partial l}{\partial \dot{u}}\right)_* = 0 \rightarrow 2u^*(t) - \lambda^*(t) = 0 \quad (5.57)$$

$$\left(\frac{\partial l}{\partial \lambda}\right)_* - \frac{d}{dt} \left(\frac{\partial l}{\partial \dot{\lambda}}\right)_* = 0 \rightarrow \dot{x}^*(t) + x^{*3}(t) - u^*(t) = 0 \quad (5.58)$$

and solve for optimal $x^*(t)$, $u^*(t)$ and $\lambda^*(t)$. We get first from (5.57) and (5.58)

$$\lambda^*(t) = 2u^*(t) = 2(\dot{x}^*(t) + x^{*3}(t)) = 0 \quad (5.59)$$

Using the equation (5.59) in (5.56)

$$2x^*(t) + (3x^{*2}(t))2(\dot{x}^*(t) + x^{*3}(t)) - 2(\ddot{x}^*(t) + \dot{x}^{*3}(t)) = 0 \quad (5.60)$$

Solving the previous equation, we get BVP.

$$\ddot{x}(t) + \dot{x}^3(t) - 3x^2(t)\dot{x}(t) - 3x^5(t) - x(t) = 0$$

$$x(0) = 1$$

$$x(10) = 3/2,$$

Where our conditions $x(0) = 1$ and $x(10) = 3/2$, are specified on the boundary of the interval of interest $t \in [0,10]$. (Though our solution will typically extend beyond this interval, the most common scenario in boundary value problems is the case in which we are only interested in values of the independent variable between the specified endpoints.) The first step in solving this type of equation is to write it as a first order system with $x_1(t) = x(t)$ and $x_2(t) = \dot{x}(t)$, for which we have

$$\dot{x}_1(t) = x_2(t)$$

$$\dot{x}_2(t) = 3x_1^2(t)x_2(t) + 3x_1^5(t) - x_2^3(t) + x_1(t).$$

We record this system in MATLAB code appendix C.

Table 5.3. Numerical results in IMS for example 5.3.

t	$x(t)$	$u(t)$	$\lambda(t)$	$dx(t)/dt$
0.0000	1.0000	0.3056	0.6112	-0.6944
0.7905	0.5908	-0.2224	-0.4448	-0.4286
3.1121	0.0704	-0.0679	-0.1358	-0.0683
5.3845	0.0160	0.0016	0.0032	0.0016
7.6625	0.0870	0.0865	0.173	0.0858
9.53990	0.6261	1.0487	2.0974	0.8033
10.0000	1.5000	8.9882	17.9764	5.6132

From this table (5.3) exact boundary condition 1.5000 and estimated boundary condition 1.5000, this show IMS exactly get the point at the given time interval $[0,10]$, also we can see in below table (5.3).

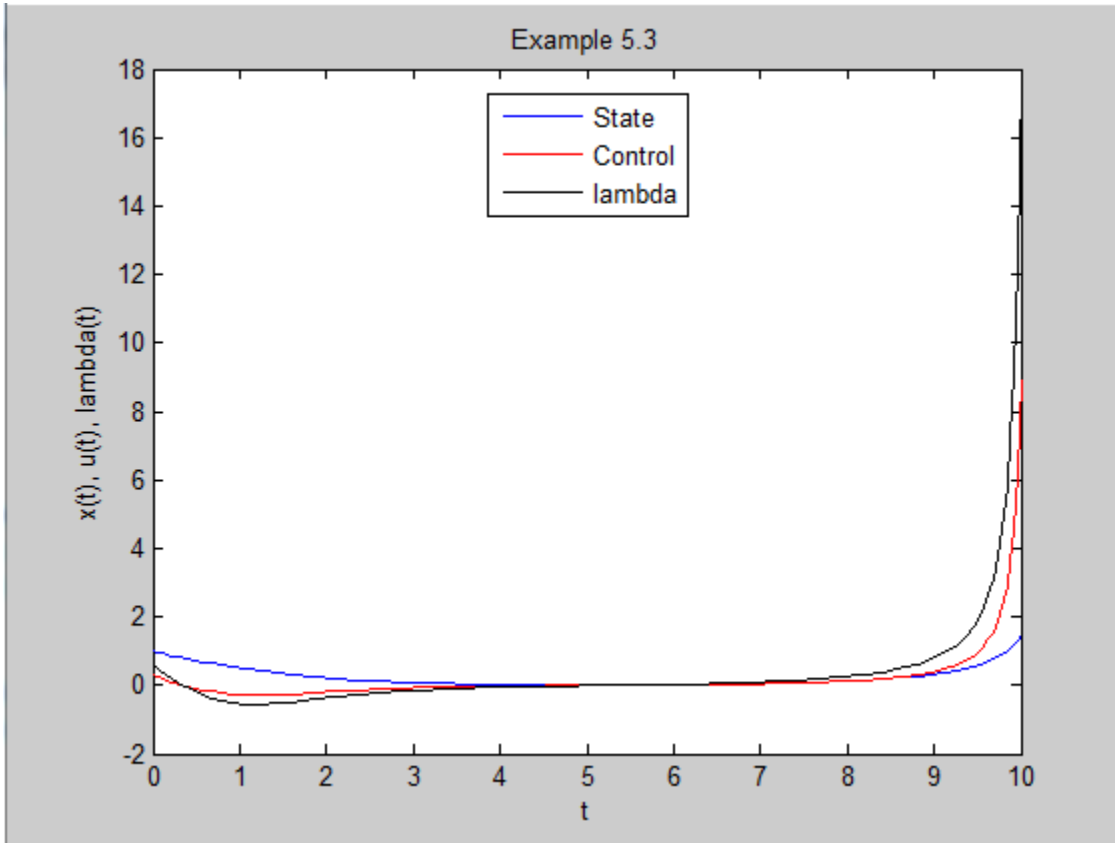


Figure 5.3. Graphs of numerical solutions for IMS in example 5.3.

Example 5.4. [Semi-batch Isothermal Reactor]. Consider the optimal problem

$$\min J(u) = \int_0^3 x(t)^2 + u(t)^2 dt \quad (5.61)$$

$$\text{s.t. } \frac{dx}{dt} = (1+x)x + u, t \in [0,3], x(0) = x_0 = 0.05 \quad (5.62)$$

$$x(t) \geq -1, x(t) \leq 1$$

$$x(3) = 0, -1 \leq u(t) \leq 1 \quad (5.63)$$

here, we use the ideas developed in the previous section on the extermination of functions conditions. Consider the optimization of the functional (5.61) with the boundary conditions (5.63) under the condition describing the ODE (5.62). First we rewrite the condition (5.62) as:

$$f(t, x(t), u(t)) = \dot{x}(t) - x(t)^2 - x(t) - u(t) = 0 \quad (5.64)$$

Now, we form an augmented functional as:

$$\begin{aligned} J &= \int_0^3 [x^2(t) + u^2(t) + \lambda(t)\{\dot{x}(t) - x^2(t) - x(t) - u(t)\}] dt \\ &= \int_0^3 [L(x(t), \dot{x}(t), u(t), \lambda(t))] dt \end{aligned} \quad (5.65)$$

where, $\lambda(t)$ is the Lagrange multiplier, and

$$L(t, x(t), u(t), \lambda(t)) = x^2(t) + u^2(t) + \lambda(t)\{\dot{x}(t) - x^2(t) - x(t) - u(t)\} \quad (5.66)$$

is the Lagrangian.

Now, we apply the Euler-Lagrange equation to the previous Lagrangian to get

$$\left(\frac{\partial L}{\partial x}\right) - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) = 0 \rightarrow 2x(t) - 2\lambda(t)x(t) - \lambda(t) - \dot{\lambda}(t) = 0 \quad (5.67)$$

$$\left(\frac{\partial L}{\partial u}\right) - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{u}}\right) = 0 \rightarrow 2u(t) - \lambda(t) = 0 \quad (5.68)$$

$$\left(\frac{\partial L}{\partial \lambda}\right) - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\lambda}}\right) = 0 \rightarrow \dot{x}(t) - x^2(t) - x(t) - u(t) = 0 \quad (5.69)$$

and solve for optimal $x^*(t)$, $u^*(t)$ and $\lambda^*(t)$. We get first from (5.68) and (5.69),

$$\lambda(t) = 2u(t) = 2(\dot{x}(t) - x^2(t) - x(t)) = 0 \quad (5.70)$$

using the equation (5.70) in (5.67), we get

$$2x(t) - 2(2\dot{x} - \dot{x}^2 - 2x) - (2\dot{x} - 2x^2 - 2x) - (2\ddot{x} - 2\dot{x}^2 - 2\dot{x}) = 0 \quad (5.71)$$

simplifying (5.71), we have

$$\ddot{x}(t) + 2\dot{x}(t) - \dot{x}^2(t) - 3x^3(t) + x(t) = 0$$

$$x(0) = 0.05$$

$$x(3) = 0,$$

where our conditions $x(0) = 0.05$ and $x(3) = 0$, are specified on the boundary of the interval of interest $t \in [0,3]$. (Though our solution will typically extend beyond

this interval, the most common scenario in boundary value problems is the case in which we are only interested in values of the independent variable between the specified endpoints.) The first step in solving this type of equation is to write it as a first order system with $x_1(t) = x(t)$ and $x_2(t) = \dot{x}(t)$, for which we have

$$\dot{x}_1(t) = x_2(t)$$

$$\dot{x}_2(t) = -2x_1(t)x_2(t) + x_2^2(t) + 2x_1^3(t) + 3x_1^2(t) + 2x_1(t).$$

we record this system in MATLAB code appendix D.

Table 5.4. Numerical results for IMS in example 5.4.

t	$x(t)$	$u(t)$	$\lambda(t)$	$dx(t)/dt$
0.000	0.0500	-0.1279	-0.2558	-0.0754
0.3503	0.0297	-0.0796	-0.1592	-0.0490
0.8753	0.0138	-0.0340	-0.0680	-0.0200
1.4003	0.0065	-0.0159	-0.0318	0.0094
1.9253	0.0029	-0.0075	-0.0150	-0.0046
2.4503	0.0012	-0.0037	-0.0074	-0.0025
3.0000	0.0000	-0.0019	-0.0038	-0.0019

In this table (5.4) exact boundary condition 0.0000 and estimated boundary condition 0.0000, that make IMS it was get directly exact boundary condition in the given time interval [0,3] we can see in table (5.4) and in addition to that time interval in [0,3] increase control variable it was increase, but state variable have been decreased.

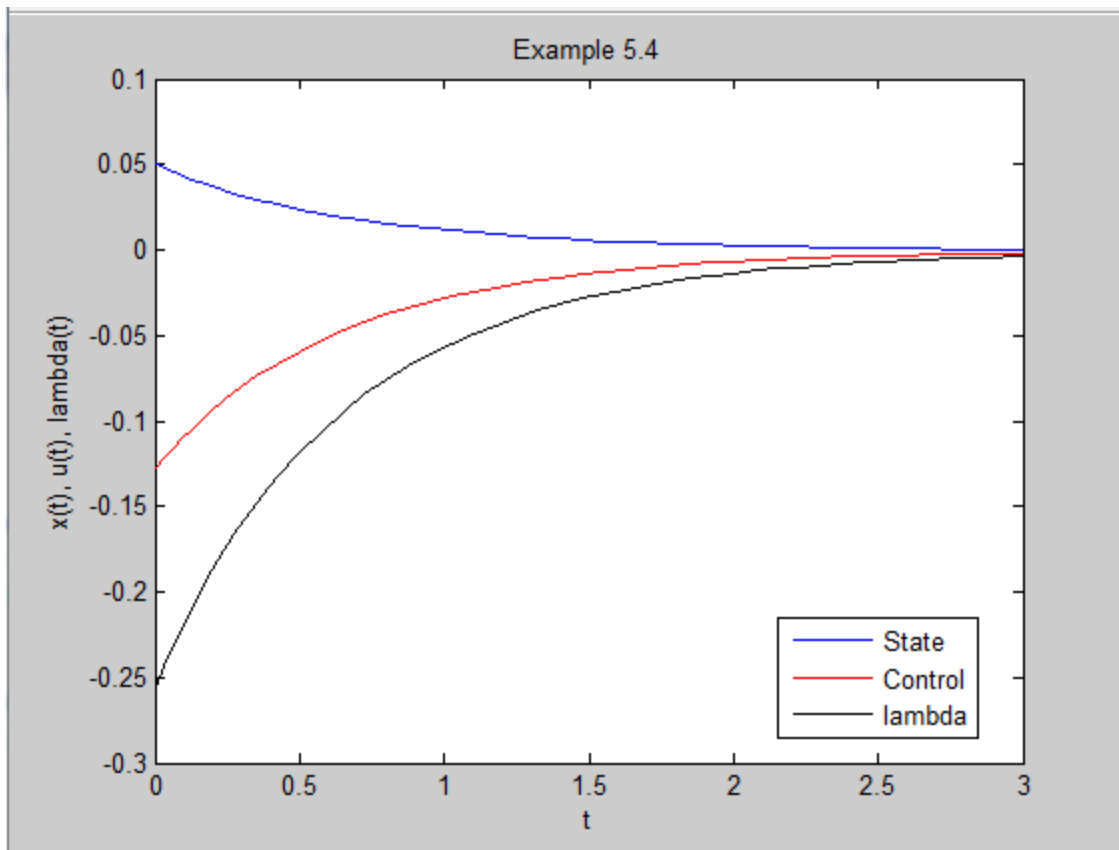


Figure 5.4. Graphs of numerical solutions for IMS in example 5.4.

5.3. Comparisons of DMS and IMS for Optimal Control (OCP)

Table 5.5. Comparisons of numerical results for DMS and IMS in example 5.1 and 5.3 respectively.

Methods	Iterations	Tims(s)	Exact value	Approximate value	Error
DMS	12	46	6.723925	16.5811	9.857175
IMS	13	24	6.723925	7.3479	0.623975

Discussion and Result

Table (5.5) was solved by direct and indirect multiple shooting methods, the approximate value is found to be 7.3479, which is very close to the optimal value

6.723925. From this table we conclude that indirect multiple shooting method it was better than direct multiple shooting method.

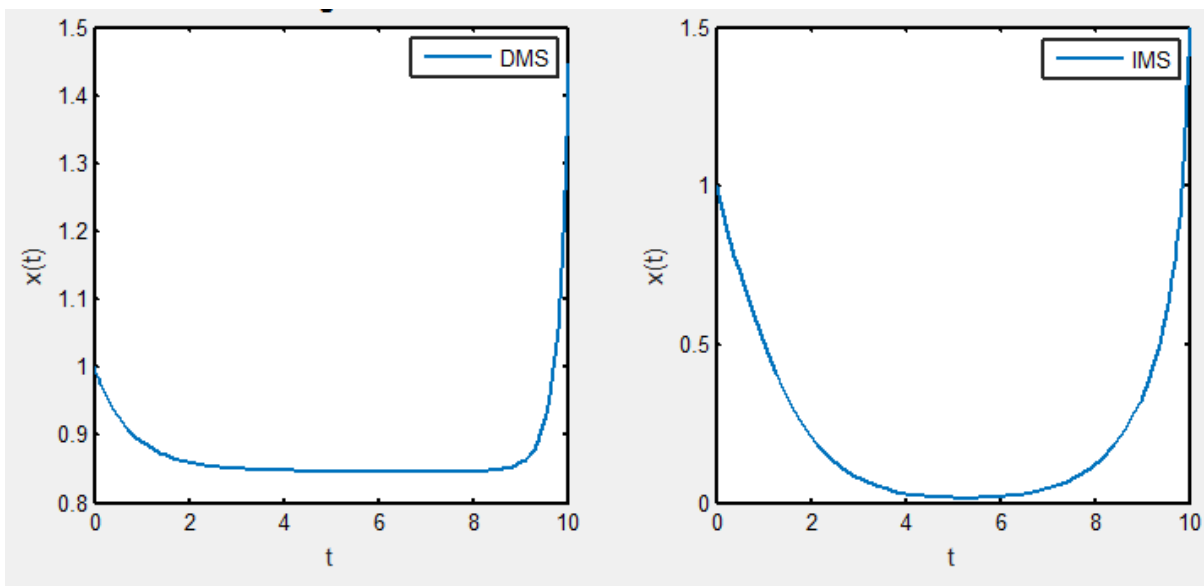


Figure 5.5. DMS and IMS in example 5.1 and 5.3 respectively.

From figure (5.5) it was 1.5000 exact boundary condition and 1.4459 estimated boundary condition for DMS and also, 1.5000 estimated boundary condition for IMS, that it was show IMS is more accurate than DMS by the difference of 0.0541 in table 5.1 and 5.3 we have seen in the figure 5.5

Table 5.6. Comparisons of numerical results for DMS and IMS in example 5.2 and 5.4.

Methods	Time(s)	Exact value	Approximate value	Error
DMS	14	0.006197	0.0071	0.000903
IMS	7	0.006197	0.0062	0.000003

Table (5.6) was solved by direct and indirect multiple shooting methods, the approximate value is found to be 0.0062, which is very close to the optimal value

0.006197 with 0.000003 error only. From this table we conclude that indirect multiple shooting method it was more accurate than direct multiple shooting method. In Subsections 5.1 and 5.2 the IM and DM methods were presented. Several aspects make them appear unrelated, the main one being the absence of any adjoint equation in DM. DM will be rewritten in a manner that reveals its close relation to IM. The current subsection, however, is dedicated to numerical example illustrating the discussed results on IMS and DMS.

In addition the last decade the applications of the optimal control problems (sometimes called also dynamic optimization) have spread over many research and industrial fields as proved (Sargent,2000). However, even if the theoretical framework (namely calculus of variation, NLP) has been quite clearly defined, its application to industrial or real problems is still difficult for the complexity and dimensionality of the system of equations to deal with. Two groups of techniques are commonly employed to solve this kind of problem. The first group involves the so called direct methods, which instead of solving the necessary condition discretize the problem in order to obtain a NLP, which may be solved by means of consolidated numerical schemes, such as Initial Value Solver (IVS) and Sequential Quadratic Programming (SQP). In this sense they are easier to implement, and probably this is the main reason why they are widely employed.

The second group consists of indirect methods, which are based on the solution of the necessary condition of optimality, as derived by the Calculus of Variation. Indirect methods usually produce very accurate solution and are quite sensitive to design parameters variations, since they solve directly the equations of first necessary condition. However, to fully exploit the indirect method one has to produce the symbolical expressions of the necessary conditions of optimality. In other words, according to the calculus of variations, the adjoint equations have to be calculated hopefully along with their jacobians for the numerical solution of the arising (TPBVP). In both cases (direct or indirect method) it is essential to find a solution of the TPBVP in reasonably fast time. Thus, in dealing with such large problems, the numerical algorithm plays a major.

6. SUMMARY AND CONCLUSIONS

6.1. Summary

The main objective this project was solve nonlinear optimal control problems, were observed in many engineering applications, they are still difficult to solve globally with high confidence. The problems arising in this project work were classified based on the structure of performance index J . If the performance index contains the terminal cost function only, it was called the Mayer problem, if the performance index had only the integral cost term, it was called the Lagrange problem, and the problem was of the Bolza type if the performance index contains both the terminal cost term and the integral cost term. But Mayer and Bolza problems are not used in this project and thus only described briefly.

In this project, we have seen how non-linear optimal control problem, which not solved analytically, was approximated by using family of two numerical methods (indirect and direct multiple shooting methods). Indirect methods are based on Pontryagin's maximum principle and optimization in an infinite-dimensional function space. Necessary conditions of optimality are used to transform the optimal control problem into a two-point boundary value problem, which can be solved by indirect multiple shooting method. For very special small cases, an analytical solution is also possible. Indirect methods were suitable for a theoretical analysis of a problem's solution structure. They yield highly accurate solutions for the optimal control profile, as the infinite-dimensional problem was solve and no approximation of the control functions took place. This was in sharp contrast to direct multiple shooting methods.

Direct method was solve optimal control problems for non-linear systems based on discretization with respect to time. That means the infinite-dimensional problem of finding optimal state and control trajectories is transformed into a finite dimensional optimization by evaluating state and control values at those samples, the cost criterion was directly optimized. The optimal control problem is usually formulated as a direct numerical method. The computational costs for those methods were analyzed for

different examples. Their efficiency was often deepened up on optimal control problem. The compressions of all family of those methods were drawn and obtain that indirect method is accurate some of their applications were presented to make the concept clear. Finally, some examples given to show the accuracy of indirect multiple shooting method. For those methods, the MATLAB codes were used to obtain the approximate value of given non-linear optimal control problem and for this project work present it in appendix.

6.2. Conclusion

This project work provided a thorough examination of existing indirect and direct multiple shooting methods approaches for nonlinear optimal control problem as well as extensions of the method in different directions. Both indirect and direct methods were formulated with in a common abstract framework, enabling their efficient application to complex problems. Indirect and direct methods were applicable in science and engineering, since they are readily evaluates different non-linear optimal control problem. Indirect methods derive the necessary conditions for optimality using co-state variables; convert the optimal control problem into a two point boundary value problem.

The convergence domain was so narrow that a large number of initial guesses have to be tried to obtain convergence. In order to eliminate the disadvantages of indirect methods and to obtain better convergence properties, direct methods were examined. In a direct method, the state and/or control of the original OCP are approximated/discretized/parameterized in some appropriate manner. In the indirect and direct methods different aspects made them appear unrelated, the main one being the absence of any adjoint equation in direct method. Furthermore, the proposed methods could be implemented in MATLAB code. Lastly, from the result of their comparison, we obtain indirect multiple shooting method is more accurate than direct multiple shooting method.

7. REFERENCES

- Arcak, M. and Teel, A. 2002. Input-to-State Stability for a Class of Lurie Systems, 38(11):1945-1949.
- Ascher, U., Christiansen, J. and Russell, R.D. 1979. A Collocation Solver for Mixed Order Systems of Boundary Value Problems. *Mathematics of Computation*, 33(146):659-679.
- Bertrand, R. and Epenoy, R. 2002. New smoothing Techniques for Solving Bang-Bang Optimal Control Problems Numerical Results and Statistical Interpretation. *Optimal Control Applications and Methods*, 23(4):171-197.
- Betts, J.T. 2010. *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, 19. Siam.
- Biegler, L.T. 2007. An Overview of Simultaneous Strategies for Dynamic Optimization. *Chemical Engineering and Processing: Process Intensification*, 46(11), 1043-1053.
- Bliss, G.A. 1946. Lectures on the Calculus of Variations, Phoenix Science Series. *The University of Chicago Press, Chicago*, 37:1-6.
- Bock, H.G. and Plitt, K.J. 1984. A multiple Shooting Algorithm for Direct Solution of Optimal Control Problems. *IFAC Proceedings Volumes*, 17(2):1603-1608.
- Bryson, A. E. and Ho, Y. 1975. *Applied Optimal Control*, Wiley.
- Bulirsch, R., Montrone, F. and Pesch, H.J. 1991. Abort Landing in the Presence of Windshear as a Minimax Optimal Control Problem, Part 2: Multiple Shooting and Homotopy. *J. of Opt. Theory and Applications*, 70(2):223-254
- Büskens, C. and Maurer, H. 2000. SQP-Methods for Solving Optimal Control Problems with Control and State Constraints: Adjoint Variables, Sensitivity Analysis and Real-Time Control. *Journal of Computational and Applied Mathematics*, 120(1-2):85-108.
- Capuzzo-Dolcetta, I. and Lions, P.L. 1990. Hamilton-Jacobi Equations with State Constraints. *Transactions of the American Mathematical Society*, 318(2), 643-683.
- Conway, B.A. ed. 2010. *Spacecraft Trajectory optimization*. Cambridge University Press.
- Dickmanns, E. D. and Will, K. H. 1974. "Approximate Solution of Optimal Control Problems Using Third Order Hermite Polynomial Functions," in Proceedings of The IFIP. Technical Conference, Springer-Verlag, 158–166.

- Diehl, M., Bock, H.G., Diedam, H. and Wieber, P.B. 2006. Fast Direct Multiple Shooting Algorithms for Optimal Robot Control. In *Fast Motions in Biomechanics and Robotics*,(65-93). Springer, Berlin, Heidelberg.
- Elsgolts,L.E. 1962. Calculus of Variations. *Addison-Wesley*.
- Freeman, R.A. and Primbs, J.A. 1996, December. Control Lyapunov Functions: New Ideas from an Old Source. In *Decision and Control, 1996., Proceedings of The 35th IEEE Conference on* , 4: 3926-3931.
- Gelfand, I.M. and Silverman, R.A. 2000. *Calculus of Variations*. Courier Corporation.
- Goh, C.J. and Teo, K.L.1988. Control parametrization: a Unified Approach to Optimal Control Problems with General Constraints. *Automatica*, 24(1), 3-18.
- Hargraves, C.R. and Paris, S.W. 1987. Direct Trajectory Optimization Using Nonlinear Programming and Collocation. *Journal of Guidance, Control, and Dynamics*,10(4):338-342.
- Hartjes, S., Dons, J. and Visser, H.G. 2014. Optimization of Area Navigation Arrival Routes for Cumulative Noise Exposure. *Journal of Aircraft*, 51(5):1432-1438.
- Hauser, K. 2017. Learning the Problem-Optimum Map: Analysis and Application to Global Optimization in Robotics. *IEEE Transactions on Robotics*, 33(1):141-152.
- Jetchev, N. and Toussaint, M. 2013. Fast Motion Planning from Experience: Trajectory Prediction for Speeding up Movement Generation. *Autonomous Robots*, 34(1-2):111-127.
- Jiang, F., Baoyin, H. and Li, J. 2012. Practical Techniques for Low Thrust Trajectory Optimization with Homotopic Approach. *Journal of Guidance, Control, and Dynamics*, 35(1): 245-258.
- Kelley, H.J. 1960. Gradient Theory of Optimal Flight Paths. *Ars Journal*, 30(10): 947-954.
- Kemble, S. 2006. *Interplanetary Mission Analysis and Design*. Springer Science & Business Media.
- Krik,D.E. 1970. Optimal Control Theory. *Prentice Hal, Englewood Cliffs, NJ*.
- Krstic, M., Kanellakopoulos, I. and Kokotovic,P.1995. Nonlinear and Adaptive Control Design. *John Wiley&Sons,NewYork*.
- Kumar, R.R. and Seywald, H. 1995. Fuel Optimal Stationkeeping via Differential Inclusions. *Journal of Guidance, Control, and Dynamics*, 18(5): 1156-1162.

- Lampariello, R., Nguyen-Tuong, D., Castellini, C., Hirzinger, G. and Peters, J. 2011, May. Trajectory Planning for Optimal Robot Catching in Real time. In *Robotics and Automation (ICRA)*, 2011 *IEEE International Conference on*, 3719-3726.
- Liberzon, D. 2011. *Calculus of Variations and Optimal Control Theory: a Concise Introduction*. Princeton University Press.
- Marco, F. 2014. Numerical Methods for Optimal Control Problems with Application to Autonomous Vehicles. University of Trento, 197:2.
- Marsden, J.E., Sirovich, L., Antman, S.S., Iooss, G., Holmes, P., Barkley, D., Dellnitz, M. and Newton, P. 2007. Texts in Applied Mathematics 52.
- Naidu, D.S. 2002. *Optimal Control Systems*. CRC press.
- Pinch, E.R.1993. Optimal Control and the Calculus of Variations. Oxford University Press: London.
- Pontryagin, L.S.1959.Optimal Control Processes.Uspekhi Matematiches Kikh Nauk,14:3–20.
- Pytlak, R.1999. Numerical Methods for Optimal Control Problems with State Constraints, 1707 of Lecture Notes in Mathematics. Springer-Verlag, Berlin, Berlin,
- Russell, R.P. 2007. Primer Vector Theory Applied to Global Low-Thrust Trade Studies. *Journal of Guidance, Control, and Dynamics*, 30(2): 460-472.
- Sage, A.P. and White, C.C. 1977. *Optimum Systems Control*. Prentice Hall.
- Sager,S. 2006. Numerical Methods for Mixed–Integer Optimal Control Problems. PhD these is, Universitat Heidelberg.
- Sargent, R.W.H. 2000. Optimal Control. *Journal of Computational and Applied Mathematics*, 124(1-2):361-371.
- Seywald, H. 1994. Trajectory Optimization Based on Differential Inclusion (Revised). *Journal of Guidance, Control, and Dynamics*, 17(3): 480-487.
- Shirazian, M. and Effati, S. 2012. Solving a Class of Nonlinear Optimal Control Problems via He’s Variational Iteration Method. *International Journal of Control, Automation and Systems*, 10(2): 249-256.
- Sira-Ramirez, H. and Agrawal, S.K. 2004. *Differentially flat systems*. Crc Press.
- Stoer, J.and Bulirsch,R. 2002.Introduction to Numerical Analysis. Springer,New York,3rd ed.
- Tang, G.Y. 2005. Suboptimal Control for Nonlinear Systems: a Successive Approximation Approach. *Systems & Control Letters*, 54(5): 429-434.

- Vinter, R. 1993. Convex duality and Nonlinear Optimal control. *SIAM Journal on Control and Optimization*, 31(2),518-538.
- Zuiani, F., Palmas, A., Avanzini, G. and Vasile, M. 2011. Direct Transcription of Low-Thrust Trajectories With Finite Trajectory Elements. *Agenda*, 8.

8. APPENDICES

8.1. Matlab Code A

```

clear all;
close all;
clc;
format compact;
IG = 1; % initial guess for dydx
Ti = 0;
Tf = 10;
Yi = 1; % first BC
Yf = 1.5; % second BC
[IGf, fval, exitf, output] = fzero(@(z) resFun(z), IG);
[x, y] = ode45(@odeDMS, [0 10], [Yi IGf] );
XX=y(:,1); DX=y(:,2);
FDM = XX.^6 + XX.^2 + DX.^2 + 2*(XX.^3).*DX;
JJ = quad(@(t) interpDMS(t, x, FDM), 0, 10)
z1 = x';
z2 = y(:,1)';
z3 = y(:,2)';
z = [z1; z2; z3];
yy = y(:,1)';
u = y(:,2)'+yy.^3;
data = [x', y(:,1)', y(:,2)'];
fileID =fopen('odeDMS_DATA.txt','w');
fprintf(fileID,'%s %5.4f %s %5.4f \r\n', '% Exact BC = ', Yf, ' Estimated BC',
y(end, 1));
fprintf(fileID,'%s %4.4f %s %5.4f \r\n', '% initial guess = ', IG, 'improved guess',
IGf);
fprintf(fileID,'%s %4i \r\n', '% Interval iterations = ', output.intervaliterations);
fprintf(fileID,'%s %4i \r\n', '% Iteration number = ', output.iterations);
fprintf(fileID,'%s \r\n', '% ');

```

```

fprintf(fileID,'%s %5.4f \r\n', '% J_min = ', JJ);
fprintf(fileID,'%s          \r\n', '% ');
fprintf(fileID,'%% t          x(t)          dx(t)/dt \r\n');
fprintf(fileID,'%5.4f      %10.4f      %10.4f \r\n', z);
fclose(fileID);

output
fval
exitf

% y(end,1)

% output.intervaliterations

% output.iterations

plot(x, y(:,1), 'b-', x, u, 'r--');
xlabel('t');
ylabel('x(t), u(t)');
title('Example 5.1');
legend('DMS state', 'DMS control', 'location', 'best')
% subplot(1,2,1)

% plot(x, y(:,1))

% subplot(1,2,2)

% plot(x, y(:,2))

function dydx = odeDMS(t, y)
dydx = [y(2); -3*y(1)^2 + 3*(y(1)^2)*y(2) + 3*y(1)^5 + y(1)];
end

function res = resFun(dyguess)
Yi = 1;
Yf = 1.5;
Ti = 0;
Tf = 10;
[t,y] = ode45(@odeDMS, [Ti Tf], [Yi dyguess]);
res = y(end,1) - Yf;
end

```

8.2. Matlab Code B

```

clear all;
close all;
clc;
format compact;
IG = 1; % initial guess for dydx
Ti = 0;
Tf = 3;
Yi = 0.05; % first BC
Yf = 0; % second BC
[IGf, fval, exitf, output] = fzero(@(z) resFun(z), IG);
[x, y] = ode45(@odeDMS, [0 3], [Yi IGf] );
XX=y(:,1); DX=y(:,2);
FDM = XX.^2 + DX.^2- 2*DX.*XX - 2*DX.*XX.^2 + XX.^4 + 2*(XX.^3) +
2*XX.^2;
JJ = quad(@(t) interpDMS(t, x, FDM), 0, 3)
z1 = x';
z2 = y(:,1)';
z3 = y(:,2)';
z = [z1; z2; z3];
yy = y(:,1)';
u = y(:,2) '-yy.^2-yy;
data = [x', y(:,1)', y(:,2)'];
fileID =fopen('odeDMS_DATA.txt','w');
fprintf(fileID,'%s %5.4f %s %5.4f \r\n', '% Exact BC = ', Yf, '
Estimated BC', y(end, 1));
fprintf(fileID,'%s %4.4f %s %5.4f \r\n', '% initial guess = ',
IG, 'improved guess', IGf);
fprintf(fileID,'%s %4i \r\n', '% Interval iterations = ',
output.intervaliterations);
fprintf(fileID,'%s %4i \r\n', '% Iteration number = ',
output.iterations);
fprintf(fileID,'%s \r\n', '% ');
fprintf(fileID,'%s %5.4f \r\n', '% J_min = ', JJ);
fprintf(fileID,'%s \r\n', '% ');
fprintf(fileID,'%s t x(t) dx(t)/dt
\r\n');
fprintf(fileID,'%5.4f %10.4f %10.4f \r\n', z);

```

```

fclose(fileID);
output
fval
exitf
% y(end,1)
% output.intervaliterations
% output.iterations
plot(x, y(:,1), 'b-', x, u, 'r--');
xlabel('t');
ylabel('x(t), u(t)');
title('Example 5.2');
legend('DMS state', 'DMS control', 'location', 'best')

% subplot(1,2,1)
% plot(x, y(:,1))
% subplot(1,2,2)
% plot(x, y(:,2))
function dydx = odeDMS(t, y)
    dydx = [y(2); 3*y(1)^2 + 2*(y(1)^3) - 2*y(1)*y(2) + 4*y(1)]
end

function res = resFun(dyguess)
    Yi = 0.05;
    Yf = 0;
    Ti = 0;
    Tf = 3;
    [t,y] = ode45(@odeDMS, [Ti Tf], [Yi dyguess]);
    res = y(end,1) - Yf
end

```

8.3. Matlab Code C

```

clear all;
close all;
clc;
format compact;
IG = 1; % initial guess for dydx
Ti = 0;
Tf = 10;
Yi = 1; % first BC
Yf = 1.5; % second BC
[IGf, fval, exitf, output] = fzero(@(z) resFun(z), IG);
[x, y] = ode45(@odeIMS, [0 10], [Yi IGf]);
%JJ = quad(@(t) interpIMS(t, x, y(:,1)), 0, 10); % interpolation and quadratur are
performed on a single line.
XX=y(:,1); DX=y(:,2);
UU = DX + XX.^3; LL = 2*UU;
FIM = XX.^2 + UU.^2 + LL.*(DX + XX.^3 - UU);
JJ = quad(@(t) interpIMS(t, x, FIM), 0, 10)
z1 = x';
z2 = y(:,1)';
z3 = y(:,2)';
z = [z1; z2; z3];
data = [x', y(:,1)', y(:,2)'];
fileID =fopen('odeIMS_DATA.txt','w');
fprintf(fileID,'%s %5.4f %s %5.4f \r\n', '% Exact BC = ', Yf, ' Estimated BC',
y(end, 1));
fprintf(fileID,'%s %4.4f %s %5.4f \r\n', '% initial guess = ', IG, 'improved guess',
IGf);
fprintf(fileID,'%s %4i \r\n', '% Interval iterations = ', output.intervaliterations);
fprintf(fileID,'%s %4i \r\n', '% Iteration number = ', output.iterations);

```

```

fprintf(fileID,'%s          \r\n', '% ');
fprintf(fileID,'%s %5.4f\r\n', '% J_min = ', JJ);
fprintf(fileID,'%s          \r\n', '% ');
fprintf(fileID,'%% t          x(t)          dx(t)/dt \r\n');
fprintf(fileID,'%5.4f      %10.4f      %10.4f \r\n', z);
fclose(fileID);

%y(end,1)

%output.intervaliterations

%output.iterations

plot(x, y(:,1), x, UU, 'r-', x, LL, 'k-');
xlabel('t');
ylabel('x(t), u(t), lambda(t)');
title('Example 5.3');
legend('State', 'Control', 'lambda', 'location', 'best')
% subplot(1,2,1)

% plot(x, y(:,1))

% subplot(1,2,2)

% plot(x, y(:,2))

function dydx = odeIMS(t, y)
    dydx = [y(2); 3*(y(1)^2)*y(2) + 3*y(1)^5 - y(1)^3 + y(1)];
end

function res = resFun(dyguess)
    Yi =1;
    Yf = 1.5;
    Ti = 0;
    Tf = 10;
    [t,y] = ode45(@odeIMS, [Ti Tf], [Yi dyguess]);
    res = y(end,1) - Yf;
end

```

8.4. Matlab Code D

```

clear all;
close all;
clc;
format compact;
IG = 1; % initial guess for dydx
Ti = 0;
Tf = 3;
Yi = 0.05; % first BC
Yf = 0; % second BC
[IGf, fval, exitf, output] = fzero(@(z) resFun(z), IG);
[x, y] = ode45(@odeIMS, [0 3], [Yi IGf] );
%JJ = quad(@(t) interpIMS(t, x, y(:,1)), 0, 10); % interpolation and
quaderatur are performed on a single line.
XX=y(:,1); DX=y(:,2);
UU = DX - XX.^2 -XX; LL = 2*UU;
FIM = XX.^2 + UU.^2 + LL.*(DX - XX.^2 - XX - UU);
JJ = quad(@(t) interpIMS(t, x, FIM), 0, 3)
z1 = x';
z2 = y(:,1)';
z3 = y(:,2)';
z = [z1; z2; z3];
data = [x', y(:,1)', y(:,2)'];
fileID =fopen('odeIMS_DATA.txt','w');
fprintf(fileID,'%s %5.4f %s %5.4f \r\n', '% Exact BC = ', Yf, '
Estimated BC', y(end, 1));
fprintf(fileID,'%s %4.4f %s %5.4f \r\n', '% initial guess = ',
IG, 'improved guess', IGf);
fprintf(fileID,'%s %4i \r\n', '% Interval iterations = ',
output.intervaliterations);
fprintf(fileID,'%s %4i \r\n', '% Iteration number = ',
output.iterations);
fprintf(fileID,'%s \r\n', '% ');
fprintf(fileID,'%s %5.4f \r\n', '% J_min = ', JJ);
fprintf(fileID,'%s \r\n', '% ');
fprintf(fileID,'%s t x(t) dx(t)/dt \r\n');

```

```

fprintf(fileID, '%5.4f %10.4f %10.4f \r\n', z);
fclose(fileID);
%y(end,1)
%output.intervaliterations
%output.iterations
plot(x, y(:,1), x, UU, 'r-', x, LL, 'k-');
xlabel('t');
ylabel('x(t), u(t), lambda(t)');
title('Example 5.4');
legend('State', 'Control', 'lambda', 'location', 'best')
% subplot(1,2,1)
% plot(x, y(:,1))
% subplot(1,2,2)
% plot(x, y(:,2))
function dydx = odeIMS(t, y)
    dydx = [y(2); -2*y(1)*y(2) + y(2)^2 + 3*y(1)^2 + 2*y(1)^3 + 2*y(1)];
end
function res = resFun(dyguess)
    Yi = 0.05;
    Yf = 0;
    Ti = 0;
    Tf = 3;
    [t,y] = ode45(@odeIMS, [Ti Tf], [Yi dyguess]);
    res = y(end,1) - Yf;
end

```