Marco Aurélio Schmitz de Aguiar

AN AUGMENTED LAGRANGIAN METHOD FOR OPTIMAL CONTROL OF CONTINUOUS TIME DAE SYSTEMS

Dissertation presented to the Graduate Program in Automation and Systems Engineering in partial fulfillment of the requirements for the degree of Master in Automation and Systems Engineering.

Advisor: Prof. Eduardo Camponogara, Ph.D.
Co-advisor: Prof. Bjarne Anton Foss, Ph.D.

Florianópolis

2016

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

de Aguiar, Marco Aurélio Schmitz
AN AUGMENTED LAGRANGIAN METHOD FOR OPTIMAL CONTROL OF CONTINUOUS TIME DAE SYSTEMS / Marco Aurélio Schmitz de Aguiar ; orientador, Eduardo Camponogara ; coorientador, Bjarne Foss. - Florianópolis, SC, 2016.
194 p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico. Programa de Pós-Graduação em Engenharia de Automação e Sistemas.

Inclui referências

1. Engenharia de Automação e Sistemas. 2. Controle Ótimo. 3. Controle de Sistemas DAE. 4. Cálculo Variacional. 5. Otimização Dinâmica. I. Camponogara, Eduardo. II. Foss, Bjarne. III. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia de Automação e Sistemas. IV. Título.

Marco Aurélio Schmitz de Aguiar

AN AUGMENTED LAGRANGIAN METHOD FOR OPTIMAL CONTROL OF CONTINUOUS TIME DAE SYSTEMS

This Dissertation is recommended in partial fulfillment of the requirements for the degree of "Master in Automation and Systems Engineering", which has been approved in its present form by the Graduate Program in Automation and Systems Engineering.

Florianópolis, June 21th 2016.

Prof. Daniel Coutinho, Ph.D
Graduate Program Coordinator
Universidade Federal de Santa Catarina

Dissertation Committee:

Prof. Eduardo Camponogara, Ph.D.
Advisor
Universidade Federal de Santa Catarina

Prof. Amit Bhaya, Ph.D.
Universidade Federal do Rio de Janeiro

Prof. Alexandre Trofino Neto, Ph.D.
Universidade Federal de Santa Catarina

Prof. Hector Bessa Silveira, Ph.D.
Universidade Federal de Santa Catarina

Marcelo Lopes de Lima, Ph.D.
Petrobras (Videoconference)

ACKNOWLEDGEMENTS

This dissertation is the concluding work of my two and a half years at a master's program. It is the result of a lot learning and effort, which made me a better researcher.

First, I would like to thank my advisor Prof. Eduardo Camponogara for being an inspiration on pushing the boundaries of my knowledge for all this years. Also, I would like to mention my co-advisor Prof. Bjarne Foss, which I'm very grateful for helping in the beginning of this research.

I would like to thank my family for all the support. I am very grateful to my partner Carolina who was always on my side. I thank my friends Ricardo and Mauricio, for relaxing times after a long day. I am also grateful to my research colleagues, and now friends, Thiago, Caio, Eduardo, Leonardo, Lauvir, Bruno, Angelo, and Luiz. With special mention to Eduardo who helped with the thesis' LaTeX class, and Caio and Bruno who helped with last hour corrections of a incorrect proof.

Last but not least, I thank the Federal University of Santa Catarina and the Department of Automation and Systems for having accepted me and allowed the use of their facilities and resources. Also, I am very grateful to Petrobras for funding the research in our country.

They say “doubt everything”, but I disagree. Doubt is useful in small amounts, but too much of it leads to apathy and confusion. No, don’t doubt everything. QUESTION everything. That’s the real trick. Doubt is just a lack of certainty. If you doubt everything, you’ll doubt evolution, science, faith, morality, even reality itself - and you’ll end up with nothing, because doubt doesn’t give anything back. But questions have answers, you see. If you question everything, you’ll find that a lot of what we believe is untrue. . . but you might also discover that some things ARE true. You might discover what your own beliefs are. And then you’ll question them again, and again, eliminating flaws, discovering lies, until you get as close to the truth as you can.

Questioning is a lifelong process. That’s precisely what makes it so unlike doubt. Questioning engages with reality, interrogating all it sees. Questioning leads to a constant assault on the intellectual status quo, where doubt is far more likely to lead to resigned acceptance. After all, when the possibility of truth is doubtful (excuse the pun), why not simply play along with the most convenient lie?

Questioning is progress, but doubt is stagnation.

Extracted from The Talos Priciple (video-game), 2014

RESUMO

Esta dissertação apresenta um algoritmo para resolver problemas de controle ótimo (OCP) de equações algébrico diferenciais (DAE) com base no método de Lagrangiano aumentado. O algoritmo relaxa as equações algébricas e resolve uma sequência de OCPs de equações diferenciais ordinárias (ODE). Os principais benefícios desta abordagem são dois. Em primeiro lugar, as variáveis de estado e as variáveis algébricas podem ter restrições limitantes, mesmo quando os métodos de solução utilizados são indiretos. Em segundo lugar, através da redução do sistema para um ODE, a representação é mais compacta e o OCP pode ser tratado por métodos computacionalmente mais eficientes. Provas matemáticas apresentadas mostram que o algoritmo converge para o valor do objetivo do OCP original e a violação da equação algébrica relaxada vai para zero. Estas propriedades são confirmadas com experimentos numéricos.

Palavras-chave: Controle Ótimo. Controle de Sistemas DAE. Cálculo Variacional. Otimização com Lagrangiano Aumentado. Otimização Dinâmica.

RESUMO ESTENDIDO

Esta dissertação tem como objetivo desenvolver um algoritmo para controle ótimo de sistemas descritos com equações algébrico diferenciais. Para atingir esse objetivo, foram compilados em dois capítulos os fundamentos da teoria de sistemas dinâmicos e da teoria de controle ótimo.

Referente a sistemas dinâmicos é dada a definição de equações diferenciais ordinárias e de equações algébrico diferenciais. Problemas que envolvem estas classes de equações são apresentados, sendo eles os problemas de valor inicial e os problemas de valor de contorno. Dois métodos para resolver estes problemas são apresentados, o método de colocação e o método de múltiplos tiros. O método dos múltiplos tiros requer a análise de sensibilidade, a qual é apresentada também neste capítulo.

Quanto à compilação sobre controle ótimo, é apresentado o cálculo variacional que serve de pilar para a teoria de controle ótimo. O caso mais simples de controle ótimo é apresentado com uma condição necessária de otimalidade. Na sequência, este caso simples é estendido para que inclua tanto restrições no final do período de integração como limites para a ação de controle. As condições necessárias de otimalidade para estes casos estendidos também são apresentadas. Para o caso com limites para a ação de controle, as condições são como o princípio de mínimo de Pontryagin. Uma condição suficiente para otimalidade de problemas de controle ótimo é apresentada. Esta condição é conhecida como equação de Hamilton-Jacobi-Bellmann (HJB). As condições necessárias apresentadas são estendidas para problemas de controle ótimo de equações algébrico diferenciais. Este capítulo termina com a apresentação de métodos numéricos diretos e indiretos para a solução de problemas de controle ótimo. Estes métodos apresentados são ilustrados com um exemplo prático utilizando o oscilador de Van der Pol.

Por fim, o algoritmo é apresentado. Ele se baseia no método do Lagrangiano aumentado, que é um método para solução de problemas de otimização restrita. Este algoritmo relaxa a equação algébrica de um problema de controle ótimo de sistemas descritos por equações algébrico diferenciais, para resolvê-lo através de uma sequência de problemas de controle ótimo de equações diferenciais ordinárias, estes chamados de problemas auxiliares. Sobre este algoritmo, três propriedades são provadas matematicamente. Primeiro é mostrado que a sequência de soluções obtidas pelo algoritmo converge para o ótimo global quando, em cada iteração, os problemas auxiliares são resolvidos até a otimalidade global. Caso os problemas auxiliares sejam resolvidos

até otimalidade local, as soluções obtidas pelo algoritmo convergem até otimalidade local. Convergência global e local são difíceis de serem obtidas numericamente, para isto a terceira propriedade garante que se as soluções de cada iteração estão perto o suficiente da otimalidade e a distância até a otimalidade diminui a cada iteração, o algoritmo converge para o ótimo. Além disto, todas estas propriedades garantem que a violação da restrição algébrica relaxada vai para zero com a convergência da solução. Para verificar que o algoritmo funciona na prática foi implementado um experimento numérico utilizando o oscilador de Van der Pol. Para verificar a flexibilidade do método, o experimento utilizou métodos diretos e indiretos, juntamente com o método dos múltiplos tiros e o método de colocação. Para todos os casos o algoritmo se comportou como esperado, mostrando que as propriedades matemáticas são válidas na prática.

Palavras-chave: Controle Ótimo. Controle de Sistemas DAE. Cálculo Variacional. Otimização com Lagrangiano Aumentado. Otimização Dinâmica.

ABSTRACT

This dissertation presents an algorithm for solving optimal control problems (OCP) of differential algebraic equations (DAE) based on the augmented Lagrangian method. The algorithm relaxes the algebraic equations and solves a sequence of OCPs of ordinary differential equations (ODE). The major benefits of this approach are twofold. First, the state and algebraic variables can be bound constrained, even when the solution methods are indirect. Second, by reducing the system to an ODE, the representation is more compact and can be handled by computationally efficient methods. Mathematical proofs are developed showing that the algorithm converges to the objective value of the original OCP and the violation of the relaxed algebraic equation goes to zero. These properties are confirmed with numerical experiments.

Keywords: Optimal Control. Control of DAE Systems. Variational Calculus. Augmented Lagrangian Optimization. Dynamic Optimization.

CONTENTS

1 INTRODUCTION

Optimal control is a subfield of the control theory that tries to establish the control trajectory by minimizing the cost of the system dynamics induced by the control signals. Given the nature of which the control theory is embedded, this minimization needs to take into account not only the instantaneous cost but the cost inferred by the dynamics of the system during some time frame, namely the optimization horizon.

The most common approach for describing these dynamic systems is with ordinary differential equations (ODE). ODEs are quite convenient for developing linear controllers, model predictive controllers, and other applications. Sometimes the ODEs may become too complex that the connection with the physical meaning starts to fade. There, the differential-algebraic equations (DAE) show their advantages. The DAE unite the ODE with algebraic equations, therefore the interpretation of the variables are kept when putting different systems together.

The use of DAE for modeling dynamical systems is advantageous. However optimal control problems (OCPs) that use DAEs have less developed supporting tools and incur a greater computation cost, when compared to OCPs of ODE systems. In this context, this dissertation investigates a manner to preserve the physical meaning provided by the representation with DAE systems while being able to use the tools that are developed for ODEs.

In the optimization field, the Augmented Lagrangian method [1] obtains a solution to a constrained optimization problem by solving a sequence of unconstrained optimization problems that, in general, are more easily solved. Depending on the problem structure, each unconstrained optimization problem can be divided into subproblems that can be solved in a distributed fashion, for instance using the Alternating Direction Multiplier Methods (ADMM) [2]. These properties, allied to the advances in parallel computing of the past decades, have fostered applications of Augmented Lagrangian methods in several disciplines.

In particular, in control engineering, augmented Lagrangian methods have been applied in discrete-time model predictive control (MPC) [3] and discrete-time nonlinear model predictive control (NMPC) [4]. In these domains, the augmented Lagrangian enabled the distributed solution of MPC and NMPC problems in discrete time.

Unlike in discrete-time control, the use of augmented Lagrangian methods to solve optimal control problems (OCP) in continu-

ous-time systems is much less developed. However, adapting constrained optimization methods for optimal control is not a novel idea. In [5], the interior-point method for constrained optimization was adapted to solve OCP of a system of ordinary differential equations (ODE) with inequality constraints, and also applied in [6] to solve OCPs of differential algebraic equations (DAE) with inequality constraints.

To this end, this work contributes to the field of optimal control by proposing an augmented Lagrangian method for optimal control of DAEs, accounting for constraints in states, algebraic, and control variables. The algorithm obtains the solution of the OCP of a DAE by solving a sequence of OCPs, in which the algebraic equations are relaxed and penalized in the objective and the DAE is recast as an ODE. Finally, mathematical properties were developed for the algorithm, including proofs of global and local convergence. To achieve this goals, a series of concepts and methods were studied and here are presented as background to facilitate the understanding of the contributions of this dissertation.

1.1 HOW TO READ THIS DISSERTATION

Since this dissertation goes through several fields, some of them might be known by the reader and can be skipped. Notice, however, that some of the definitions and theorems might be required for later development.

Chapter 2 presents ordinary differential equations (ODE), in Section 2.1, and differential-algebraic equations, in Section 2.2. The problems involving ODE and DAE, which are initial value problems (IVPs) and boundary value problems (BVPs), are presented in Section 2.3. To solve BVPs, the shooting methods are presented in Section 2.4. Section 2.5 introduces sensitivity analysis, which are fundamental to the shooting methods. As an alternative to the shooting methods, the collocation method is given in Section 2.6.

Chapter 3 starts by introducing variational calculus (Section 3.1), which is the foundation of the optimal control theory. The simplest optimal control is discussed in Section 3.2. The Pontryagin's minimum principle, which gives the necessary optimality conditions for OCPs with bounded controls, is presented in Section 3.3. The sufficient optimality conditions of an OCP are presented in Section 3.4. The necessary conditions for optimality are extended to OCPs of DAE systems in Section 3.5. Sections 3.6 and 3.7 present the ap-

plication of indirect and direct methods to OCP of an illustrative system. If after reading a section, the theory is not clear, the reader is recommended to take a look at the respective example in Section 3.6.

Chap 4 presents the main contributions of this work. Section 4.2 gives an overview of the proposed algorithm. Some mathematical properties of the algorithm are presented and proved in Section 4.3. Numerical experiments with the algorithm are performed in Section 4.6.

Chap 5 concludes this work with a brief conclusion and suggestions for future works.

1.2 NOTATION

If $x \in \mathbb{R}^{N_x}$ is a (column) vector, we present x by

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{N_x} \end{bmatrix} \tag{1.1}$$

where x_i is the i-th element of the vector.

Likewise, let $f : \mathbb{R}^{N_x} \to \mathbb{R}^{N_f}$ be a vector valued function, the representation is given by

$$f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_{N_f}(x) \end{bmatrix} \tag{1.2}$$

where $f_i(x)$ is the i-th element of the vector-function.

In this work, the Jacobian of every function is noted using the partial derivatives notation and all (partial) derivatives are consider a particular case of the Jacobian. Therefore, the derivative of a function $f : \mathbb{R}^{N_x} \to \mathbb{R}$ with respect to a vector $x \in \mathbb{R}^{N_x}$ is a row vector,

$$\frac{\partial f}{\partial x} = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \cdots & \frac{\partial f}{\partial x_{N_x}} \end{bmatrix} \tag{1.3}$$

The derivative of a vector-function $f : \mathbb{R} \to \mathbb{R}^{N_f}$ with respect to a scalar variable $x \in \mathbb{R}$ is a column vector,

$$\frac{\partial f}{\partial x} = \begin{bmatrix} \frac{\partial f_1}{\partial x} \\ \frac{\partial f_2}{\partial x} \\ \vdots \\ \frac{\partial f_{N_f}}{\partial x} \end{bmatrix} \tag{1.4}$$

The Jacobian of a vector-function $f : \mathbb{R}^{N_x} \to \mathbb{R}^{N_f}$ with respect to a vector $x \in \mathbb{R}^{N_x}$ is a matrix in the form

$$\frac{\partial f}{\partial x} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_{N_x}} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_{N_x}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_{N_f}}{\partial x_1} & \frac{\partial f_{N_f}}{\partial x_2} & \cdots & \frac{\partial f_{N_f}}{\partial x_{N_x}} \end{bmatrix} \tag{1.5}$$

which is also known as the Jacobian.

2 DYNAMIC SYSTEMS

As an introduction to dynamic systems, this chapter will cover the basics of non-controlled systems, however all the content is applicable to controlled systems. Specifically, the chapter gives a review of ordinary differential equations (ODE), differential algebraic equations (DAE), mathematical problems associated with these types of equations (boundary value problems and initial value problems), and methods for solving these problems. Those with knowledge in these subjects do not need to read this chapter.

2.1 ORDINARY DIFFERENTIAL EQUATION (ODE)

Differential equations come from the mathematical area in which the behavior of the variables are described by the ratio of change, *e.g.* $dx/dt = -x$, instead of using a function that describes the relation between two variables algebrically, *e.g.* $x(t) = t^2$.

Take as an example the evolution of the velocity of a particle in free fall. According to Physics laws, the velocity of a particle in free fall satisfies the equation

$$v_f = v_0 + g(t_f - t_0), \tag{2.1}$$

where t_0 and t_f are the initial and final time, v_0 and v_f are the velocities at the initial and final time, and g is the gravitational acceleration.

The same movement can be described by an ordinary differential equation. Intuitively, the acceleration is the rate of change of the velocity. Thus, (2.1) can be expressed in the form

$$(v_f - v_0) = g(t_f - t_0). \tag{2.2}$$

Let d represent the operator that correspond to an infinitesimal change. Then dt is a small change in time, which leads to the small change in the velocity dv,

$$dv = g\,dt \tag{2.3}$$

or in the more usual form

$$\frac{dv}{dt} = g. \tag{2.4}$$

Note that, by integrating (2.3) and evaluating it at the initial and final values,

$$\int_{v_0}^{v_f} dv = \int_{t_0}^{t_f} g\, dt, \tag{2.5a}$$

$$v\Big|_{v_0}^{v_f} = gt\Big|_{t_0}^{t_f}, \tag{2.5b}$$

$$v_f = v_0 + g(t_f - t_0), \tag{2.5c}$$

equation (2.1) is recovered.

Of course, the system representation using (2.1) is easier. However, these functions are not always easy to obtain and in most physical systems the natural description comes from the rates of change. Moreover, a vast class of physical systems are described by nonlinear relations using not one, but several variables to describe the behavior of these systems. For such complex systems, the analytical solution of the differential equations cannot be obtained in practice. Thus, numerical tools are required to describe the evolution of the variables in time. Some numerical methods are presented in [7].

Usually, the explicit standard form is preferable to represent the dynamic system for its simplicity. For a system with a vector of variables $x \in \mathbb{R}^{N_x}$, where N_x is the number of elements in the vector, and a function $f : \mathbb{R}^n \to \mathbb{R}^n$ that represents the dynamics, the explicit standard form is

$$\frac{dx}{dt} = f(x) \tag{2.6}$$

where x is also called state vector and $f(x)$ is a function that describes the behavior of the state. For compactness, the term $\frac{dx}{dt}$ is commonly denoted as $\dot{x}$.

In the following system, strategies for reducing the order of the differentiation are presented so as to fit the standard form.

Example 1. *In this example, a second order ODE is reduced to the standard form and solved analytically.*

The system is an extension of the particle on free fall described previously in this dissertation. Herein, the position of the particle on free fall is considered rather than the velocity. As seen before, the object in free fall will have its velocity increased by the gravitational acceleration. At the same time, the position along the vertical axis is driven by velocity.

Let p be the particle position, v be the particle velocity, and g be the gravitational acceleration. The equation that describes the position as consequence of the action of gravity is

$$\frac{d^2p}{dt^2} = \ddot{p} = g \tag{2.7}$$

which will fit the standard form if the system is modeled using v as an intermediary variable

$$\frac{dp}{dt} = \dot{p} = v \tag{2.8a}$$

$$\frac{dv}{dt} = \dot{v} = g \tag{2.8b}$$

Solving (2.8b) *by the method of separation of variables, which consists in isolating the dv and dt terms and integrating both sides, the following results*

$$dv = g\,dt \tag{2.9a}$$

$$\int dv = \int g\,dt \tag{2.9b}$$

$$v = gt \tag{2.9c}$$

Having the solution of v as a function of time, it can be replaced in (2.8a) *in order to obtain*

$$\frac{dp}{dt} = v = gt \tag{2.10a}$$

$$dp = gt\,dt \tag{2.10b}$$

$$\int dp = \int gt\,dt \tag{2.10c}$$

$$p = \frac{gt^2}{2} \tag{2.10d}$$

Hence the solution of the position and velocity of the system is given by

$$v = gt \tag{2.11a}$$

$$p = \frac{gt^2}{2} \tag{2.11b}$$

The existence of boundary conditions, such as initial or final position and velocity, and related problems will be explained later in

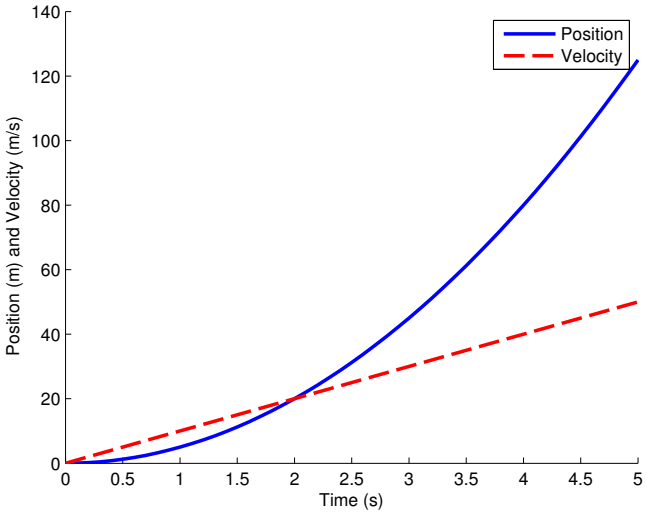

Figure 2.1: Position and Velocity of a free falling particle as function of time..

this chapter. But assuming that the initial conditions are null, Figure 2.1 gives the behavior of p and v in time.

Example 2. *This example considers a pendulum, which is a more complex system that has been used in academia for the development of advanced control techniques. The system is composed by a particle (typically represented as a sphere) and a rigid rod, as shown in Figure 2.2. At one end, the particle is connected to the rod and at the other end the rod is connected to a pivot, where the system is free to spin around.*

The particle has mass m, the rod has length ℓ, and the system is influenced by the gravitational acceleration g. The mass of the rod can be neglected for most of the cases. There are different ways to describe this system. The representation here is based on the angular position, the angular velocity, and the angular acceleration.

Let us define θ to be the angle of the intersection of the rod and the vertical axis, as depicted in Figure 2.2. Then, the equation that drives the angle θ is

$$\ddot{\theta} = \frac{g}{\ell} \sin \theta \tag{2.12}$$

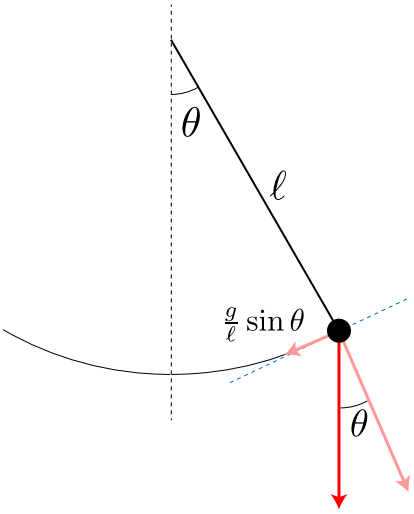

Figure 2.2: Pendulum scheme.

Let $\omega = \dot{\theta}$, then

$$\dot{\theta} = \omega \tag{2.13a}$$

$$\dot{\omega} = \frac{g}{\ell} \sin\theta \tag{2.13b}$$

So the system can be expressed in the standard form

$$\dot{x} = f(x) \tag{2.14a}$$

with

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \theta \\ \omega \end{bmatrix} \tag{2.14b}$$

$$f(x) = \begin{bmatrix} x_2 \\ \frac{g}{\ell} \sin x_1 \end{bmatrix} \tag{2.14c}$$

This is the standard form for the pendulum system. In the next section a different representation will be introduced. Both representations will be used later to simulate the pendulum behavior.

2.2 DIFFERENTIAL ALGEBRAIC EQUATION (DAE)

This section provides the definition and classification of differential algebraic equations (DAE), which can represent a great variety of physical systems. Differential algebraic equations can be seen as ordinary differential equations with algebraic constraints, which bind the dynamic of one or more variables. It is not wrong to say that the set of dynamic algebraic equations contains the set of ordinary differential equations.

To describe a system of DAE, assume $t \in [t_0, t_f]$ as the independent variable (usually representing time), $x(t) \in \mathbb{R}^{N_x}$ as the state variable, and $y(t) \in \mathbb{R}^{N_y}$ as the algebraic variable, where N_x is the number of states and N_y is the number of algebraic variables. In addition, let $f(x(t), y(t), t)$ be the dynamic vector-function and $g(x(t), y(t), t)$ be the algebraic vector-function. For the sake of readability, from now on the state variables x and the algebraic variables y will not be explicitly given as functions of t. Mathematically, the DAE can be put in the semi-explicit form as follows:

$$\dot{x} = f(x, y, t) \tag{2.15a}$$

$$0 = g(x, y, t) \tag{2.15b}$$

The $g(x, y, t)$ function might come in different forms. There are some techniques to classify the DAE regarding the constitution of the algebraic function. The classification method that is more relevant is the classification by differentiation.

The classification consists in counting the number of differentiations required to obtain a function $\hat{g}(x, y, t)$ so as to represent y by its time derivative, meaning that

$$\dot{y} = \hat{g}(x, y, t) \tag{2.16}$$

The classification of the DAE system is important because some of the numerical methods and some of the mathematical properties only apply to specific classes of DAE systems.

Example 3 (DAE Index-1)**.** *Here we will take the simplest case of a DAE system. Let x be a scalar state variable and y be a scalar algebraic variable, and the system be defined by*

$$\dot{x} = -y \tag{2.17a}$$

$$y = x \tag{2.17b}$$

By differentiating (2.17b) *with respect to t, the following is obtained*

$$\dot{y} = \dot{x} \Rightarrow \dot{y} = -y. \tag{2.18}$$

Since (2.17b) *was differentiated only once to obtain an ODE for y, the system is a DAE index-1.*

Note that for the system in Example 3, we could have used the equation $y = x$ in $\dot{x} = -y$ to obtain a simplified representation $\dot{x} = -x$. As a general rule, if the gradient of the function g with respect to y is not singular, by the implicit function theorem it is possible to find g_y that gives y as function of x and t. This modification leads to an equivalent system

$$\dot{x} = f(x, g_y(x, t), t). \tag{2.19}$$

Although this operation seems trivial for the case of Example 3, the application in more complex systems can result in a costly operation, induce numerical instabilities, and violate energy and mass conservation [7].

The process of differentiating an algebraic equation is called index reduction. The differentiation of an algebraic equation leads to a new algebraic equation with a lower index. The process is repeated until the differentiated DAE is index-1, and an index-0 is obtained. A DAE index-0 is an ODE system with no algebraic equations. The following examples will illustrate the process of reduction and classification of two DAE systems, one being index-2 and the other index-3.

Example 4 (DAE Index-2)**.** *For this example let us assume a vector state $x = [x_1\, x_2]^T$ and a scalar algebraic variable y. The system equation is*

$$\dot{x}_1 = 1 - y \tag{2.20a}$$
$$\dot{x}_2 = -x_2 + y \tag{2.20b}$$
$$0 = x_1 - x_2 \tag{2.20c}$$

Differently from the last example, the algebraic variable is not directly linked to the algebraic equation. This configures a case where it is not possible to invert the algebraic function and put y as a function of x ($\nabla_y g$ is singular).

By taking the first derivative of the algebraic equation, the following is obtained

$$\dot{x}_1 - \dot{x}_2 = 1 - y + x_2 - y = 0 \tag{2.21}$$

that results in

$$y = 0.5(1 + x_2) \tag{2.22}$$

which is differentiated once again to obtain a differential equation for y*,*

$$\dot{y} = 0.5(\dot{x}_2) \tag{2.23}$$

and therefore, we obtain

$$\dot{y} = 0.5(y - x_2) \tag{2.24}$$

Because the system had to be differentiated twice to obtain a differential equation for y*, this is a DAE index-2.*

Example 5 (DAE Index-3 - Pendulum [7])**.** *This example presents an alternative formulation for the pendulum system. Rather than modeling the pendulum using angle and angular velocity as states, this approach uses the vertical and horizontal position and velocities as states and an algebraic variable* λ*, that can be seen as the centripetal force imposed by the rod.*

The problem has the state x *as the horizontal position,* y *as the vertical position,* v_x *and* v_y *as the horizontal and vertical velocities. The system parameters are the gravitational acceleration* g *and the rod length* ℓ*. The equation that describes the pendulum dynamics is*

$$\dot{x} = v_x \tag{2.25a}$$
$$\dot{y} = v_y \tag{2.25b}$$
$$\dot{v}_x = -\lambda x \tag{2.25c}$$
$$\dot{v}_y = -\lambda y - g \tag{2.25d}$$
$$x^2 + y^2 = \ell \tag{2.25e}$$

Notice that the algebraic equation, which is a positional constraint, does not contain the algebraic variable λ*. Thus, an index reduction operation is performed, we obtain*

$$2(\dot{x}x + \dot{y}y) = 0 \Rightarrow v_x x + v_y y = 0. \tag{2.26a}$$

which is velocity constraint.

Differentiating the velocity constraint and substituting with the state differential equations, results in

$$\begin{aligned} &\dot{v}_x x + v_x \dot{x} + \dot{v}_y y + v_y \dot{y} = 0 \Rightarrow && (2.27a)\\ &\quad -\lambda x^2 + v_x^2 + (-\lambda y - g)y + v_y^2 = 0 \Rightarrow \\ &\quad -\lambda(x^2 + y^2) + v_x^2 + v_y^2 - gy = 0 \Rightarrow \\ &\lambda = \ell^{-1}(v_x^2 + v_y^2 - gy) && (2.27b) \end{aligned}$$

Which can be seen as an acceleration constraint. Using this algebraic equation, the system can be put in in the semi-explicit form (2.15)

$$\begin{aligned} \dot{x} &= v_x && (2.28a)\\ \dot{y} &= v_y && (2.28b)\\ \dot{v}_x &= -\lambda x && (2.28c)\\ \dot{v}_y &= -\lambda y - g && (2.28d)\\ \lambda &= \ell^{-1}(v_x^2 + v_y^2 - gy) && (2.28e) \end{aligned}$$

By differentiating one more time we obtain the ODE for λ:

$$\dot{\lambda} = \ell^{-1}[2(-\lambda x)v_x + 2(-\lambda y - g)v_y - gv_y] \qquad (2.29)$$

This DAE is an index-3 system because it was differentiated three times before obtaining an ODE for the algebraic variable λ.

2.3 PROBLEM TYPES

The problems related to ODE and DAE are classified by the information available. If information on the initial conditions of a system is available, the problem is classified as an initial value problem (IVP). On the other hand, if partial information on the initial condition and on the final condition of the system are given, the problem is a boundary value problem (BVP).

When only the final conditions of the system are known, the problem can be recast as an IVP by integrating backwards in time. Mathematically, having the time variable $t \in [t_0, t_f]$, a new time variable $\hat{t} \in [\hat{t}_0, \hat{t}_f]$ is defined, such that $\hat{t}_0 = t_f$ and $\hat{t}_f = t_0$. Therefore for the recast problem the initial conditions $x(\hat{t}_0)$ are given.

2.3.1 Initial Value Problem

The initial value problem (IVP) is the most frequent type of problem of ODE/DAE systems. For the majority of cases, the ini-

tial conditions are known and the goal is to find a function that describes the time evolution of the states and algebraic variables.

Definition 1 (Initial Value Problem)**.** *Let $t \in [t_0, t_f]$ be the time variable, $x \in \mathbb{R}^{N_x}$ be the state vector, $y \in \mathbb{R}^{N_y}$ be the algebraic variables vector, f be the system dynamic function, and g be the algebraic function. Having the initial condition $x_0 \in \mathbb{R}^{N_x}$, the initial value problem (IVP) for a semi-explicit DAE system is represented as follows*

$$\dot{x} = f(x, y, t) \tag{2.30a}$$
$$0 = g(x, y, t) \tag{2.30b}$$
$$x(t_0) = x_0 \tag{2.30c}$$

In order to have the problem well defined, the number of initial conditions has to be equal to the number of states. Assuming that g has an unique solution for y with a fixed x_0, one can determine the initial conditions for the y variable.

The IVP problem can be solved analytically for some particular systems. However, for more complex systems, numerical methods are needed. Some methods for solving IVP of ODE systems are the forward Euler method, backward Euler method, explicit Runge-Kutta, and implicit Runge-Kutta [7]. Commercial solvers use these methods and some additional techniques to ensure low integration errors and stability properties. Among the commercial ODE solvers it can be cited MATLAB's *ode23* and *ode45*, and Sundials' *CVODES*. Sundials also offers the DAE solver *IDAS*.

The following example demonstrates how to solve an IVP analytically for a simple linear system.

Example 6 (IVP - Linear System)**.** *Using Definition 1, the time variable is $t \in [0, 10]$, $x(t)$ is a scalar, $x_0 = 1$ is the initial condition, and $f(x) = -x$ is the system dynamic. The IVP is set*

$$\dot{x} = -x \tag{2.31a}$$
$$x(0) = 1 \tag{2.31b}$$

By manipulating the first equation, the following equation is retrieved

$$\frac{dx}{x} = -dt \tag{2.32}$$

and the integration of both sides produces

$$x(t) = ce^{-t} \tag{2.33}$$

where c is an unknown coefficient. To obtain the value for c the system equation is evaluated at the initial time $t = 0$,

$$x(0) = ce^{-0} = 1 \Rightarrow c = 1. \tag{2.34}$$

The function that describes the state over t with this particular initial condition is

$$x(t) = e^{-t} \tag{2.35}$$

Notice that the evolution of the system depends not only on the ODE and DAE equations but also in the initial conditions for the system.

2.3.2 Boundary Value Problem

The boundary value problem (BVP) is the class of problems in which conditions are imposed on both boundaries of time. Here a brief explanation is given, but a full description of this problem with solution methods can be found in [8].

Definition 2 (Boundary Value Problem (BVP))**.** *Let $t \in [t_0, t_f]$ be the time variable, $x \in \mathbb{R}^{N_x}$ be the state vector, $y \in \mathbb{R}^{N_y}$ be the vector of algebraic variables, f be the dynamics function, and g be the algebraic function. Let h_0 be a function that is zero when the initial conditions are met, and h_f a function that is zero when the final conditions are met.*

$$\dot{x} = f(x, y, t) \tag{2.36a}$$
$$0 = g(x, y, t) \tag{2.36b}$$
$$h_0(x(t_0)) = 0 \tag{2.36c}$$
$$h_f(x(t_f)) = 0 \tag{2.36d}$$

The majority of the BVPs have the boundary conditions

$$x(t_0) = x_0 \tag{2.37a}$$
$$x(t_f) = x_f \tag{2.37b}$$

where

$$h_0(\mathrm{x}) = \mathrm{x}_i - x_{0,i} \qquad i \in X_0 \tag{2.38a}$$
$$h_f(\mathrm{x}) = \mathrm{x}_i - x_{f,i} \qquad i \in \{1, \ldots, N_x\} \setminus X_0 \tag{2.38b}$$

where X_0 is the set of index of states that have initial conditions.

This problem is considerably more difficult to solve than an IVP, in particular when using numerical methods. The numerical methods that solve this kind of problem will be introduced later on this chapter. But to understand the application of a BVP, the following example solves a BVP analytically.

Example 7 (Free fall BVP)**.** *Using Definition 2, let $t \in [0, 10]$ be time variable, the vertical position x_1 and the vertical velocity x_2 be the state, $x_{0,1} = -10$ be initial condition (initial position) and $x_{f,2} = 20$ be final condition (final velocity), the parameter $g = 10$ be the gravitational acceleration, and*

$$f(x) = \begin{bmatrix} x_2 \\ -g \end{bmatrix}. \tag{2.39}$$

The BVP is defined as

$$\dot{x}_1 = x_2 \tag{2.40a}$$
$$\dot{x}_2 = -g \tag{2.40b}$$
$$x_{0,1} = -10 \tag{2.40c}$$
$$x_{f,2} = 20 \tag{2.40d}$$

Some questions arise for this BVP problem:

- *What is the value for the final condition of x_1?*
- *What is the value for the initial condition of x_2?*
- *What are the functions that describe the behaviors of x_1 and x_2 over time?*

The following procedure gives an analytical approach to answer these questions.

By solving the second ODE, the following result is obtained

$$x_2(t) = -gt + c_1 \tag{2.41}$$

and by applying for the final time,

$$x_2(10) = -10g + c_1 = 20 \implies c_1 = 120. \tag{2.42}$$

This gives the initial condition $x_2(0) = 120$, and the function for $x_2(t)$,

$$x_2(t) = -gt + 120. \tag{2.43}$$

By replacing (2.43) *in the first ODE, the following ODE is obtained*

$$\dot{x}_1 = -gt + 120 \tag{2.44}$$

whose solution is

$$x_1(t) = c_2 + 120t - \frac{gt^2}{2}. \tag{2.45}$$

Finally, by applying to the initial time, the particular solution is obtained

$$x_1(0) = c_2 + 120 \times 0 - 5 \times 0^2 = -10 \implies c_2 = -10. \tag{2.46}$$

So, the position x_1 *can be described over time by*

$$x_1(t) = -10 + 120t + 5t^2. \tag{2.47}$$

From the evaluation (2.47) *at the final time, it is obtained the final condition* $x_1(10) = 2690$.

A boundary value problem can be reduced to an initial value problem if we manage to find the unknown initial conditions. The problem of finding such unknown initial conditions can be formulated as a nonlinear equation. The shooting method, that will be presented in the following section, is a method that takes advantage of this idea to solve BVPs.

2.4 SHOOTING METHODS

The shooting methods are a class of methods for solving mathematical problems, commonly nonlinear systems of equations, which include a DAE system of equations that has to be solved numerically [9]. These methods can be better understood from an analogy with archery.

Imagine that you want to hit the center of a target using a bow and arrow. Let us say that the position of the bow is fixed and you are able to choose the amount that the string will be drawn for each shot. For the first shot you will mostly likely miss the target but with the information obtained you will be able to give a better shot with the next arrow. If the arrow falls before the target, the next shot will have the string more tensioned. On the other hand, if the arrow passes over the target the tension on the string should be reduced. After some number of arrows you eventually hit the center of the target.

In the same fashion, the shooting methods can be used to solve boundary value problems. Some of the initial conditions are known, as the position of the bow being fixed. Some of the final conditions are known, as we wish to place the arrow at the center of the target. The system of DAE can be seen as the dynamic of the arrow from the bow to the final position. And finally, there are some unknown initial conditions which are equivalent to the tension to be imposed on the bow.

To solve the BVP, an IVP is formulated containing the system dynamics and the known initial condition and a guess of the unknown initial condition. A nonlinear equality if formulated so that the final condition of the IVP is equal to the final condition given by the BVP. The shooting methods solve a sequence of IVPs, each iteration getting closer to the solution of the nonlinear equation. Here the method is explained for obtaining the solution of a BVP, however it can be used for solving optimal control problems as will be explained later in Sections 3.6 and 3.7.

Let us define a function F that given an initial condition $\widehat{x}_0 \in \mathbb{R}^{N_x}$ and a time interval T, an IVP of DAE system is solved and the final condition $x(t_f)$ is returned. The function is defined by

$$F(\widehat{x}_0, T) = \left\{ x(t_f) \text{ subject to } \begin{array}{c} \dot{x} = f(x, y, t), \\ 0 = g(x, y, t), \\ x(t_0) = \widehat{x}_0 \\ t \in T = [t_0, t_f] \end{array} \right\} \tag{2.48}$$

In addition, let us define a function G that takes a vector $\widehat{x}_0 \in \mathbb{R}^{N_x}$ and a vector $\widehat{x}_f \in \mathbb{R}^{N_x}$, such that the roots of G are achieved when the boundary conditions are met. Mathematically,

$$G(\widehat{x}_0, \widehat{x}_f) = \begin{bmatrix} h_0(\widehat{x}_0) \\ h_f(\widehat{x}_f) \end{bmatrix} \tag{2.49}$$

where for most of the cases,

$$G(\widehat{x}_0, \widehat{x}_f) = \left[\begin{array}{ll} \widehat{x}_{0,i} - x_{0,i}, & i \in X_0 \\ \widehat{x}_{f,i} - x_{f,i}, & i \in \{1, \ldots, N_x\} \setminus X_0 \end{array} \right] \tag{2.50}$$

Then, the solution of a BVP can be expressed by the following nonlinear system of equations

$$\widehat{x}_f = F(\widehat{x}_0, T) \tag{2.51a}$$
$$G(\widehat{x}_0, \widehat{x}_f) = 0 \tag{2.51b}$$

which has to be solved with respect to $\widehat{x}_0$. Let the solution of the nonlinear equation be $\widehat{x}_0^*$, then the BVP has the same solution of an IVP with (2.36a) and (2.36b), with the known initial conditions $x(t_0) = \widehat{x}_0^*$. The nonlinear equation can be solved using some of the several methods available for solving nonlinear equations, being the Newton's method the most common.

The procedure described is known as the single shooting method and it is formalized in the following.

Definition 3 (Single Shooting Method)**.** *Let x_0 be the known initial condition and x_f the known final condition. Given a function F that represents the solution of the IVP as described in* (2.48)*, which takes as inputs the initial condition $\widehat{x}_0$ and an integration interval T. The single shooting method follows the steps below:*

1. *Choose a starting vector for the unknown initial conditions $\widehat{x}_0^{(0)}$.*

For $j = 0, 1, 2, \ldots$:

2. *Compute $\widehat{x}_f^{(j)} = F(\widehat{x}_0^{(j)}, T)$ and $\frac{\partial F}{\partial \widehat{x}_0}(\widehat{x}_0^{(j)}, T)$.*

3. *If $\|G(\widehat{x}_0^{(j)}, \widehat{x}_f^{(j)})\| < \varepsilon$, for some tolerance ε, a solution is found.*

4. *Calculate the next initial condition $\hat{x}_0^{(j+1)}$ that reduces $\|G(\widehat{x}_0, \widehat{x}_f)\|$ using $F(x_0, \widehat{x}_0^{(j)})$, $\frac{\partial F}{\partial \widehat{x}_0}(\widehat{x}_0^{(j)}, T)$, $\frac{\partial G}{\partial \widehat{x}_0}(\widehat{x}_0^{(j)}, \widehat{x}_f^{(j)})$, and $\frac{\partial G}{\partial \widehat{x}_f}(\widehat{x}_0^{(j)}, \widehat{x}_f^{(j)})$.*

The procedure given in Definition 3 is a mere schematic, in practice the solution of the nonlinear system (2.51) is performed by an off-the-shelf nonlinear solver.

The single shooting method is illustrated by the next example, where a BVP is solved for a multivariate linear system using the single shooting method. Differently from the common practice with this method, the IVP is solved analytically rather than numerically.

Example 8 (Multivariable System)**.** *Consider a system in the time interval $t \in [0, 5]$, having the state variables x_1 and x_2, the initial conditions $x_{0,1} = 3$, the final condition $x_{f,2} = -\frac{1}{4}$, with the system*

equation given by

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \underbrace{\begin{bmatrix}-1/2 & 1/4\\ -1/2 & -1/8\end{bmatrix}}_{A}\begin{bmatrix}x_1\\ x_2\end{bmatrix} \tag{2.52a}$$

$$x_1(0) = x_{0,1} = 3 \tag{2.52b}$$

$$x_2(5) = x_{f,2} = -\frac{1}{4} \tag{2.52c}$$

where $A \in \mathbb{R}^2$ is the given constant real matrix. The solution of the ODE is given by the formula

$$\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix} = e^{At}C, \quad \text{where} \quad C = \begin{bmatrix}c_1\\ c_2\end{bmatrix} \tag{2.53}$$

where C is a vector of constants that define a particular solution that is consistent with the initial and final conditions, and e^{At} is the matrix exponential, which is defined by the infinite series $e^X = \sum_{k=0}^{\infty} \frac{1}{k!}X^k$. However using Sylvester's formula it is possible to obtain the value of the infinite sum [10]. Evaluating this solution for $t = 0$ with the initial condition $\widehat{x}_0$, results in

$$\begin{bmatrix}x_1(0)\\ x_2(0)\end{bmatrix} = e^{A\times 0}C = \begin{bmatrix}\widehat{x}_{0,1}\\ \widehat{x}_{0,2}\end{bmatrix} \implies C = \begin{bmatrix}\widehat{x}_{0,1}\\ \widehat{x}_{0,2}\end{bmatrix}. \tag{2.54}$$

Substituting C in (2.53)*, we obtain the F function*

$$\widehat{x}_f = F(\widehat{x}_0, [0,5]) = e^{A\times 5}\begin{bmatrix}\widehat{x}_{0,1}\\ \widehat{x}_{0,2}\end{bmatrix}, \tag{2.55}$$

and using the knowledge of initial and final boundary conditions, we define the function G

$$G(\widehat{x}_0, \widehat{x}_f) = \begin{bmatrix}\widehat{x}_{0,1} - x_{0,1}\\ \widehat{x}_{f,2} - x_{f,2}\end{bmatrix} = 0. \tag{2.56}$$

Equations (2.55) *and* (2.56) *can be put together into a system of equations, with 4 equations and 4 unknown variables. Substituting the known parameters into the system of equation results in*

$$\widehat{x}_f = e^{A\times 5}\begin{bmatrix}\widehat{x}_{0,1}\\ \widehat{x}_{0,2}\end{bmatrix} \tag{2.57a}$$

$$\widehat{x}_{0,1} - 3 = 0 \tag{2.57b}$$

$$\widehat{x}_{f,2} + \frac{1}{4} = 0 \tag{2.57c}$$

from which we conclude that $\widehat{x}_{0,1} = 3$ *and* $\widehat{x}_{f,2} = -\frac{1}{4}$*. Substituting in the first equation and applying the matrix exponential leads to*

$$\begin{bmatrix} \widehat{x}_{f,1} \\ -\frac{1}{4} \end{bmatrix} = \begin{bmatrix} -0.1157 & 0.1744 \\ -0.3487 & 0.1459 \end{bmatrix} \begin{bmatrix} 3 \\ \widehat{x}_{0,2} \end{bmatrix}. \tag{2.58}$$

Rearranging to put in the standard form of linear systems ($Ax = b$)*,*

$$\begin{bmatrix} 0.1744 & -1 \\ 0.1459 & 0 \end{bmatrix} \begin{bmatrix} \widehat{x}_{0,2} \\ \widehat{x}_{f,1} \end{bmatrix} = \begin{bmatrix} 0.3470 \\ 0.7962 \end{bmatrix} \tag{2.59}$$

which has the solution $\widehat{x}_{0,2} = 5.4582$ *and* $\widehat{x}_{f,1} = 0.6047$*.*

Having both initial conditions and the ODE system, it is possible to put the BVP as an IVP. The solution of the IVP has the form (2.53)*, substituting* C *with the obtained initial conditions, the following equation is obtained for* $x_1(t)$ *and* $x_2(t)$*,*

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} 0.5639 & 0.1802 \\ -0.3604 & 0.8341 \end{bmatrix}^t \begin{bmatrix} 3 \\ 5.4582 \end{bmatrix}. \tag{2.60}$$

For some problems, in which the period of integration is too long, the nonlinearity is too severe, or the DAE system has unstable dynamics, finding at each iteration an initial condition $\widehat{x}_0^{(j+1)}$ that reduces the distance from the final boundary conditions ($\|G\|$) might be a difficult task. Thus the single shooting method can have poor convergence. To overcome ill-convergence, the multiple shooting method breaks down the integration interval in small subintervals in such way that the final condition is not far from the initial condition. The result of this process is a set of IVPs, one for each subinterval. To ensure the continuity of the states during the whole integration interval, continuity equalities are included to the system of nonlinear equations. These equations make the initial condition of one interval to be equal to the final condition of the previous one.

The number of subintervals is given by the integer N and, assuming an equal splitting, the length of the subinterval T_i is given by

$$h_i = \frac{t_f - t_0}{N}, \tag{2.61}$$

although an uniform length distribution can be used. The final time of each subinterval is given by

$$t_i = t_{i-1} + h_i, \qquad \forall i \in \{1, \ldots, N\} \tag{2.62}$$

where t_0 is given and $t_N = t_f$. Each interval T_i is defined by

$$T_i = [t_{i-1}, t_i], \qquad i \in \{1, \ldots, N\}. \tag{2.63}$$

If we define a function F and G in the same manner that was defined for the single shooting method, we have

$$F(\widehat{x}_0, T) = \left\{ x(t_f^T) \text{ subject to } \begin{array}{c} \dot{x} = f(x, y, t), \\ 0 = g(x, y, t), \\ x(t_0^T) = \widehat{x}_0 \\ t \in T = [t_0^T, t_f^T] \end{array} \right\} \tag{2.64a}$$

$$G(\widehat{x}_0, \widehat{x}_f) = \begin{bmatrix} h_0(\widehat{x}_0) \\ h_f(\widehat{x}_f) \end{bmatrix} \tag{2.64b}$$

where t_0^T and t_f^T are the start and the end time of the interval T, the function F solves an IVP for a given DAE system with initial conditions $\widehat{x}_0$ during the interval T, and function G is equal to zero when the boundary conditions are satisfied with h_0 and h_f being the boundary conditions.

Let $\widehat{x}_0^i$ and $\widehat{x}_f^i$ be the initial and final conditions of subinterval T_i. Then the nonlinear system of equations that defines the multiple shooting method is given by

$$\widehat{x}_f^i = F(\widehat{x}_0^i, T_i) \qquad i = 1, \ldots, N \tag{2.65a}$$

$$\widehat{x}_f^{i-1} = \widehat{x}_0^i \qquad i = 2, \ldots, N \tag{2.65b}$$

$$0 = G(\widehat{x}_0^1, \widehat{x}_f^N) \tag{2.65c}$$

where the solution of the IVP for each subinterval is given by (2.65a), the continuity of the states is given by (2.65b), and the boundary conditions are enforced by (2.65c).

Definition 4 (Multiple Shooting Method)**.** *Let x_0 be the known initial condition and x_f the known final condition. For every subinterval T_i with $i \in \{1, \ldots, N\}$, where N is the number of subintervals, let $F(\widehat{x}_0, T)$ be a function that represents the solution of the IVP as given by* (2.64)*, which takes as inputs $\widehat{x}_0$ and an integration interval T. The multiple shooting method follows the steps:*

1. *Choose a starting vector of initial guess for the initial $\widehat{x}_0^{i,(0)}$, with $i \in \{1, \ldots, N\}$.*

For $j = 0, 1, 2, \ldots$:

2. *Compute* $\widehat{x}_f^{i,(j)} = F(\widehat{x}_0^{i,(j)}, T_i)$ *and* $\frac{\partial F}{\partial \widehat{x}_0}(\widehat{x}_0^{i,(j)}, T_i)$ *for all* $i \in \{1, \ldots, N\}$.

3. *If the error on the nonlinear system of equations* (2.65) *is less than some tolerance* ε*, take it as a solution of the problem.*

4. *Otherwise, find the next initial condition* $\widehat{x}_0^{i,(j+1)}$ *with* $i \in \{1, \ldots, N\}$*, using* $F(\widehat{x}_0, T_i)$ *and* $\frac{\partial F}{\partial \widehat{x}_0}(\widehat{x}_0^{i,(j)}, T_i)$*,* $\frac{\partial G}{\partial \widehat{x}_0}(\widehat{x}_0^{(j)}, \widehat{x}_f^{i,(j)})$*, and* $\frac{\partial G}{\partial \widehat{x}_f}(\widehat{x}_0^{i,(j)}, \widehat{x}_f^{(j)})$.

Notice that these methods require the partial derivatives of F with respect to initial condition $\widehat{x}_0$, which is not trivial to obtain since F relies on the solution of an IVP for which the derivatives are not defined in the traditional fashion. Some techniques for obtaining those partial derivatives are presented in the following section.

2.5 SENSITIVITY ANALYSIS [1]

This section addresses the problem of obtaining the partial derivatives of a function that depends on an IVP with respect to a parameter vector. Let this parameter vector be $p \in \mathbb{R}^{N_p}$, and the function for which we want to obtain the Jacobian to be $\Phi : \mathbb{R}^{N_x} \times \mathbb{R}^{N_y} \times \mathbb{R}^{N_p} \to \mathbb{R}^{N_\Phi}$. This function can be an objective function, a constraint for some particular time, or the final conditions for the case of the BVP.

There are different ways to obtain these derivatives. The most trivial manner is to perturbe the vector p and evaluate how the function changes. By doing so, the derivatives are approximated. However, this methodology may induce noise and incorrect derivatives, which may lead to poor convergence and numerical instabilities. On the other hand, the methods presented in this section, forward sensitivity and adjoint sensitivity, are able to obtain the Jacobian without introducing errors.

To better understand the importance of these methods, consider the following problem of specifying an electric circuit.

Example 9 (Capacitor Charge)**.** *In this example we want to design a component that delays an on/off signal in* 5 *seconds. For this, a Schmitt trigger and resistor-capacitor (RC) circuit can be combined to achieve the desired objective. A Schmitt trigger is a comparator that*

[1] This section was written based on [11]

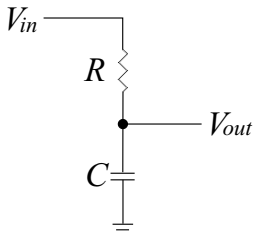

Figure 2.3: RC Circuit with one input V_{in} and one output V_{out}.

outputs $5\,V$ *if the input is greater than some voltage, in this case* $3\,V$. *The RC circuit is arranged as shown in Figure 2.3. Assuming the capacitor with fixed capacitance* $C = 100\,\mu F$, *the problem consists is to find the resistance* R *that makes the output voltage* $V_{out}(t)$ *equal to* $3\,V$ *after* 5 *seconds, meaning*

$$V_{out}(5) = 3. \tag{2.66}$$

There is an implicit dependence of $V_{out}(5)$ *and the resistance* R. *As the resistance* R *is the parameter of interest, define* $p = R$. *The dependence of* $V_{out}(5)$ *and* R *is given by the function* $\Phi(p)$. *Using* (2.66) *the definition of* Φ *is given by*

$$\Phi(p) = V_{out}(5) - 3 \tag{2.67}$$

and we want to find p^* *such that* $\Phi(p^*) = 0$. *To do so, there are two tasks to be completed:*

1. *Find an algorithm to solve the nonlinear equation* $\Phi(p) = 0$.
2. *Develop a representation of* V_{out}.

The first task can be done by Newton's Method, which is an iterative method that computes a sequence $\{p_k\}$ *of parameters that are drawn closer to* p^*. *The computation of* p_{k+1} *is given by*

$$p_{k+1} = p_k - \frac{\Phi(p_k)}{\Phi'(p_k)}, \tag{2.68}$$

where $\Phi'(p_k)$ *is the derivative of* Φ *with respect to* p.

At the same time that Newton's method gives a manner to solve the nonlinear equation $\Phi(p) = 0$ *efficiently, it requires the derivative* $\Phi'(p)$.

The open problem consists of finding how V_{out} changes over time and how the change of the resistance R will affect the voltage V_{out} after charging the capacitor for the period $t \in [0, 5]$.

The variables are the output voltage V_{out}, the capacitance $C = 100\,\mu F$, and the input voltage $V_{in} = 5\,V$. Assuming that the initial voltage on the capacitor is zero, the system equation for the capacitor charging is given by

$$\dot{V}_{out} = \frac{V_{in} - V_{out}}{RC} \tag{2.69a}$$

$$V_{out}(0) = 0. \tag{2.69b}$$

Using Definition 1, the IVP can be rewritten as

$$\dot{x} = \frac{V_{in} - x}{pC} \tag{2.70}$$

$$x(0) = 0 \tag{2.71}$$

where $x = V_{out}$ and $p = R$.

To obtain $V_{out}(5)$, the IVP is solved analytically. The solution is given by

$$x(t) = V_{in}(1 - e^{-\frac{t}{pC}}). \tag{2.72}$$

Then, by introducing the given parameter values the following function is obtained

$$x(t) = 5(1 - e^{-\frac{t}{p \times 10^{-4}}}). \tag{2.73}$$

Having the value for $x(t)$ ($V_{out}(t)$), the function $\Phi(p)$ is retrieved

$$\Phi(p) = 5(1 - e^{-\frac{5}{p \times 10^{-4}}}) - 3. \tag{2.74}$$

The derivative for $\Phi(p)$ is given by

$$\Phi'(p) = \frac{d\Phi}{dp} = \frac{-25e^{-\frac{t}{p10^{-4}}}}{p^2 10^{-4}}. \tag{2.75}$$

Consider $p = 47\,k\Omega$ as the initial point for Newton's iterative algorithm. The voltage $V_{out}(t)$ for $t \in [0, 5]$ is plotted in Figure 2.4, which also shows the capacitor voltage for other values of the resistor.

Evaluating the derivative for $p = 47k\Omega$ results

$$\frac{d\Phi}{dp}(47 \times 10^3) = -39.06\,\frac{\mu V}{\Omega}. \tag{2.76}$$

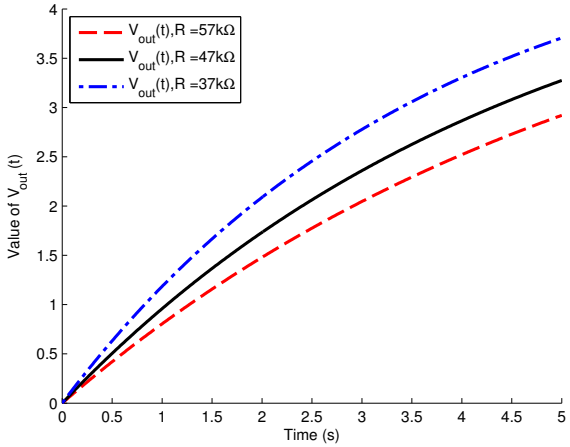

Figure 2.4: Voltage of the capacitor for $t \in [0, 5]$.

Applying Newton's Method the result obtained is $p = 54.56\,k\Omega$. *Figure 2.5 shows how the value of* R *affects the capacitor voltage* V_{out} *at time* $t = 5\,s$ *and the derivative for the function at point* R. *It can be noticed that this curve has a nonlinear behavior.*

In the following, formal approaches will be presented for obtaining the derivative $\frac{d\Phi}{dp}$ without the analytical solution of the IVP.

2.5.1 Forward Sensitivity

The forward sensitivity calculation is a method for obtaining the derivatives based on a numerical simulation. The forward denomination comes from the fact that the derivatives are calculated in the positive direction of the time axis.

Let us consider a DAE of index-1 system, with the state $x \in \mathbb{R}^{N_x}$, the algebraic variables $y \in \mathbb{R}^{N_y}$, the time $t \in [t_0, t_f]$, and a vector $p \in \mathbb{R}^{N_p}$ of decision parameters for which we want to obtain the derivatives. Let the functions f and g be continuously differentiable with respect to all their arguments. Let the initial state

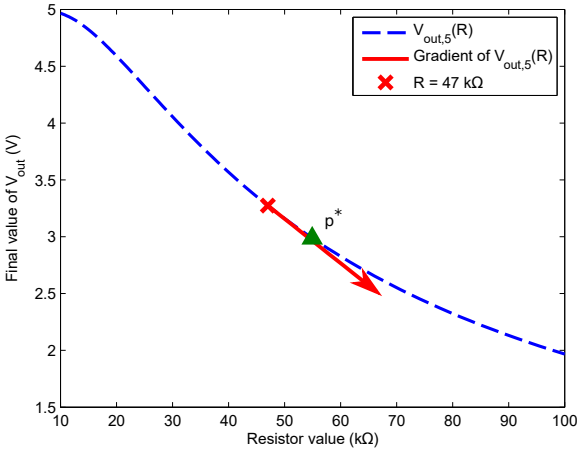

Figure 2.5: Final voltage of the capacitor as a function of the resistor R and the gradient vector at $R = 47\,k\Omega$.

be $x_0 \in \mathbb{R}^{N_x}$. Then an IVP problem can formulated

$$\dot{x} = f(x, y, t, p) \tag{2.77a}$$
$$0 = g(x, y, t, p) \tag{2.77b}$$
$$x(t_0) = x_0 \tag{2.77c}$$

For this system, there is a function $\Phi(x(t_f), y(t_f), p)$ for which the derivatives must be obtained.

If the system is differentiated with respect to p, the following equations are obtained

$$\frac{d\dot{x}}{dp} = \frac{df}{dp} = \frac{\partial f}{\partial x}\frac{dx}{dp} + \frac{\partial f}{\partial y}\frac{dy}{dp} + \frac{\partial f}{\partial p} \tag{2.78a}$$
$$\frac{dg}{dp} = \frac{\partial g}{\partial x}\frac{dx}{dp} + \frac{\partial g}{\partial y}\frac{dy}{dp} + \frac{\partial g}{\partial p} = 0 \tag{2.78b}$$
$$\frac{dx}{dp}(t_0) = \frac{dx_0}{dp} \tag{2.78c}$$

and by differentiating the function $\Phi(x(t_f), y(t_f), p)$

$$\frac{d\Phi}{dp} = \frac{\partial \Phi}{\partial x}\frac{dx}{dp} + \frac{\partial \Phi}{\partial y}\frac{dy}{dp} + \frac{\partial \Phi}{\partial p}. \tag{2.78d}$$

Here the dependency of x, y, and p are omitted for the sake of readability. Notice that the initial conditions might depend on the decision parameters p. First, we define matrix variables

$$S = \frac{dx}{dp} = \begin{bmatrix} \frac{dx_1}{dp_1} & \cdots & \frac{dx_1}{dp_{N_p}} \\ \vdots & \ddots & \vdots \\ \frac{dx_{N_x}}{dp_1} & \cdots & \frac{dx_{N_x}}{dp_{N_p}} \end{bmatrix} \tag{2.79a}$$

$$R = \frac{dy}{dp} = \begin{bmatrix} \frac{dy_1}{dp_1} & \cdots & \frac{dy_1}{dp_{N_p}} \\ \vdots & \ddots & \vdots \\ \frac{dy_{N_y}}{dp_1} & \cdots & \frac{dy_{N_y}}{dp_{N_p}} \end{bmatrix} \tag{2.79b}$$

where $S(t)$ is the $N_x \times N_p$ Jacobian matrix of x with respect to p; and $R(t)$ is the $N_y \times N_p$ Jacobian matrix between y and p. These variables are introduced in (2.78) to obtain the forward sensitivity, resulting in the following definition.

Definition 5 (Forward Sensitivity Calculation)**.** *Let $\Phi(x(t_f), y(t_f), p) \in \mathbb{R}^{N_\Phi}$ be a function for which we want to calculate the derivative with respect to vector $p \in \mathbb{R}^{N_p}$, where the variable $x \in \mathbb{R}^{N_x}$ is the state vector, $y \in \mathbb{R}^{N_y}$ is the algebraic vector, and $t \in [t_0, t_f]$ is the time variable. Let f, the dynamic function, and g, the algebraic function, be continuously differentiable with respect to x, y, and p. Then the derivative of Φ with respect to p at the time t_f is obtained by the following DAE system*

$$\frac{d\Phi}{dp}(t_f) = \frac{\partial \Phi}{\partial x} S(t_f) + \frac{\partial \Phi}{\partial y} R(t_f) + \frac{\partial \Phi}{\partial p} \tag{2.80a}$$

$$\frac{dS}{dt} = \frac{\partial f}{\partial x}(x, y, t, p) S(t) + \frac{\partial f}{\partial y}(x, y, t, p) R(t) + \frac{\partial f}{\partial p}(x, y, t, p) \tag{2.80b}$$

$$0 = \frac{\partial g}{\partial x}(x, y, t, p) S(t) + \frac{\partial g}{\partial y}(x, y, t, p) R(t) + \frac{\partial g}{\partial p}(x, y, t, p) \tag{2.80c}$$

$$S(t_0) = \frac{dx_0}{dp} \tag{2.80d}$$

where $S(t)$ is a $\mathbb{R}^{N_x} \times \mathbb{R}^{N_p}$ matrix and $R(t)$ is a $\mathbb{R}^{N_y} \times \mathbb{R}^{N_p}$ matrix that are given by (2.79).

Notice that

$$\frac{dS}{dt} = \frac{d}{dt}\frac{dx}{dp} \tag{2.81}$$

meaning that the derivative of x with respect to p is applied before the derivative with respect to t, while in (2.78c) the derivative with respect to t is applied before p. This change in the order of the derivative is possible because the function f is continuously differentiable, hence Schwarz's theorem is applicable [12].

To obtain the derivative of a function $\Phi(x(t_f), y(t_f), p)$, the $N_x + N_y$ equations (2.80b) and (2.80c) are included in the original DAE system creating an augmented DAE system. The reason that both systems cannot be solved apart is the need of the values of x and y for all $t \in [t_0, t_f]$ to calculate the sensitivity.

In the following, the method is illustrated using the capacitor charge example.

Example 10 (Forward Sensitivity - Capacitor Charge)**.** *Let us consider the same system from Example 9, which has the equations*

$$\dot{x} = \frac{V_{in} - x}{pC} \tag{2.82a}$$

$$x(0) = 0 \tag{2.82b}$$

being p the decision parameter and $\Phi = x(5) - 3$ the function of interest.

Let $S = \frac{dx}{dp}$ be the sensitivity of x with respect to p, where $S(t)$ a scalar since there is one state and one parameter. There is no sensitivity matrix R since there is no algebraic variable. Therefore, by applying Definition 5, the following system for the sensitivity is obtained

$$\frac{d\Phi}{dp} = \frac{d\Phi}{dx}S(5) + \frac{d\Phi}{dp} = S(5), \tag{2.83a}$$

$$\dot{S} = \frac{\partial f}{\partial x}S + \frac{\partial f}{\partial p} = \frac{-1}{pC}S - \frac{V_{in} - x}{p^2 C}, \tag{2.83b}$$

$$S(0) = 0. \tag{2.83c}$$

Solving the IVP (2.82) *together with* (2.83b) *and* (2.83c) *using numerical integration, leads to $S(5)$ which, when substituted in* (2.83a)*, gives*

$$\frac{d\Phi}{dp} = S(5) = -39.06 \times 10^{-6} \tag{2.84}$$

which has the same value obtained in Example 9, and does not involve the explicit calculation of Φ as a function of p.

The following example is broader than the former in the sense that the system is a multivariable DAE and the sensitivities are calculated for the initial condition, the dynamic function parameter, and the algebraic functions parameter.

Example 11 (Forward Sensitivity for a DAE System)**.** *Let us consider the following system*[2]

$$\dot{x}_1 = x_1^2 + x_2^2 - 3y \tag{2.85a}$$
$$\dot{x}_2 = x_1 x_2 + x_1(y + p_2) \tag{2.85b}$$
$$0 = x_1 y + p_3 x_2 \tag{2.85c}$$
$$x(0) = \begin{bmatrix} 5 \\ p_1 \end{bmatrix} \tag{2.85d}$$

where $x = [x_1\ x_2]^T$ is the state vector, y is the algebraic variable, and $t \in [0, t_f]$ is the time variable. In addition, consider the sensitivity state S_{ij} which is the sensitivity of the state x_i with respect to the parameter p_j, and the algebraic sensitivity variable R_j which is the sensitivity of the variable y with respect to the parameter p_j. Applying Definition 5 for a general Φ, the sensitivity DAE system is obtained,

$$\dot{S} = \begin{bmatrix} \dot{S}_{11} & \dot{S}_{12} & \dot{S}_{13} \\ \dot{S}_{21} & \dot{S}_{22} & \dot{S}_{23} \end{bmatrix} = \begin{bmatrix} 2x_1 & 2x_2 \\ x_2 + y + p_2 & x_1 \end{bmatrix} S + \begin{bmatrix} -3 \\ x_1 \end{bmatrix} R + \begin{bmatrix} 0 & 0 & 0 \\ 0 & x_1 & 0 \end{bmatrix} \tag{2.86a}$$

$$0 = \begin{bmatrix} y & p_3 \end{bmatrix} S + x_1 \begin{bmatrix} R_1 & R_2 & R_3 \end{bmatrix} + \begin{bmatrix} 0 & 0 & x_2 \end{bmatrix} \tag{2.86b}$$

$$\begin{bmatrix} S_{11}(0) & S_{12}(0) & S_{13}(0) \\ S_{21}(0) & S_{22}(0) & S_{23}(0) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \tag{2.86c}$$

Notice that the DAE system does not depend on the interest function Φ, which is a property that can be exploited for the cases where Φ has a high number of rows. To illustrate this advantage, the derivatives are now calculated for two functions:

[2]Extracted from Example 9.1 of [11]

- *For the first case, define the function*

$$\Phi_1 = x(t_f), \tag{2.87}$$

then the derivative is given by

$$\frac{d\Phi_1}{dp}(t_f) = I_2 S(t_f) = \begin{bmatrix} S_{11}(t_f) & S_{12}(t_f) & S_{13}(t_f) \\ S_{21}(t_f) & S_{22}(t_f) & S_{23}(t_f) \end{bmatrix} \tag{2.88}$$

where I_2 is the 2×2 identity matrix, and $\frac{d\Phi_1}{dp}$ is a $N_x \times N_p$ matrix.

- *For the second case, consider the nonlinear function*

$$\Phi_2 = \frac{1}{2} x(t_f)^T x(t_f), \tag{2.89}$$

for wich the derivative is given by

$$\begin{aligned} \frac{d\Phi_2}{dp}(t_f) &= \frac{\partial \Phi_2}{\partial x} S(t_f) \\ &= \begin{bmatrix} x_1(t_f) & x_2(t_f) \end{bmatrix} \begin{bmatrix} S_{11}(t_f) & S_{12}(t_f) & S_{13}(t_f) \\ S_{21}(t_f) & S_{22}(t_f) & S_{23}(t_f) \end{bmatrix} \end{aligned} \tag{2.90}$$

which is a $1 \times N_p$ matrix.

Summarizing, the forward sensitivity augments the original DAE system with the sensitivity variables S and R, and their respective equations. This procedure incurs the additional computational cost of calculating $N_p(N_x + N_y)$ extra DAEs.

2.5.2 Adjoint Sensitivity

The forward sensitivity has an advantage when the function of interest Φ has a large number of rows. However for problems where the vector p has high dimension, the calculation of such sensitivity can be costly, since the number of additional variables is $N_p(N_x + Ny)$. For these cases, there is a more efficient approach called the adjoint sensitivity. Differently from the forward sensitivity, the cost for calculating the adjoint sensitivity does not increase with the number of parameters, however it increases with the number of rows in the interest function Φ. This property makes the adjoint sensitivity more suitable for direct methods for optimal control, which will be seen in the next chapter.

The background theory of adjoint sensitivity relies on variational calculus, which is clarified later in Section 3.1. Herein, the resulting method is presented, while the underlying theory is omitted.

The method is named adjoint because it uses adjoint variables to calculate the derivatives. Alternatively, the method is also called as backwards sensitivity. The reason is that the method takes two steps, firstly a simulation from t_0 to t_f solves the DAE systems, secondly a backwards integration, from t_f to t_0, solves the adjoint DAE system.

Consider for now that the interest function $\Phi(x(t_f), y(t_f), p)$ is a scalar valued function. Consider the adjoint functions $\lambda : [t_0, t_f] \rightarrow \mathbb{R}^{N_x}$ and $\nu : [t_0, t_f] \rightarrow \mathbb{R}^{N_y}$. Notice that the following equality is valid if the equations of the DAE systems are satisfied,

$$\Phi(x(t_f), y(t_f), p) = \Phi(x(t_f), y(t_f), p) + \int_{t_0}^{t_f} \Big\{ \lambda(t)^T [f(x(t), y(t), p) - \dot{x}(t)] + \nu(t)^T g(x(t), y(t), p) \Big\} dt \quad (2.91)$$

Integrating by parts the term $\int_{t_0}^{t_f} -\lambda^T \dot{x}\, dt$ we obtain

$$\int_{t_0}^{t_f} -\lambda(t)^T \dot{x}\, dt = -x(t_f)^T \lambda(t_f) + x(t_0)^T \lambda(t_0) + \int_{t_0}^{t_f} \dot{\lambda}^T x(t)\, dt \quad (2.92)$$

which leads to

$$\Phi(x(t_f), y(t_f), p) = \Phi(x(t_f), y(t_f), p) - x(t_f)^T \lambda(t_f) + x(t_0)^T \lambda(t_0) + \int_{t_0}^{t_f} \Big[\lambda(t)^T f(x(t), y(t), p) + x(t)^T \dot{\lambda} + \nu^T g(x(t), y(t), p) \Big] dt \quad (2.93)$$

The derivative of function f at point x is given by

$$\frac{df}{dx} = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h} \quad (2.94)$$

where h is a small perturbation defined in the same space of x. Likewise, let δx be a perturbation on the state x, δy be a perturbation in the variable y, and δp be a perturbation in the variable p. Since

x and y are functions, their perturbations, δx and δy, are also functions. Using the calculus of variations, by perturbing the variables x, y, and p, a resulting perturbation is obtained on Φ, namely $\delta\Phi$, which is obtained with

$$\delta\Phi = \left[\frac{\partial\Phi}{\partial x}(t_f) - \lambda(t_f)^T\right]\delta x(t_f) + \lambda(t_0)^T\delta x(t_0) + \frac{\partial\Phi}{\partial p}(t_f)\delta p + \int_{t_0}^{t_f}\left\{\left[\lambda^T\frac{\partial f}{\partial x} + \dot{\lambda}^T + \nu^T\frac{\partial g}{\partial x}\right]\delta x + \left[\lambda^T\frac{\partial f}{\partial y} + \nu^T\frac{\partial g}{\partial y}\right]\delta y + \left[\lambda^T\frac{\partial f}{\partial p} + \nu^T\frac{\partial g}{\partial p}\right]\delta p\right\}dt \quad (2.95)$$

Since the interest is in determining how a perturbation δp affects $\delta\Phi$, the adjoint variables are chosen in such a way that the terms that depend on δx and δy are canceled out.

1. To avoid the perturbation in $\delta\Phi$ caused by the perturbation of the final state $\delta x(t_f)$, we define

$$\lambda(t_f) = \frac{\partial\Phi}{\partial x}^T(t_f) \quad (2.96)$$

which defines a boundary condition for λ.

2. To vanish with influence of the perturbation on the state $\delta x(t)$, we enforce

$$\dot{\lambda} = -\frac{\partial f}{\partial x}^T\lambda - \frac{\partial g}{\partial x}^T\nu \quad (2.97)$$

which gives a differential equation for λ.

3. Similarly, for the perturbation on the algebraic variable δy,

$$\frac{\partial f}{\partial y}^T\lambda + \frac{\partial g}{\partial y}^T\nu = 0 \quad (2.98)$$

which defines the algebraic adjoint variable ν.

4. Finally, we consider that the perturbation on the initial state $\delta x(0)$ depends on δp, therefore

$$\lambda(t_0)^T\delta x(t_0) = \lambda(t_0)^T\frac{\partial x}{\partial p}(t_0)\delta p \quad (2.99)$$

which can be arranged with the other terms that depend on dp.

By eliminating the terms of (2.95) that do not depend on δp, we are left with

$$\delta\Phi = \left\{ \lambda(t_0)^T \frac{\partial x}{\partial p}(t_0) + \int_{t_0}^{t_f} \left[\lambda^T \frac{\partial f}{\partial p} + \nu^T \frac{\partial g}{\partial p} \right] dt \right\} \delta p \qquad (2.100)$$

If the perturbation $\delta p \to 0$, then $\frac{\delta\Phi}{\delta p} = \frac{d\Phi}{dp}$. Therefore, the solution of the adjoint DAE system, obtained from gathering equations from item 1 to 3, allows to obtain the derivative of Φ by evaluating (2.100).

Definition 6 (Adjoint Sensitivity)**.** *Let $\Phi(x(t_f), y(t_f), p) \in \mathbb{R}^{N_\Phi}$ be a function for which we want to calculate a derivative, where the variable $x(t) \in \mathbb{R}^{N_x}$ is the state vector, $y(t) \in \mathbb{R}^{N_y}$ is the algebraic vector, $t \in [t_0, t_f]$ is the time variable, and $p \in \mathbb{R}^{N_p}$ is vector of decision parameters. Let f, the dynamic function, and g, the algebraic function, be at least once differentiable with respect to the variables x, y, and p. Then, the adjoint DAE system to obtain the derivative of the function Φ_i, with $i = 1, \ldots, N_\Phi$, is given by*

$$\dot{\lambda} = -\frac{\partial f}{\partial x}^T \lambda - \frac{\partial g}{\partial x}^T \nu \qquad (2.101a)$$

$$\frac{\partial f}{\partial y}^T \lambda + \frac{\partial g}{\partial y}^T \nu = 0 \qquad (2.101b)$$

$$\lambda(t_f) = \frac{\partial \Phi_i}{\partial x}^T (t_f) \qquad (2.101c)$$

and the derivative $\frac{d\Phi_i}{dp}$ at the time t_f is obtained by

$$\frac{d\Phi_i}{dp}(x(t_f), y(t_f), p) = \lambda(t_0)^T \frac{\partial x}{\partial p}(t_0) + \int_{t_0}^{t_f} \left[\lambda^T \frac{\partial f}{\partial p} + \nu^T \frac{\partial g}{\partial p} \right] dt \qquad (2.102)$$

To obtain the derivative requires the solution of an IVP of the adjoint DAE (2.101), and an integration in t to evaluate (2.102), both require the storage of states and algebraic variables, which can be costly. On the other hand, generally the adjoint sensitivity requires less additional variables, if compared to forward sensitivity.

Notice that if Φ is a vector-function with the value in the space $\mathbb{R}^{N_\Phi}$, then the DAE system (2.101) has to be repeated N_Φ times. For each row $i \in \{1, \dots, \}$ of Φ, the system uses a different final condition of the adjoint variable,

$$\lambda(t_f) = \frac{\partial \Phi_i}{\partial x}^T. \tag{2.103}$$

So, as a rule of thumb, for systems with a large parameter vector p, the adjoint sensitivity is a better option. However, if the dimension of Φ is far greater than the dimension of p, the forward sensitivity is preferred.

Example 12 (Adjoint Sensitivity)**.** *This example uses the same DAE system and interest function Φ_2 of Example 11. They are*

$$\dot{x}_1 = x_1^2 + x_2^2 - 3y \tag{2.104a}$$
$$\dot{x}_2 = x_1 x_2 + x_1(y + p_2) \tag{2.104b}$$
$$0 = x_1 y + p_3 x_2 \tag{2.104c}$$
$$x(0) = \begin{bmatrix} 5 \\ p_1 \end{bmatrix} \tag{2.104d}$$

with the interest function Φ_2 given by

$$\Phi_2 = \frac{1}{2} \begin{bmatrix} x_1(t_f) & x_2(t_f) \end{bmatrix} \begin{bmatrix} x_1(t_f) \\ x_2(t_f) \end{bmatrix}. \tag{2.105}$$

Using the DAE system and Definition 6, the adjoint DAE system is obtained

$$\dot{\lambda}_1 = -\left[2x_1\lambda_1 + (x_2 + y + p_2)\lambda_2\right] - y\nu, \tag{2.106a}$$
$$\dot{\lambda}_2 = -\left[2x_2\lambda_1 + x_1\lambda_2\right] - p_3\nu, \tag{2.106b}$$
$$0 = -3\lambda_1 + x_1\lambda_2 + x_1\nu. \tag{2.106c}$$

For the interest function Φ_2, the boundary conditions are

$$\lambda_1(t_f) = x_1(t_f) \tag{2.107a}$$
$$\lambda_2(t_f) = x_2(t_f) \tag{2.107b}$$

By solving the DAE system (2.106)*, the profiles for λ_1, λ_2, and ν are obtained. Then, according to* (2.102)*, derivatives can be obtained*

with

$$\frac{\partial \Phi_2}{\partial p_1} = \lambda_2(t_0) \tag{2.108a}$$

$$\frac{\partial \Phi_2}{\partial p_2} = \int_{t_0}^{t_f} x_1 \lambda_2 \, dt \tag{2.108b}$$

$$\frac{\partial \Phi_2}{\partial p_3} = \int_{t_0}^{t_f} x_2 \nu \, dt \tag{2.108c}$$

If we were going to solve for the Φ_1 function, then the DAE system would have to be solved twice. One with the initial condition with first row of the Φ_1 function, and one with the second row.

2.6 COLLOCATION METHOD[3]

A collocation method is a method for numerical solution of ODEs, PDEs, etc. The idea is to choose a finite dimensional space of candidate solutions, such as polynomials of a fixed degree, and a number of points in the domain, called collocation points, and them to choose the solution which satisfies the given equation at the collocation points.

The collocation method is a method for solving mathematical problems with a DAE system, which shares some similarities with the Runge-Kutta method for ODE system. The shooting methods, Section 2.4,are known as implicit methods because the solution of the IVP is handled by some numerical solver, which is not part of the system of nonlinear equations. The collocation method differs from those methods insofar as it is an explicit method, which approximates the solution of the IVP by a family of polynomials. The term explicit refers to the fact that there is access to the states at any time, which can be of advantage for optimal control problems. In contracts with a single shooting approach, only the states at the beginning and at the end of the integration period are available, and, in a similar fashion, the states are only available at the boundary of each subinterval for the multiple shooting method.

The collocation method, like the multiple shooting method, splits the integration interval into N subintervals. For each subinterval T_i, with $i = 1, \ldots, N$ and a length h_i, a polynomial of K-th order approximates state, algebraic, and control variables. A visual illustration of these concepts is presented in Figure 2.6.

[3] This section was written based on [11].

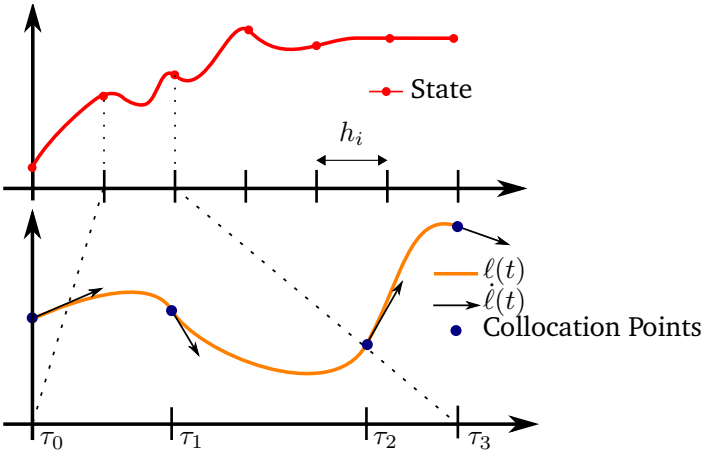

Figure 2.6: Illustration of a state profile, where ℓ is the polynomial.

The polynomial can be represented in several forms, *i.e.* power series, Newton divided differences, or B-splines. However, the Lagrangian interpolation polynomials is the most suitable method for the collocation method. This particular class of polynomials is preferred for having stability properties and null approximation error for some types of problems. Also, these polynomials are more easily expressed because there is a direct relation between the states and the polynomial coefficients. When applied to optimal control, constraints can be imposed on the states by setting constraints on the polynomial coefficients.

The approximation with $K + 1$ interpolation points in each subinterval T_i can be understood using an analogy with a string. If you fix the left end of the string and hold the right end, the string will curve in such way to pass through both points. If you put a finger between both points, the string will shape so it passes between the three points. The same happens if you put another finger and so on. By holding the string and putting the fingers you are defining the points that the string has to pass. In a similar fashion, the coefficients of the Lagrange polynomial define the points that the polynomial has to pass at some predefined times, these points are known as collocation points.

Let us define the variable τ as the time variable normalized in the subinterval, being 0 at the beginning and 1 at the end of the

subinterval. At the i-th subinterval, the time variable is given by

$$t = t_{i-1} + h_i \tau \tag{2.109}$$

in which

$$t \in T_i = [t_{i-1}, t_i], \tag{2.110a}$$
$$t_i = t_{i-1} + h_{i-1} \qquad \tau \in [0, 1] \tag{2.110b}$$

where t_0 is given and $t_N = t_f$.

The basis for the Lagrangian polynomial is given by

$$\ell_j(\tau) = \prod_{k=0, \neq j}^{K} \frac{(\tau - \tau_k)}{(\tau_j - \tau_k)} \tag{2.111}$$

where τ_j and τ_k are the collocation points, with the property that $\tau_0 = 0$ and $\tau_j < \tau_{j+1}$ for $j = 0, \ldots, K-1$. The polynomial basis has the property that when $\tau = \tau_j$ the value of $\ell_j(\tau_j)$ is 1, and the value of $\ell_k(\tau_j)$ is 0 for $k = 0, \ldots, K-1$ but $k \neq j$. Therefore, if the polynomial ℓ_j is multiplied by some value z_{ij}, the product will assume the value z_{ij} when $\tau = \tau_j$, but all the other ℓ_k polynomials will be 0. This idea can be used to describe a curve that passes through every point (τ_j, z_{ij}) for $j = 1, \ldots, K$ by making a summation of all $\ell_j(\tau) z_{ij}$ terms,

$$z(t) = \sum_{j=0}^{K} \ell_j(\tau) z_{ij} \tag{2.112}$$

where $z(t)$ is the resulting curve. The variable z_{ij} is the state value at the collocation point j in the interval i. The values for τ_j can be obtained using the Gaussian quadrature. There are several rules to obtained the collocation points, for instance the Legendre-Gauss (LG) roots or the Legendre-Gauss-Radau (LGR) roots. The points obtained from these rules are presented in Table 2.1 for polynomials with degrees from 1 to 5. Being the LGR roots the mostly used for collocation, hence it was used in this work.

Now imagine that instead of fixing the positions the string has to pass in some given lengths, you are fixing the inclination of the string at these lengths. If you fix the inclination in some particular manner, the string will have the same shape that would be obtained by fixing the positions.

Table 2.1: Legendre-Gauss (LG) and Legendre-Gauss-Radau (LGR) roots as collocation points (Table 10.1 of [11]).

Degree (K)	LG roots	LGR roots
1	0.500000	1.000000
2	0.211325	0.333333
	0.788675	1.000000
3	0.112702	0.155051
	0.500000	0.644949
	0.887298	1.000000
4	0.069432	0.088588
	0.330009	0.409467
	0.669991	0.787659
	0.930568	1.000000
5	0.046910	0.057104
	0.230765	0.276843
	0.500000	0.583590
	0.769235	0.860240
	0.953090	1.000000

Recall that for the states, the information concerning where the states will be at time τ is not given. The information available are the initial condition (the point at the left of the string) and the time derivative of the states (a function that describes the inclination), which is given by (2.15), that is

$$\dot{x} = \frac{dx}{dt} = f(x,t). \tag{2.113}$$

To use the information available, the derivative of (2.112) is calculated

$$\frac{dz}{d\tau}(t) = \sum_{j=0}^{K} \frac{d\ell_j}{d\tau}(\tau) z_{ij}. \tag{2.114}$$

Since the time variable τ is t normalized, we can obtain the relation between dt and $d\tau$ by differentiating (2.109),

$$dt = h_i \, d\tau. \tag{2.115}$$

At the collocation points $t_{ik} = t_{i-1} + h_i \tau_k$, with $k = 1, \ldots, K$, it is desired that the model and the polynomial approximation have the same derivative, so it is stated

$$\frac{dz}{dt}(t_{ik}) = f(z(t_{ik}), t_{ik}), \quad k = 1, \ldots, K. \tag{2.116}$$

Using the relation (2.115), the derivative on the left-hand side of (2.116) can be changed to be with respect to τ. Then, (2.114) can be used to obtain

$$\sum_{j=0}^{K} z_{ij} \frac{d\ell_j}{d\tau}(\tau_k) = h_i f(z_{ik}, t_{ik}), \quad k = 1, \ldots, K. \tag{2.117}$$

For this system of nonlinear equations to admit a single solution, the number of free variables and equations have to be the same. Each subinterval i has $K+1$ variables z_{ij}. To define these variables, (2.117) has K equations and an additional equation is obtained from the initial condition. For the first subinterval $i = 1$, the initial condition is given by

$$z_{1,0} = z_0 \tag{2.118}$$

where $z_{1,0}$ is the first state value at the begin of the first subinterval. For the remaining subintervals, the initial condition is obtained from the continuity equation

$$z_{i+1,0} = \left[\sum_{j=0}^{K} \ell_j(\tau) z_{ij} \right] \Bigg|_{\tau=1}, \quad i = 1, \ldots, N-1 \tag{2.119}$$

where $z_{i+1,0}$ is the state at the begin of the subinterval $i+1$.

The final condition is given by

$$z_f = \left[\sum_{j=0}^{K} \ell_j(\tau) z_{Nj} \right] \Bigg|_{\tau=1} \tag{2.120}$$

where z_f is the value of the state at the end of the simulation period. In the case of a boundary value problem, an additional equation can force z_f to be equal to the given final condition.

It can be shown that the collocation method has truncation error ranging from $\mathcal{O}(h_i^{2K-2})$ to $\mathcal{O}(h_i^{2K})$, depending on which scheme of collocation points are chosen [11]. Therefore if a sufficient number of collocation points ($K+1$) and subintervals (N) are used, the approximation error can be neglected and the approximation $z(t)$ is equal to $x(t)$.

For the controls and algebraic variables, a similar approach can be used. Let $\widehat{\ell}_j(\tau)$ be Lagrangian polynomial basis defined by

$$\widehat{\ell}_j(\tau) = \prod_{k=1,\neq j}^{K} \frac{(\tau - \tau_k)}{(\tau_j - \tau_k)} \tag{2.121}$$

In the interval T_i the approximation of $y(t)$, represented by $\widetilde{y}_i(t)$, is given by

$$\widetilde{y}_i(\tau) = \sum_{j=1}^{K} \widehat{\ell}_j(\tau)\widetilde{y}_{ij} \tag{2.122}$$

where $\widetilde{y}_{ij}$ is the value of the algebraic variable in the subinterval T_i and at time $\tau = \tau_j$.

The Lagrangian polynomial has the property

$$\widetilde{y}_i(\tau_{ij}) = \widetilde{y}_{ij} \tag{2.123}$$

which allows to define $\widetilde{y}_{ij}$ by applying the algebraic equation directly into collocation points,

$$g(\widetilde{x}_{ij}, \widetilde{y}_{ij}, \widetilde{u}_{ij}, t_{ij}) = 0 \tag{2.124}$$

Using the same Lagrange polynomial basis (2.121), the control variable can be approximated

$$\widetilde{u}_i(\tau) = \sum_{j=1}^{K} \widehat{\ell}_j(\tau)\widetilde{u}_{ij} \tag{2.125}$$

Summarizing, this method creates a time mesh, each subinterval being approximated by a polynomial of K-th order represented with the Langrangrian base. The polynomials are differentiated and forced to be equal to the system derivative at each collocation point. The value of the state at the beginning of the sub-interval is equal to the value at the end of the prior subinterval. To help the reader to understand the collocation method, and illustrative example with an IVP is presented.

Example 13 (Collocation Method)**.** *For this example, we consider the dynamic system*

$$\frac{dz}{dt} = z^2 - 2z + 1, \quad z(0) = -3 \tag{2.126}$$

with $t \in [0,1]$. *This system has an analytical solution given by* $z(t) = (4t-3)/(4t+1)$. *However we are going to calculate the numerical approximation using the collocation method with Legendre-Gauss-Radau collocation points and a polynomial of order* $K = 3$. *A polynomial of third order requires four collocation points, those are* $\tau_0 = 0$, $\tau_1 = 0.155051$, $\tau_2 = 0.644949$, *and* $\tau_3 = 1$.

Using (2.117) *with* $N = 1$ *subinterval, and* $h = 1/N = 1$ *we have*

$$\sum_{j=0}^{3} z_{ij} \frac{d\ell_j(\tau_k)}{d\tau} = h(z_{ik}^2 - 2z_{ik} + 1), \quad k = 1, \ldots, 3, \; i = 1 \tag{2.127}$$

Developing the Lagrangian base described in (2.111) *and taking its derivative we find*

$$\frac{d\ell_0(\tau_k)}{d\tau} = -30\tau_k^2 + 36\tau_k - 9 \tag{2.128a}$$

$$\frac{d\ell_1(\tau_k)}{d\tau} = 46.7423\tau_k^2 - 51.2392\tau_k - 10.0488 \tag{2.128b}$$

$$\frac{d\ell_2(\tau_k)}{d\tau} = -23.7423\tau_k^2 + 20.5925\tau_k - 1.38214 \tag{2.128c}$$

$$\frac{d\ell_3(\tau_k)}{d\tau} = 10\tau_k^2 - \frac{16}{3}\tau_k + \frac{1}{3} \tag{2.128d}$$

Substituting (2.128) *in* (2.127) *we find:*

$$\begin{aligned} &z_{10}(-30\tau_k^2 + 36\tau_k - 9) + z_{11}(46.7423\tau_k^2 - 51.2392\tau_k + 10.0488) \\ &+ z_{12}(-23.7423\tau_k^2 + 20.5925\tau_k - 1.38214) + z_{13}\left(10\tau_k^2 - \frac{16}{3}\tau_k + \frac{1}{3}\right) \\ &= (z_{1k}^2 - 2z_{1k} + 1), \quad k = 1, \ldots, 3 \end{aligned} \tag{2.129}$$

Solving this equations system we have $z_{11} = -1.65701$, $z_{12} = 0.032053$, $z_{13} = 0.207272$ *with* $z_{10} = -3$. *Figure 2.7 compares the solution using the collocation method and the analytical solution. To increase the accuracy of the approximation we could increase the number of time discretizations* N *or the order* K *of the polynomial approximation.*

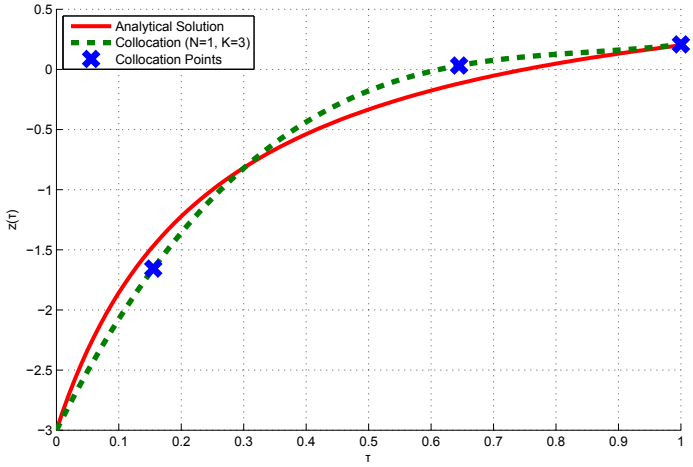

Figure 2.7: Example of analytical and approximation solutions.

3 OPTIMAL CONTROL

This chapter gives the background on the optimal control theory.

It starts with calculus of variations, providing a solid ground on the space that optimal control theory is developed. The first variation is a tool that is introduced and used to demonstrate the Euler-Lagrange equation.

In the following, a basic optimal control problem (OCP) is stated and a necessary condition for optimality is given. Afterwards, the Pontryagin's minimum principle is demonstrated for a broader case. At last, the Hamilton-Bellman-Jacobi equation is presented.

OCPs are extended for differential algebraic systems and the necessary conditions for such systems are given.

The chapter ends by presenting tools for solving OCP in practice. The tools can be split in two main classes, the direct methods and the indirect methods. The indirect methods use the optimality condition developed in this chapter to find a set of controls that fulfill the conditions. On the other hand, the indirect methods use the sensitivity calculation and other methods to find a set of controls that minimize an objective functional.

3.1 CALCULUS OF VARIATIONS [1]

The subject of the calculus of variations are the functionals. Functionals have functions as domain and real numbers as image, similarly, functions have real vectors in the domain and real numbers in the image.

For instance, consider a function $f : \mathbb{R}^2 \to \mathbb{R}$ where

$$f(x_1, x_2) = x_1 + x_2. \tag{3.1}$$

We refer to f as a function, but $f(1, 2)$ is the real number 3. Therefore given a functional $I : X \to \mathbb{R}$, if a function $x : [t_0, t_f] \to X$ is applied to I, denoted $z = I(x)$, we obtain a real number $z \in \mathbb{R}$, where $[t_0, t_f]$ is the interval that x is defined and X is a function space.

The concept of functionals may be better understood with illustrative examples:

[1] This section was written based on [?], and the works [13, 14] were used as reference.

1. **Area below a curve x:** The area below a curve characterizes the classical notion of integral. We know that for a function x, the functional for the area $I_a(x)$ is given by

$$I_a(x) = \int_{t_0}^{t_f} x(t)\, dt. \tag{3.2}$$

2. **Length of a curve x:** Having as input the function x in the set of rectifiable curves (curves that can be approximated by an infinite number of tiny straight lines), the length of x can be represented by the functional

$$I_\ell(x) = \int_{t_0}^{t_f} \sqrt{1 + \dot{x}^2}\, dt, \tag{3.3}$$

whose derivation follows (3.10b), described in Example 14 later in this Chapter.

3. **Quadratic error:** The quadratic error is a classical measure in several areas of mathematics, including control. For a function x, the functional that calculates the quadratic error with respect to a reference x_{ref} during a period $t \in [t_0, t_f]$ is given by

$$I_e(x) = \int_{t_0}^{t_f} (x_{ref} - x(t))^2\, dt. \tag{3.4}$$

Within the scope of this work, a functional is given by an integral of a function, which is clarified in the following definition.

Definition 7 (Functional)**.** *A functional $I : X \to \mathbb{R}$ is given by the equation*

$$I(x) = \int_{t_0}^{t_f} F(x, \dot{x}, t)\, dt \tag{3.5}$$

where X is a function space, with $x : [t_0, t_f] \to \mathbb{R}^{N_x}$, and the function $F : \mathbb{R}^{N_x} \times [t_0, t_f] \to \mathbb{R}$ which is referred as the Lagrangian function.

Remark. *Although there are other types of functionals, e.g. the functional of the derivative at the point 0 ($I(x) = \dot{x}(0)$), the functionals of interest are those defined by an integral.*

In particular, we are usually interested in the critical curves $x^* \in X$ that are solutions for the problem of minimizing (or maximizing) a functional I.

Example 14 (The Brachistochrone Problem)**.** *The brachistochrone problem was defined by the Swiss mathematician Johan Bernoulli in 1696. The problem is named from Greek words* brakhistos khrónos*, that means "shortest time". The proposed problem consists in finding the shape of a ramp that yields the least time for a ball rolling from the point $(0, 0)$ to a point (x_f, y_f), being y_f in a lower level. The problem is depicted in Figure 3.1.*

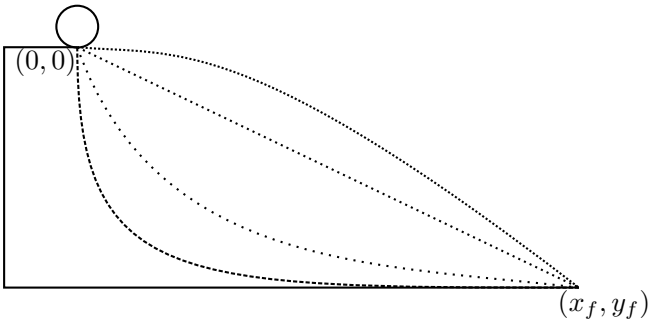

Figure 3.1: Illustration of different shapes for a ramp for the Brachistochrone Problem.

Contrary to common sense, the solution of this problem is not a straight line between the initial and final points. Intuitively, we can imagine that if the curve has large negative slope at the beginning it will quickly increase the ball velocity and therefore reach faster the final point. On the other hand, a large negative slope increases the length of the curve that the ball has to travel. Therefore, the optimal shape seems to lie between a straight line and a very steep curve.

To model this problem, we use the energy conservation laws,

$$E_k - E_p = \frac{1}{2}mv^2 - mgy = 0 \tag{3.6}$$

where E_k is the kinetic energy and E_p is the potential energy, being the initial height as the point of zero potential energy (this point does not change the results, however choosing it as the initial height is convenient). The variables are the height y and the ball velocity v, and the parameters are the ball mass m and the gravitational acceleration

g, which lead to

$$v = \frac{ds}{dt} = \sqrt{2gy} \tag{3.7}$$

where s is the arc length of the curve. Rearranging the equation,

$$\frac{1}{\sqrt{2gy}}\frac{ds}{dt} = 1 \tag{3.8}$$

which can be integrated in both sides with respect to t

$$\int_{t_0}^{t_f} \frac{1}{\sqrt{2gy}}\frac{ds}{dt}\,dt = \int_{t_0}^{t_f} dt = T \tag{3.9}$$

where T is the time that the ball takes to reach the final point. Using Pythagoras' Theorem we can write a small change in the arc length (ds) as a function of small change in the x axis (dx) and y axis (dy),

$$ds^2 = dx^2 + dy^2 \implies \left[\frac{ds}{dx}\right]^2 = 1 + \left[\frac{dy}{dx}\right]^2 \tag{3.10a}$$

$$ds = \left[\sqrt{\left(\frac{dy}{dx}\right)^2 + 1}\right] dx \tag{3.10b}$$

Changing the limits of integration and replacing (3.10b) *in* (3.9)*, we obtain*

$$T = \int_{x(t_0)=0}^{x(t_f)=x_f} \sqrt{\frac{1 + \left(\frac{dy}{dx}\right)^2}{2gy}}dx \tag{3.11}$$

Notice that the problem is stated with y as the independent variable and x as the dependent variable. In order to fit it to the standard form, as given in Definition 7, a change of system of variables can be performed. Let t be the independent variable and x be the dependent variable, the resulting functional is

$$T(x) = \int_{t_0}^{t_f} \sqrt{\frac{1 + \dot{x}^2}{2gx}}dt \tag{3.12}$$

where $t_0 = 0$ and $t_f = x_f$.

This is a practical application of functionals, however we are not yet able to solve the minimization problem. Before the problem can be solved, the space in which the minimization of functionals take place and its properties must be specified.

3.1.1 Function Space

When working with functions of a variable x, it is enough to say that they belong to a single space, for instance, by saying that the vector x is in the Euclidean space $\mathbb{R}^n$. However, for function space this type of assignment is hardly made. The characteristics of the problem will define the space. For example, the functional

$$I = \int_{t_0}^{t_f} F(x, \dot{x}, t)\, dt \tag{3.13}$$

requires that the function x belongs to the set of functions with continuous first derivative.

The concept of continuity of functionals is as important as the continuity for functions in classical analysis. To develop this concept for functionals, first a notion of distance must be developed for functions. The distance can be conveniently measured by a norm, but we are still to develop the concept of norm of a function.

Before discussing norms and linear normed spaces, we shall recall the concept of a general linear space.

A linear space over $\mathbb{R}$ is defined by a set X together with the operations of addition, $+ : X \times X \to X$, and scalar multiplication, $\cdot : X \times X \to X$, that satisfy the following properties:

1. Associativity: $x_1 + (x_2 + x_3) = (x_1 + x_2) + x_3$.
2. Commutative of addition: $x_1 + x_2 = x_2 + x_1$.
3. Identity element of addition: There exists an element $0 \in X$, such that $x_1 + 0 = x_1$.
4. Inverse element: For every $x_1 \in X$ there is a $-x_1 \in X$, such that $(x_1) + (-x_1) = 0$.
5. Compatibility of multiplication: $a(bx_1) = (ab)x_1$.
6. Identity element of scalar multiplication: There exists and element 1, such that $1x_1 = x_1$.
7. Distributivity of scalar multiplication with respect to vector addition: $a(x_1 + x_2) = ax_1 + ax_2$.
8. Distributivity of scalar multiplication with respect to field addition: $(a + b)x_1 = ax_1 + bx_1$.

In addition, a *linear function* $L : X \to \mathbb{R}$ is a map that satisfies:

1. $L(x_1 + x_2) = L(x_1) + L(x_2)$ for all $x_1, x_2 \in X$.
2. $L(ax_1) = aL(x_1)$ for all $a \in \mathbb{R}$ and for all $x_1 \in X$.

A linear space is considered a normed space, if there is a function $\|\cdot\| : X \to [0, \infty)$, named the norm, that has the following properties:

1. Zero norm: $\|x\| = 0$ if and only if $x = 0$.
2. Absolute scalability: $\|ax\| = |a| \, \|x\|$ for all $a \in \mathbb{R}$ and for all $x \in X$.
3. Triangle inequality: $\|x_1 + x_2\| \leq \|x_1\| + \|x_2\|$.

In a normed space it makes sense to establish distances between elements. We define a distance function $d : X \times X \to \mathbb{R}$, where the distance between an element x_1 and x_2 is given by $d(x_1, x_2) = \|x_1 - x_2\|$. A distance function has the following properties:

1. $d(x, y) \geq 0$ for all x and y in X.
2. $d(x, y) = 0$ if and only if $x = y$.
3. $d(x, y) = d(y, x)$ for all x and y in X.
4. $d(x, z) \leq d(x, y) + d(y, z)$ for all x, y, and z in X.

A linear space X equipped with a distance function d is called a metric space.

The elements of a normed linear space can be of any type, *e.g.* numbers, matrices, functions, *etc*. However, working with functionals the following normed spaces are more important:

1. $C[t_0, t_f]$ is the normed space formed by all the continuous functions x in the interval $[t_0, t_f]$. The addition operation is defined by a pointwise addition, meaning $x_1(t) + x_2(t) = (x_1 + x_2)(t)$ for all $t \in [t_0, t_f]$, and the scalar multiplication is defined by $ax(t) = (ax)(t)$ for all $t \in [t_0, t_f]$. The norm function is defined by

$$\|x\| = \max_{t \in [t_0, t_f]} \|x(t)\|_\infty \tag{3.14}$$

Therefore, saying that the distance between $x_1(t)$ and $x_2(t)$ does not exceed ε means that if we plot these curves we would see that $x_2(t)$ lays inside a band with the borders $(x_1 + \varepsilon)(t)$ and $(x_1 - \varepsilon)(t)$, as shown in Figure 3.2.

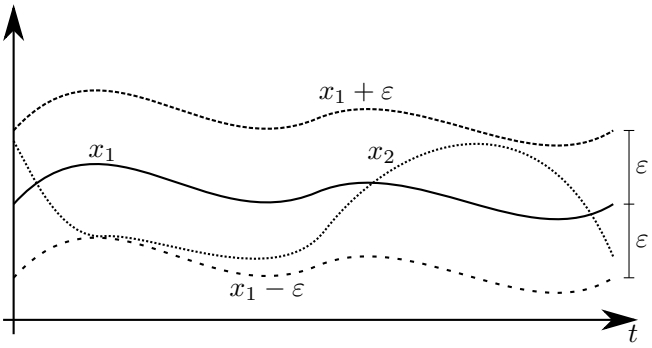

Figure 3.2: The distance between functions x_1 and x_2 is ε .

2. $C^1[t_0, t_f]$ is the normed space that contains all the functions $x(\cdot)$ with continuous first derivative in the interval $t \in [t_0, t_f]$. The addition and multiplication operations are the same of the space $C[t_0, t_f]$, however the norm is defined by

$$\|x\| = \max_{t\in[t_0,t_f]} \|x(t)\|_\infty + \max_{t\in[t_0,t_f]} \|\dot{x}(t)\|_\infty \tag{3.15}$$

Thus, if $x_1(t)$ and $x_2(t)$ have a distance that does not exceed ε this means that

$$|x_1(t) - x_2(t)| < \varepsilon, \quad \forall t \in [t_0, t_f], \text{ and} \tag{3.16a}$$
$$|\dot{x}_1(t) - \dot{x}_2(t)| < \varepsilon, \quad \forall t \in [t_0, t_f] \tag{3.16b}$$

The function space presented so far contains the continuous and continuous differentiable functions with domain $[t_0, t_f]$ and codomain $\mathbb{R}$. However, when describing practical minimization problems we want functions from the interval $[t_0, t_f]$ to a particular space $\mathbb{R}^d$. Therefore we can define the space of functions $(\mathcal{C}[t_0, t_f])^d$, the space of continuous functions from $[t_0, t_f]$ to $\mathbb{R}^d$. For example, in the brachistochrone problem we are interested in continuously differentiable functions from $[t_0, t_f]$ to $\mathbb{R}$, or equivalently, interested in functions in the space $x \in (\mathcal{C}[t_0, t_f])^1$. The same definitions can be applied to define the space of $(\mathcal{C}^1[t_0, t_f])^d$. In practice, these definitions are not used and the target space of the functions are left implicit by the problem description. In the brachistochrone problem, the only functions that would fit the formulation are those from $[t_0, t_f]$ to $\mathbb{R}$.

Beside the function continuity, which is given by the chosen function space, the functional continuity must be considered. A functional $I : X \to \mathbb{R}$ is said to be continuous at $\bar{x}$ if for every $\varepsilon > 0$ there is a $\delta > 0$ such that

$$|I(x) - I(\bar{x})| < \varepsilon \text{ for all } x \in X \text{ such that } \|x - \bar{x}\| < \delta \tag{3.17}$$

where the norm used in the second inequality is the norm of the function space X. A functional $I : X \to \mathbb{R}$ is said to be continuous if it is continuous at all $x \in X$.

We define a local minimum of the functional I as the function x^* such that

$$I(x^*) \leq I(x), \text{ for all } x \text{ that satisfies } \|x^* - x\| < \varepsilon \tag{3.18}$$

for some $\varepsilon > 0$, where the norm is a norm of a function. Conversely, a local maximum of a functional I is a functional x^* such that

$$I(x^*) \geq I(x), \text{ for all } x \text{ that satisfies } \|x^* - x\| < \varepsilon \tag{3.19}$$

for some $\varepsilon > 0$. A function is said to be a local extremum, or merely an extremum, of the functional I if it is a local minimum or a local maximum. Moreover, the function x^* is global minimum for I if

$$I(x^*) \leq I(x), \text{ for all } x \in X \tag{3.20}$$

The concept of global maximum and global extremum are analogous to their local counterparts.

3.1.2 Derivative of Functionals

In the previous subsection, we defined the grounds for the calculus of variations by introducing functionals, function spaces, and continuity in this context. However, most of the problems have the candidate functions that are not in a linear space. For instance, the brachistochrone problem has a set of candidate functions, namely X_b, as the continuously differentiable functions that have the initial value at $(0, 0)$ and the final value at (x_f, y_f).

It can be verified that X_b is not a linear space, by defining a function x_3 as

$$x_3 = x_1 + x_2, \text{ where } x_1, x_2 \in X_b, \tag{3.21}$$

where $x_3 \notin X_b$, since for $t = t_f$

$$x_3(t_f) = x_1(t_f) + x_2(t_f) = 2y_f \tag{3.22}$$

Therefore it can be concluded that X_b is not a linear space.

The same kind of problem happens with problems with vectors and functions, when they do not belong to a linear subspace. To cope with this situation we linearize the system around a region of interest. The same argument can be used for functionals. To this end, we need to recall the definition of the derivative of functions and then define the derivative of functionals.

Given a function $f : \mathbb{R} \to \mathbb{R}$, the derivative of f at point $\bar{x}$ is the approximation of f around $\bar{z}$ by an affine linear map $f'(\bar{z})$,

$$f(\bar{z} + h_z) = f(\bar{z}) + f'(\bar{z})h_z + \epsilon(h)|h_z| \tag{3.23}$$

where $h_z \in \mathbb{R}$ is a small scalar and the approximation error $\epsilon(h_z) \to 0$ as $h_z \to 0$.

In the same manner, for a functional I, the derivative at the *function* $\bar{x}$ is given by the linear map $I'(\bar{x})$, such that

$$I(\bar{x} + h) = I(\bar{x}) + [I'(\bar{x})]\,(h) + \epsilon(h)\|h\| \tag{3.24}$$

where $h : [t_0, t_f] \to \mathbb{R}$ and the error functional $\epsilon(h) \to 0$ as $\|h\| \to 0$.

A linear map $L : \mathbb{R}^n \to \mathbb{R}$ is always continuous, yet this assumption does not hold if $\mathbb{R}^n$ is replaced by an infinite dimensional space X. Therefore, when specifying linear maps in an infinite dimensional space X, one must also specify its continuity. A linear map $L : X \to \mathbb{R}$ is a continuous linear functional if it is linear and it is continuous.

Given the notion of derivatives for a functional, the concept can be formalized with the following definition.

Definition 8 (Fréchet Derivative)**.** *Let X be a normed linear space and $I : X \to \mathbb{R}$ be a functional. Then I is Fréchet differentiable at $\bar{x} \in X$ if there is a continuous linear map $A : X \to \mathbb{R}$ and a map $\epsilon : X \to \mathbb{R}$ such that, for all $h \in X$,*

$$I(\bar{x} + h) = I(\bar{x}) + A(h) + \epsilon(h)\|h\|, \tag{3.25}$$

and the error functional $\epsilon(h) \to 0$ as $\|h\| \to 0$. We write $A = I'(\bar{x})$, where $I'(\bar{x})$ is called the Fréchet derivative of I at $\bar{x}$. Since $I'(\bar{x})$ is a continuous linear operator we can write

$$A(h) = [I'(\bar{x})](h) = I'(\bar{x})h \tag{3.26}$$

We choose the last form to improve the readability in large equations.

In addition, if I is Fréchet differentiable for every function $x \in X$, then the functional I is Fréchet differentiable.

The Fréchet derivative has some desirable properties. It can be shown that for a functional I at $\bar{x}$ the Fréchet derivative is unique (see Theorem 16 in Appendix A), and if x^* is an extremum of the functional I, then its Fréchet derivative is $I'(x^*) = 0$, in the same manner that if a function f has a local extremum at the point z^*, then $\frac{df}{dz}(z^*) = 0$.

Theorem 1. *Let X be a normed linear space and $I : X \to \mathbb{R}$ be a Fréchet differentiable functional in $x^* \in X$. If I has a local extremum at x^*, then $I'(x^*) = 0$.*

Proof. The proof of this theorem is in Appendix A. □

The Fréchet derivative has good properties however its definition using limits is difficult to work. We need a way to obtain the derivative that relies on already known and well developed tools. For this matter, we can use a different notion of differentiability. The Gateaux derivative of a functional is similar to directional derivative for functions, however instead of taking the derivative in the direction of a vector we take the derivative in the direction of a function $h \in X$.

Definition 9 (Gateaux Derivative or First Variation)**.** *The functional $I : X \to \mathbb{R}$ at the point $\bar{x} \in X$ has the Gateaux derivative (also known as first variation) in the direction $h \in X$ defined by*

$$\delta I(\bar{x}, h) \equiv \lim_{\xi \to 0} \frac{I(\bar{x} + \xi h) - I(\bar{x})}{\xi} = \frac{dI}{d\xi}(\bar{x} + \xi h)\bigg|_{\xi=0} \tag{3.27}$$

if the limit exists. If the Gateaux derivative exists for all directions $h \in X$, then I is Gateaux differentiable.

To clarify the Gateaux derivative, let us go through an example.

Example 15. *For this example we want to obtain the Gateaux derivative of a functional that calculates the length of a curve that connects the point $x(t_0) = 0$ to $x(t_f) = 1$, with $t_0 = 0$ and $t_f = 1$.*

The functional that gives the length of a curve x is given by

$$I_\ell(x) = \int_{t_0}^{t_f} \sqrt{1 + \dot{x}^2}\, dt \tag{3.28}$$

Consider that we want to evaluate the derivative for the function $\bar{x}$ *given by* $\bar{x}(t) = t$*, which clearly has the smallest length. The value of the function for* $I_\ell(\bar{x})$ *is*

$$I_\ell(\bar{x}) = \int_{t_0}^{t_f} \sqrt{1 + (1)^2}\, dt = \left[\sqrt{2}t\right]\Big|_{t=t_0}^{t=t_f} = \sqrt{2} \tag{3.29}$$

Let h *be a function that satisfies* $h(t_0) = 0$ *and* $h(t_f) = 0$ [2]*. The Gateaux derivative of the functional* I_ℓ *at* x *in the direction* h *is*

$$\delta I_\ell(x) = \frac{d}{d\xi} \int_{t_0}^{t_f} \sqrt{1 + \left[\dot{x} + \xi\dot{h}\right]^2}\, dt \Bigg|_{\xi=0}. \tag{3.30}$$

Since the function inside the integral is continuous the derivative can be moved inside the integral

$$\delta I_\ell(x) = \int_{t_0}^{t_f} \frac{d}{d\xi} \sqrt{1 + \left[\dot{x} + \xi\dot{h}\right]^2}\Bigg|_{\xi=0} dt. \tag{3.31}$$

Applying the derivation results in

$$\delta I_\ell(x) = \int_{t_0}^{t_f} \frac{(\dot{x} + \xi\dot{h})\dot{h}}{\sqrt{1 + \left[\dot{x} + \xi\dot{h}\right]^2}}\Bigg|_{\xi=0} dt \tag{3.32}$$

and applying for $\xi = 0$*,*

$$\delta I_\ell(\bar{x}) = \int_{t_0}^{t_f} \frac{\dot{x}}{\sqrt{1 + \dot{x}^2}} \dot{h}\, dt \tag{3.33}$$

which is the result for the derivative of I_ℓ *at* $\bar{x}$ *in some direction* h*.*

For $\bar{x} = t$ *the value of* δI_ℓ *is*

$$\delta I_\ell(\bar{x}) = \int_{t_0}^{t_f} \frac{1}{\sqrt{1 + 1^2}} \dot{h}\, dt, \tag{3.34}$$

for any h*.*

[2] Otherwise $x(t) + \xi h(t)$ does not pass at 0 when $t = t_0$ and 1 at $t = t_f$.

Applying the integral

$$\delta I_\ell(\bar{x}) = \left[\frac{1}{\sqrt{2}}h(t)\right]\Bigg|_{t=t_0}^{t=t_f} = \frac{1}{\sqrt{2}}\left[h(t_f) - h(t_0)\right] = 0, \tag{3.35}$$

since we assumed that $h(t_0) = h(t_f) = 0$.

We conclude that for any variation h *in the line* $\bar{x}(t)$*, the value for* $\delta I_\ell(\bar{x}) = 0$*. This makes sense, given that a straight line between* $(0, 0)$ *and* $(1, 1)$ *has the smallest length.*

The notion of differentiability defined by Gateaux is weaker than the one defined by Fréchet, since $\delta I(\bar{x}, h)$ might be neither linear nor continuous with respect to h. However for some cases, the Fréchet and the Gateaux derivative can be related:

Theorem 2. *Let functional* $I : X \to \mathbb{R}$ *be Fréchet differentiable. Then* I *is also Gateaux differentiable. Furthermore, the Gateaux and Fréchet derivative agree,*

$$I'(\bar{x})h = \delta I(\bar{x}, h) \tag{3.36}$$

for all $h \in X$.

Proof. The proof of this theorem is in Appendix A. □

Note that the converse is not true, Gateaux differentiability does not imply Fréchet differentiability.

One of the important outcomes of Theorem 2 is that we are able to compute the Fréchet derivative applied to h by differentiating with respect to ξ at $\xi = 0$, as defined by Gateaux derivative. Meaning that,

$$I'(\bar{x})h = \frac{d}{d\xi}I(\bar{x} + \xi h)\bigg|_{\xi=0}. \tag{3.37}$$

Some rules that are valid for the classical notion of derivatives carry to Fréchet derivative [15], for instance the chain rule and the product rule.

The definition of the Fréchet also allows to expand functionals using Taylor series [16]. Given a functional $I : X \to \mathbb{R}$, it can be expanded in the following form

$$I(x + h) = I(x) + I'(x)h + \text{higher order terms}. \tag{3.38}$$

In the case that the higher order terms are truncated the approximation error $\epsilon(h) \to 0$ as $\|h\| \to 0$.

This notion leads to an important tool for a calculus of variations, the first variation. The first variation does not differ from the definition of the Fréchet derivative obtained using the Gauteaux derivative. However it can be understood with a different intuition.

Let I be a Fréchet differentiable functional, $\bar{x}$ be a function in the space X, and δx be a function close enough to 0 that can assume any shape given that it satisfies the regularity condition of the space X. The variation δI on the functional I is caused by "perturbing" the function $\bar{x}$ with δx. The variation δI can be obtained using the Gateaux derivative,

$$\delta I(x, \delta x) = \frac{d}{\delta \xi} I(\bar{x} + \xi \delta x)\bigg|_{\xi=0} \tag{3.39}$$

Since the target functional has the form (3.5),

$$I(x) = \int_{t_0}^{t_f} F(x, \dot{x}, t)\, dt, \tag{3.40}$$

we can develop a generic form for the first variation of this functional.

Theorem 3 (First Variation)**.** *Given a Fréchet differentiable functional $I : X \to \mathbb{R}$ defined by*

$$I(x) = \int_{t_0}^{t_f} F(x, \dot{x}, t)\, dt, \tag{3.41}$$

where F is a continuously differentiable function with respect to x, $\dot{x}$, and t.

The first variation of I, namely δI, with a perturbation δx is given by

$$\delta I(x, \delta x) = \int_{t_0}^{t_f} \left[\frac{\partial F}{\partial x}(x, \dot{x}, t)\delta x + \frac{\partial F}{\partial \dot{x}}(x, \dot{x}, t)\delta \dot{x} \right] dt. \tag{3.42}$$

Proof. If the functional I is Fréchet differentiable, then the definition of the Gateaux and Fréchet differentiations are interchangeable. Therefore

$$\delta I(x, \delta x) = I'(x)\delta x = \frac{d}{d\xi} I(x + \xi \delta x)\bigg|_{\xi=0}. \tag{3.43}$$

Replacing $I(x + \xi\delta x)$ with its definition results in

$$\delta I(x, \delta x) = \frac{d}{d\xi} \int_{t_0}^{t_f} F(x + \xi\delta x, \dot{x} + \xi\delta\dot{x}, t)\, dt \Big|_{\xi=0}. \tag{3.44}$$

Let us define an auxiliary variable $y = x + \xi\delta x$, with the first derivative with respect to t given by $\dot{y} = \dot{x} + \xi\delta\dot{x}$. By replacing y in the definition of $I(x + \xi\delta x)$, we obtain

$$\delta I(x, \delta x) = \frac{d}{d\xi} \int_{t_0}^{t_f} F(y, \dot{y}, t)\, dt, \Big|_{\xi=0}. \tag{3.45}$$

Since F is continuously differentiable with respect to its arguments, we can move the differentiation to inside the integral,

$$\delta I(x, \delta x) = \int_{t_0}^{t_f} \frac{dF}{d\xi}(y, \dot{y}, t) \Big|_{\xi=0} dt, \tag{3.46}$$

and applying the total derivative rule we obtain

$$\delta I(x, \delta x) = \int_{t_0}^{t_f} \left[\frac{\partial F}{\partial y}(y, \dot{y}, t)\frac{dy}{d\xi} + \frac{\partial F}{\partial \dot{y}}(y, \dot{y}, t)\frac{d\dot{y}}{d\xi} \right] \Big|_{\xi=0} dt. \tag{3.47}$$

Replacing y with $x + \xi\delta x$, $\dot{y}$ with $\dot{x} + \xi\delta\dot{x}$, and applying $\xi = 0$, we obtain

$$\delta I(x, \delta x) = \int_{t_0}^{t_f} \left[\frac{\partial F}{\partial x}(x, \dot{x}, t)\delta x + \frac{\partial F}{\partial \dot{x}}(x, \dot{x}, t)\delta\dot{x} \right] dt. \tag{3.48}$$

□

Sometimes the first variation is written as $\delta I(x)$, without exposing the dependence on the perturbation variable δx, for the sake of simplicity and readability. Alternatively, the dependence on x can also be suppressed, representing only as δI.

The first perturbation can be applied to multivariate functionals. For instance given the functional

$$I(x, y) = \int_{t_0}^{t_f} F(x, y, \dot{x}, \dot{y}, t)\, dt, \tag{3.49}$$

the first variational $\delta I(x, y, \delta x, \delta y)$ is given by the sum of the partial derivatives times the perturbation,

$$\delta I(x,y) = \int_{t_0}^{t_f} \left[\frac{\partial}{\partial x} F(x,y,t)\delta x + \frac{\partial}{\partial \dot{x}} F(x,y,t)\delta \dot{x} + \frac{\partial}{\partial y} F(x,y,t)\delta y + \frac{\partial}{\partial \dot{y}} F(x,y,t)\delta \dot{y} \right] dt. \quad (3.50)$$

3.1.3 Euler-Lagrange equation

In Theorem 1 we have seen that if a function x^* is an extremum of a functional I, then the Fréchet derivative is zero at x^*. The Euler-Lagrange equation gives a more tangible *necessary* condition for optimality of functionals with integrals.

Theorem 4 (Euler-Lagrange Equation)**.** *Let I be a Fréchet differentiable functional of the form*

$$I(x) = \int_{t_0}^{t_f} F\left(x(t), \dot{x}(t), t\right)\, dt, \quad (3.51)$$

where the function F is differentiable with respect to x, $\dot{x}$, and t. The function $x(t) \in \mathcal{C}^1[t_0, t_f]$ passes through the points $x(t_0) = x_0$ and $x(t_f) = x_f$. If I has an extremum at x^*, then x^* satisfies the Euler-Lagrange equation:

$$\frac{\partial F}{\partial x}\left(x^*(t), \dot{x}^*(t), t\right) - \frac{d}{dt}\frac{\partial F}{\partial \dot{x}}\left(x^*(t), \dot{x}^*(t), t\right) = 0 \quad (3.52)$$

for all t in $[t_0, t_f]$. Alternatively, in a more compact format:

$$\frac{\partial F}{\partial x} - \frac{d}{dt}\frac{\partial F}{\partial \dot{x}} = 0. \quad (3.53)$$

Proof. The proof follows by showing that the Euler-Lagrange is an implication of Theorem 1. However, note that the set of curves in $\mathcal{C}^1[t_0, t_f]$ that meet the conditions $x(t_0) = x_0$ and $x(t_f) = x_f$ does not form a linear space, so Theorem 1 does not apply directly. Therefore, let us define the linear space H given by

$$H = \left\{ \delta x \in \mathcal{C}^1[t_0, t_f] \,\middle|\, \delta x(t_0) = \delta x(t_f) = 0 \right\}, \quad (3.54)$$

which has the property that for all $\delta x \in H$, $x^*(t_0) + \delta x(t_0) = x_0$ and $x^*(t_f) + \delta x(t_f) = x_f$. Which means that δx is allowed to perturbe at

all t except at the end points of the interval of integration, otherwise the sum of $x^* + \delta x$ will not pass through (t_0, x_0) and (t_f, x_f).

Let $J : H \to \mathbb{R}$ be ka functional defined by

$$J(\delta x) = I(x^* + \delta x) \tag{3.55}$$

for all $\delta x \in H$. Notice that J has an extremum at $J(0) = I(x^*)$.

By Theorem 1, if 0 is an extremum of J then $J'(0) = 0$. So if the first variation is applied with a perturbation $\delta x \in H$,

$$\delta J(\delta x) = \int_{t_0}^{t_f} \left[\frac{\partial F}{\partial x}(x^*, \dot{x}^*, t)\delta x + \frac{\partial F}{\partial \dot{x}}(x^*, \dot{x}^*, t)\delta \dot{x} \right] dt = 0, \tag{3.56}$$

integrating by parts the second term, results in

$$\left[\frac{\partial F}{\partial \dot{x}}(x^*, \dot{x}^*, t)\delta x(t) \right] \Bigg|_{t_0}^{t_f} + \int_{t_0}^{t_f} \left[\frac{\partial F}{\partial x}(x^*, \dot{x}^*, t) - \frac{d}{dt}\frac{\partial F}{\partial \dot{x}}(x^*, \dot{x}^*, t) \right] \delta x \, dt = 0. \tag{3.57}$$

Notice that $\delta x(t_0) = \delta x(t_f) = 0$, therefore the term outside the integral vanishes. The remaining terms are

$$\int_{t_0}^{t_f} \left[\frac{\partial}{\partial x} F(x^*, \dot{x}^*, t) - \frac{d}{dt}\frac{\partial}{\partial \dot{x}} F(x^*, \dot{x}^*, t) \right] \delta x \, dt = 0, \tag{3.58}$$

as the function δx can be any function in H, we use the Lemma 1, which is stated in the following, to conclude that

$$\frac{\partial}{\partial x} F(x^*, \dot{x}^*, t) - \frac{d}{dt}\frac{\partial}{\partial \dot{x}} F(x^*, \dot{x}^*, t) = 0. \tag{3.59}$$

□

In the following we give the fundamental lemma for the calculus of variations, which supports the condition established by the Euler-Lagrange equation.

Lemma 1 (Fundamental Lemma)**.** *Given a particular function $f \in \mathcal{C}[t_0, t_f]$, if*

$$\int_{t_0}^{t_f} f(t)h(t) \, dt = 0 \tag{3.60}$$

for every $h \in \mathcal{C}^1[t_0, t_f]$ *such that*

$$h(t_0) = h(t_f) = 0 \tag{3.61}$$

we conclude that

$$f(t) = 0 \qquad \forall t \in [t_0, t_f]. \tag{3.62}$$

Proof. We will develop a proof by contradiction. Assume that there exists an interval $[t_1, t_2] \subset [t_0, t_f]$ in which $f(t) > 0$ (an equivalent demonstration can be made for the assumption $f(t) < 0$).

As (3.60) is valid for every h, we choose h as

$$h(t) = \begin{cases} 0 & \text{for } t \in [t_0, t_1) \\ (t - t_1)^2(t - t_2)^2 & \text{for } t \in [t_1, t_2] \\ 0 & \text{for } t \in (t_2, t_f] \end{cases} \tag{3.63}$$

Notice that the chosen h is continuously differentiable, even at the points t_1 and t_2.

The integral (3.60) becomes

$$\int_{t_0}^{t_f} f(t)h(t)\,dt = \int_{t_0}^{t_1} f(t)h(t)\,dt + \int_{t_1}^{t_2} f(t)h(t)\,dt + \int_{t_2}^{t_f} f(t)h(t)\,dt \tag{3.64a}$$

$$= \int_{t_1}^{t_2} f(t)h(t)\,dt \tag{3.64b}$$

$$= \int_{t_1}^{t_2} f(t)(t - t_1)^2(t - t_2)^2\,dt > 0 \tag{3.64c}$$

The inequality (3.64c) results from the assumption that $f(t) > 0$, and, at the same time, h only takes nonnegative values in this interval. Since (3.64c) contradicts (3.60), we conclude that

$$f(t) = 0 \tag{3.65}$$

must hold for all $t \in [t_0, t_f]$. □

Remark. *Lemma 1 can be modified to hold for a more restrictive condition. For instance the case where* $h \in \mathcal{C}^p[t_0, t_f]$*. The proof follows in the same way as the proof given in the lemma, however one must choose* $h = (t - t_1)^{p+1}(t - t_2)^{p+1}$ *for the interval* $[t_1, t_2]$*, which guarantees that the chosen* h *is* p*-differentiable.*

Although the Euler-Lagrange equations define a necessary condition, for most of the cases there is some additional knowledge about the problem and this condition might be sufficient for finding a minimum. For instance, the Brachistochrone problem is mostly likely to have one particular curve that minimizes the functional, while there is no curve that maximizes the functional (one could easily find a curve which makes the value for the functional to tend to infinity), therefore finding an extremum curve that satisfies the Euler-Lagrange equation is enough.

The brachistochrone is a problem with the initial and final positions fixed, but that might not be the case for some other problems. In the case that the final points are not fixed, additional conditions have to be established.

Corollary 1 (Transversality Conditions)**.** *Let $I : X \to \mathbb{R}$ be a functional, in which $x^* \in X$ is an extremum function. If the function space X does not have any restriction on $x(t_0)$, then it is necessary that the additional condition is met*

$$\left.\frac{\partial F}{\partial \dot{x}}(x, \dot{x}, t)\right|_{t=t_0} = 0. \tag{3.66}$$

beside the Euler-Lagrange equation.

Likewise, if the final condition $x(t_f)$ is free the additional necessary condition is required

$$\left.\frac{\partial F}{\partial \dot{x}}(x, \dot{x}, t)\right|_{t=t_f} = 0 \tag{3.67}$$

in addition to the Euler-Lagrange equation.

If the problem has both, initial and final free boundaries, then both condition should be met.

Proof. In the proof of Theorem 4 we define δx such that $\delta x(t_0) = \delta x(t_f) = 0$. If the problem has a free initial condition the assumption that $\delta x(t_0) = 0$ is not necessary. As a consequence, to make the term outside the integral vanish in (3.57) at t_0, we should enforce

$$\left.\frac{\partial F}{\partial \dot{x}}(x, \dot{x}, t)\right|_{t=t_0} = 0. \tag{3.68}$$

An equivalent proposition can be made for a free final condition. □

Corollary 2 (Beltrami Identity)**.** *The Beltrami identity is a special case of the Euler-Lagrange equation. Consider the same assumptions of Theorem 4, with the particular assumption*

$$\frac{\partial F}{\partial t}(x, \dot{x}, t) = 0. \tag{3.69}$$

For this case, the necessary condition is given by

$$F - \frac{\partial F}{\partial \dot{x}}\dot{x} = C \tag{3.70}$$

where C is constant.

Proof. Multiplying the Euler-Lagrange equation by $\dot{x}$ we obtain

$$\frac{\partial F}{\partial x}\dot{x} = \left[\frac{d}{dt}\frac{\partial F}{\partial \dot{x}}\right]\dot{x}. \tag{3.71}$$

Applying the chain rule to obtain the total derivative of F with respect to t result in

$$\frac{dF}{dt} = \frac{\partial F}{\partial x}\dot{x} + \frac{\partial F}{\partial \dot{x}}\ddot{x} + \frac{\partial F}{\partial t}, \tag{3.72}$$

since $\frac{\partial F}{\partial t} = 0$, we can rearrange to

$$\frac{\partial F}{\partial x}\dot{x} = \frac{dF}{dt} - \frac{\partial F}{\partial \dot{x}}\ddot{x}. \tag{3.73}$$

Substituting into (3.71) leads to

$$\frac{dF}{dt} = \left[\frac{d}{dt}\frac{\partial F}{\partial \dot{x}}\right]\dot{x} + \frac{\partial F}{\partial \dot{x}}\ddot{x} \tag{3.74}$$

Using the product rule of the derivative, if we differentiate $\frac{\partial F}{\partial \dot{x}}\dot{x}$ with respect to t we obtain

$$\frac{d}{dt}\left[\frac{\partial F}{\partial \dot{x}}\dot{x}\right] = \left[\frac{d}{dt}\frac{\partial F}{\partial \dot{x}}\right]\dot{x} + \frac{\partial F}{\partial \dot{x}}\ddot{x} \tag{3.75}$$

Isolating the term $\left[\frac{d}{dt}\frac{\partial F}{\partial \dot{x}}\right]\dot{x}$ and substituting in (3.74)

$$\frac{dF}{dt} = \frac{d}{dt}\left[\frac{\partial F}{\partial \dot{x}}\dot{x}\right] - \frac{\partial F}{\partial \dot{x}}\ddot{x} + \frac{\partial F}{\partial \dot{x}}\ddot{x} \tag{3.76}$$

Eliminating the term $\frac{\partial F}{\partial \dot{x}}\ddot{x}$ and factorizing the derivative $\frac{d}{dt}$ leads to

$$\frac{d}{dt}\left(F - \frac{\partial F}{\partial \dot{x}}\dot{x}\right) = 0, \tag{3.77}$$

and by integrating with respect to t, we obtain

$$F - \frac{\partial F}{\partial \dot{x}}\dot{x} = C \tag{3.78}$$

where C is a constant. □

In the following, we solve the Brachistochrone problem using the Euler-Lagrange equation with the Beltrami identity.

Example 16 (Brachistochrone Problem and Euler-Lagrange Equation)**.** *From Example 14 we have that the time taken to travel is given by the functional*

$$T = \int_0^{t_f} \sqrt{\frac{1+\dot{x}^2}{2gx}}dt, \tag{3.79}$$

where the integrand function is

$$F_T(x, \dot{x}, t) = \sqrt{\frac{1+\dot{x}^2}{2gx}}. \tag{3.80}$$

We notice that F_T does not depend explicitly on the time variable t ($\frac{\partial F_T}{\partial t} = 0$), therefore we can use the Beltrami identity

$$F - \dot{x}\frac{\partial F}{\partial \dot{x}} = C. \tag{3.81}$$

Applying for F_T we have

$$\sqrt{\frac{1+\dot{x}^2}{2gx}} - \dot{x}\frac{\dot{x}}{\sqrt{(1+\dot{x}^2)}}\frac{1}{\sqrt{2gx}} = C \tag{3.82}$$

which yields

$$x(\dot{x}^2 + 1) = \frac{1}{2gC^2} \tag{3.83}$$

The solution of this ODE is given by the following parametric equation

$$t = \frac{1}{4gC^2}(\theta - \sin\theta), \tag{3.84a}$$

$$x = \frac{-1}{4gC^2}(\cos\theta - 1). \tag{3.84b}$$

If we bring back to the original system of variables we have

$$x = \frac{1}{4gC^2}(\theta - \sin\theta), \tag{3.85a}$$

$$y = \frac{-1}{4gC^2}(1 - \cos\theta). \tag{3.85b}$$

Let us assume that we have $(x_f, y_f) = (1, -1)$*, which yields the following nonlinear system of equations*

$$1 = \frac{1}{4gC^2}(\theta_f - \sin\theta_f) \tag{3.86a}$$

$$-1 = \frac{-1}{4gC^2}(1 - \cos\theta_f) \tag{3.86b}$$

Solving the system for C *and* θ_f*, we obtain* $C = 0.211$ *and* $\theta_f = 2.412$*. The resulting curve is given by Figure 3.3.*

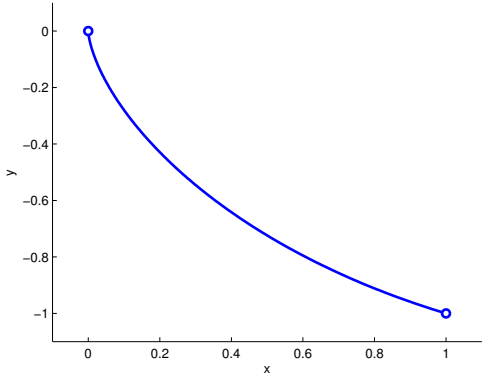

Figure 3.3: The solution of the Brachistochrone problem.

3.2 OPTIMAL CONTROL PROBLEMS[3]

Before addressing the problems with algebraic equations and constraints in the control variables, let us consider an initial case to introduce optimal control. The variables of this problem are the state function $x : [t_0, t_f] \to X$ and the control function $u : [t_0, t_f] \to U$. Since the problem has no constraints, the space of states is given by $X = \mathbb{R}^{N_x}$ and the space of controls is $U = \mathbb{R}^{N_u}$. More complex cases will be seen later. The simplest case for an optimal control problem (OCP), here denominated $\mathcal{P}_s$, can be stated in the form

$$\mathcal{P}_s : \quad \min_{u,t_f} \quad J(x,y,u) = \Phi(x(t_f),t_f) + \int_{t_0}^{t_f} L(x,u,t)\,dt \tag{3.87a}$$

$$\text{s.t.:} \quad \dot{x} = f(x,u,t) \tag{3.87b}$$

$$x(t_0) = x_0 \tag{3.87c}$$

where Φ is the final cost function (known as the Mayer term), L is the dynamic cost (known as the Lagrange term), and f is the dynamic function. The vector x is the state, u is the control, and t is the time. The values of x_0 and t_0 are fixed. The final time t_f and the final state x_f might be fixed or not, independently, herein all the cases will be considered.

Some basic properties on the underlying functions are required in order to problem $\mathcal{P}_s$ to be well defined.

Assumption 1. *With respect to problem $\mathcal{P}_s$ (3.87), the following assumptions are made:*

1. *The function Φ is continuously differentiable with respect to x and t.*
2. *The function L is continuously differentiable with respect to x, u, and t.*
3. *The dynamic function f is continuously differentiable with respect to x, u, and t.*
4. *The function x is continuously differentiable with respect to t, and the function u is piecewise continuous with respect to t.*

[3]This section was written mainly based on [17], however other sources were used as [18, 19]

If the third assumption holds and u is piecewise continuous, then the solution of the ODE system (3.87b) *exists and is unique ([20, Theorem 3.2]).*

In the previous section the functional which was being minimized only took into account the integral term, given by the integral of function L. At the same time, some numerical solvers only accept the minimization of a final cost function, as Φ in problem $\mathcal{P}_s$. The following theorem shows that they are interchangeable.

Theorem 5. *If an OCP has the following objective*

$$\min \quad \Phi(x(t_f)) + \int_{t_0}^{t_f} L(x,u,t)\,dt, \tag{3.88}$$

and $\Phi(x(t_f),t_f)$ is continuously differentiable in x and t, then there is an equivalent objective that only has the Lagrangian term,

$$\min \quad \int_{t_0}^{t_f} \widehat{L}(x,u,t)\,dt. \tag{3.89}$$

Conversely, it is possible to find a representation of (3.88) *that has only the Mayer term,*

$$\min \quad \widehat{\Phi}(x_{t_f}). \tag{3.90}$$

Proof. The first statement can be written as

$$\Phi(x(t_f)) = \Phi(x(t_0)) + \Phi(x(t_f)) - \Phi(x(t_0)) \tag{3.91}$$

which is the same as

$$\Phi(x(t_f)) = \Phi(x(t_0)) + \int_{t_0}^{t_f} \frac{d\Phi}{dt}(x,t)\,dt \tag{3.92a}$$

$$= \Phi(x(t_0)) + \int_{t_0}^{t_f} \left[\frac{\partial\Phi}{\partial t}(x,t) + \frac{\partial\Phi}{\partial x}(x,t)\frac{dx}{dt}\right] dt \tag{3.92b}$$

$$= \Phi(x(t_0)) + \int_{t_0}^{t_f} \left[\frac{\partial\Phi}{\partial t}(x,t) + \frac{\partial\Phi}{\partial x}(x,t) f(x,u,t)\right] dt. \tag{3.92c}$$

Since $x(t_0)$ and t_0 are fixed, $\Phi(x(t_0),t_0)$ is fixed by consequence. Therefore, the initial condition does not affect the solution and we can choose $\widehat{L}$ in the form

$$\widehat{L}(x,u,t) = L(x,u,t) + L_\Phi(x,u,t). \tag{3.93}$$

where

$$L_\Phi = \frac{\partial \Phi(x)}{\partial t} + \frac{\partial \Phi(x)}{\partial x} f(x,u,t) \tag{3.94}$$

and the minimization of the integral of $\widehat{L}$ will be equivalent to (3.88).

For the second statement, let us introduce a new state x_L such that

$$\dot{x}_L = L(x,u,t), \tag{3.95}$$

and the initial condition is $x_L(t_0) = 0$. Therefore we can define

$$\widehat{\Phi}(t_f) = \Phi(x(t_f)) + \int_{t_0}^{t_f} L(x,u,t)\, dt \tag{3.96a}$$

$$= \Phi(x(t_f)) + \int_{t_0}^{t_f} \dot{x}_L(t)\, dt \tag{3.96b}$$

$$= \Phi(x(t_f)) + x_L(t_f). \tag{3.96c}$$

□

In calculus of variations to check if a path x is optimal for a functional I, we developed the Euler-Lagrange equations. Likewise, we can develop a condition for the optimality of control u^* that induces an optimal path x^*.

Theorem 6 ([17, **?**])**.** *Let $\mathcal{P}_s$ (3.87) be an OCP for which Assumption 1 holds. If the control profile u^* defined in the interval $[t_0, t_f]$, which induces the state profile x^*, is a minimum for $\mathcal{P}_s$ then there exists a function $\lambda^* : [t_0, t_f] \to \mathbb{R}^{N_x} \in \mathcal{C}^1[t_0, t_f]$ that satisfies*

$$-\dot{\lambda}^* = \frac{\partial L}{\partial x}(x^*,u^*,t)^T + \frac{\partial f}{\partial x}(x^*,u^*,t)^T \lambda^*, \qquad t \in [t_0, t_f] \tag{3.97a}$$

$$0 = \frac{\partial L}{\partial u}(x^*,u^*,t) + \lambda^{*T} \frac{\partial f}{\partial u}(x^*,u^*,t) \qquad t \in [t_0, t_f], \tag{3.97b}$$

and x^* is given by

$$\dot{x}^* = f(x^*,u^*,t), \qquad t \in [t_0, t_f] \tag{3.97c}$$

$$x^*(t_0) = x_0. \tag{3.97d}$$

Proof. In optimization theory, the Lagrangian is a function that augments the objective function by incorporating the constraints multiplied by a vector, known as Lagrangian multipliers. Likewise we can define an equivalent augmented functional that combines the final cost, the dynamic cost function, and the system dynamic equation adjointed by a variable $\lambda(t) \in \mathcal{C}^1[t_0, t_f]$, as follows

$$J_a(x, u, \lambda) = \Phi(x(t_f), t_f) + \int_{t_0}^{t_f} \left\{ L(x, u, t) + \lambda^T \left[f(x, u, t) - \dot{x} \right] \right\} dt \tag{3.98}$$

where λ can be referred to as the adjoint variable or the costate. Notice that if x^* satisfies the ODE equation $\dot{x}^* = f(x^*, u^*, t)$, then $J_a = J$ (3.87a).

From Theorem 1, if x^*, u^*, and some λ^* are the extremum of J_a, then the first variation of the augmented functional J_a has to be zero. Let δx with $\delta x(t_0) = 0$ be the variation of the state x, δu be variation of u, the $\delta\lambda$ be the variation of the adjoint variable λ, and δt_f be a variation on the final time t_f. The first variation of J_a, using Theorem 3, at x^*, λ^*, and u^* is given by

$$\begin{aligned}
\delta J_a = & \left[\frac{\partial \Phi}{\partial x}(x^*(t_f), t_f) \right] \delta x(t_f) + \left[\frac{\partial \Phi}{\partial t}(x^*(t_f), t_f) \right. \\
& + \frac{\partial \Phi}{\partial x}(x^*(t_f), t_f) \dot{x}^*(t_f) + L(x^*(t_f), u^*(t_f), t_f) \\
& \left. + \lambda^{*T} \left[f(x^*(t_f), u^*(t_f), t_f) - \dot{x}^*(t_f) \right] \right] \delta t_f \\
& + \int_{t_0}^{t_f} \left\{ \left[\frac{\partial L}{\partial x}(x^*, u^*, t) + \lambda^{*T} \frac{\partial f}{\partial x}(x^*, u^*, t) \right] \delta x \right. \\
& + \left[\frac{\partial L}{\partial u}(x^*, u^*, t) + \lambda^{*T} \frac{\partial f}{\partial u}(x^*, u^*, t) \right] \delta u \\
& \left. + \left[f(x^*, u^*, t) - \dot{x}^* \right]^T \delta\lambda - \lambda^{*T} \delta\dot{x} \right\} dt
\end{aligned} \tag{3.99}$$

where $\delta\dot{x}$ is the derivative of the perturbation δx, and $\delta x(t_f)$ is the value of δx at time t_f. The terms $L(x^*(t_f), u^*(t_f), t_f) + \lambda^{*T} \left[f(x^*(t_f), u^*(t_f), t_f) - \dot{x}^*(t_f) \right]$ appears multiplied by δt_f are appears from the differentiation of the integral with respect to t_f, and the terms related to $\Phi(x(t_f), t_f)$ originate from the first order Tay-

lor's expansion

$$\delta\Phi(x(t_f),t_f) = \left[\frac{\partial\Phi}{\partial t}(x^*(t_f),t_f) + \frac{\partial\Phi}{\partial x}(x^*(t_f),t_f)\dot{x}\right]\delta t_f + \frac{\partial\Phi}{\partial x}(x^*(t_f),t_f)\delta x(t_f). \quad (3.100)$$

The same approach used to handle the $\delta\dot{x}$ term in the Euler-Lagrange equation can be used here, using the integration by parts, the term related to $\delta\dot{x}$ can be transformed

$$\int_{t_0}^{t_f} -\lambda^{*T}\delta\dot{x}\,dt = -\lambda^{*T}(t_f)\delta x(t_f) + \int_{t_0}^{t_f} \dot{\lambda}^{*T}\delta x\,dt \quad (3.101)$$

where the term $\lambda^{*T}(t_0)\delta x(t_0)$ does not appear since $\delta x(t_0) = 0$.

The term $\delta x(t_f)$ is a perturbation on the variable $x^*(t_f)$, which is defined by the value on the x axis at time $t = t_f$. Further, let us recall that δt_f is a perturbation at the point $t = t_f$ in the time axis. Let us define the perturbation δx_f as a perturbation on the final value of the trajectory of x^*, that is x_f. Notice that $\delta x(t_f) = \delta x_f$ only if t_f is the final time, but if the final time is perturbed then the final time is $t_f + \delta t_f$ and the final state is $x_f = x(t_f + \delta t_f)$.

Observing Figure 3.4, it can be noticed that by perturbing the value of $x(t)$ at $t = t_f$, that is $x(t_f)$, the value of the end of the trajectory x_f is directly affected. In addition, note that by perturbing the final time t_f the value of the end of trajectory x_f is also affected depending on the inclination of trajectory, $\dot{x}$.

As observed in Figure 3.4, we have the following relation

$$\delta x_f = \delta x(t_f) + \dot{x}\delta t_f \quad (3.102)$$

which can be rearranged in the form

$$\delta x(t_f) = \delta x_f - \dot{x}\delta t_f \quad (3.103)$$

Using equations (3.101), (3.103), and the fact that $\dot{x}(t_f) = f(x(t_f),u(t_f),t_f)$, the first variation of the augmented functional

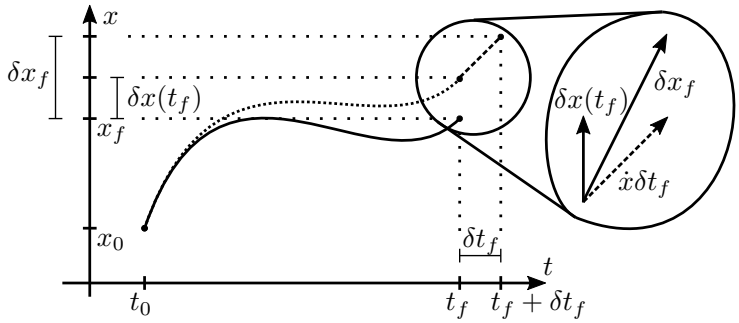

Figure 3.4: Illustrations on the variations δx_f, $\delta x(t_f)$, and δt_f. The solid line is the original trajectory and dotted line is the perturbed trajectory.

can be rewritten as

$$\begin{aligned}\delta J_a = &\left[L(x^*(t_f),u^*(t_f),t_f) + \lambda^*(t_f)^T f(x^*(t_f),u^*(t_f),t_f)\right.\\ &\left.+\frac{\partial \Phi}{\partial t}(x^*(t_f),t_f)\right]\delta t_f + \left[\frac{\partial \Phi}{\partial x}(x^*(t_f),t_f) - \lambda^*(t_f)\right]\delta x_f\\ &+\int_{t_0}^{t_f}\left\{\left[\frac{\partial L}{\partial x}(x^*,u^*,t) + \lambda^{*T}\frac{\partial f}{\partial x}(x^*,u^*,t) + \dot{\lambda}^{*T}\right]\delta x\right.\\ &\qquad+\left[\frac{\partial L}{\partial u}(x^*,u^*,t) + \lambda^{*T}\frac{\partial f}{\partial u}(x^*,u^*,t)\right]\delta u\\ &\qquad\left.+\left[f(x^*,u^*,t) - \dot{x}^*\right]^T\delta\lambda\right\}dt. \quad (3.104)\end{aligned}$$

Since it is required that $\delta J_a = 0$, for the trajectories (x^*, u^*, t) be an extremum, Lemma 1 can be applied to the integral term in (3.104) obtaining the necessary conditions

$$-\dot{\lambda}^* = \frac{\partial L}{\partial x}(x^*,u^*,t)^T + \frac{\partial f}{\partial x}(x^*,u^*,t)^T\lambda^* \quad (3.105a)$$

$$0 = \frac{\partial L}{\partial u}(x^*,u^*,t) + \lambda^{*T}\frac{\partial f}{\partial u}(x^*,u^*,t) \quad (3.105b)$$

$$\dot{x}^* = f(x^*,u^*,t) \quad (3.105c)$$

for all $t \in [t_0, t_f]$.

By meeting the conditions (3.105), the first variation of δJ_a becomes

$$\delta J_a = \Bigg[L(x^*(t_f), u^*(t_f), t_f) + \lambda^*(t_f)^T f(x^*(t_f), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f) \Bigg] \delta t_f + \left[\frac{\partial \Phi}{\partial x}(x^*(t_f), t_f) - \lambda^*(t_f) \right] \delta x_f. \quad (3.106)$$

Since the final time t_f and $x(t_f)$ might be fixed or not, the development of further conditions will be given separately depending if those variables are fixed or not. □

The necessary conditions (3.105) for (x^*, u^*, λ^*) to be optimal to problem $\mathcal{P}_s$ (3.87) can be expressed in the form of a Hamiltonian system. Let us define an additional state x_L, as done in Theorem 5, such that the differential equation is given by

$$\dot{x}_L = L(x, u, t) \quad (3.107)$$

which is associated to an adjoint state λ_L. Let the vector function $\boldsymbol{x} : [t_0, t_f] \to \mathbb{R}^{N_x+1}$ be an augmented state vector that accounts for x_L and for $x_1, \ldots, x_{N_x}$, such that

$$\boldsymbol{x} = [x_L, x_1, \ldots, x_{N_x}]. \quad (3.108)$$

In addition, let $\boldsymbol{\lambda} : [t_0, t_f] \to \mathbb{R}^{N_x+1}$ be an augmented vector for the costates λ_L and $\lambda_1, \ldots, \lambda_{N_x}$, defined in the form

$$\boldsymbol{\lambda} = [\lambda_L, \lambda_1, \ldots, \lambda_{N_x}]. \quad (3.109)$$

Finally, let the function $\boldsymbol{f}$ be given by

$$\boldsymbol{f} = [L, f_1, \ldots, f_{N_x}], \quad (3.110)$$

where $\boldsymbol{f}$ only depends on x, u, and t but not on x_L.

If a Hamiltonian function is given by

$$\boldsymbol{H} = \boldsymbol{\lambda}^T \boldsymbol{f}(x, u, t) \qquad \forall t \in [t_0, t_f] \quad (3.111)$$

where $\boldsymbol{f}$ has as an argument x, not $\boldsymbol{x}$, since it does not depend on x_L.

The ODE system formed by the conditions (3.97c) and (3.97a) can be expressed by the Hamiltonian system

$$\dot{\boldsymbol{x}} = \frac{\partial \boldsymbol{H}}{\partial \boldsymbol{\lambda}} = \boldsymbol{f}(x, u, t), \quad (3.112a)$$

$$-\dot{\boldsymbol{\lambda}} = \frac{\partial \boldsymbol{H}}{\partial \boldsymbol{x}} = \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}, \quad (3.112b)$$

and the condition on the control variable (3.97b) is given by

$$\frac{\partial \boldsymbol{H}}{\partial u} = \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{f}}{\partial u}(x, u, t) = 0 \tag{3.113}$$

Notice that

$$-\dot{\lambda}_L = \frac{\partial \boldsymbol{H}}{\partial x_L} = 0 \tag{3.114}$$

and therefore $\lambda_L(t) = q$, for some real constant q. Consider the cases for the values of q:

1. If $q < 0$, then the Hamiltonian is given by

$$\boldsymbol{H}(\boldsymbol{\lambda}, x, u) = qL(x, u, t) + \lambda^T f(x, u, t), \tag{3.115}$$

   Then looking for an infimum of $\boldsymbol{H}$ will lead to a maximum of L, because the $qL(x, u, t)$ will have the opposite sign of $L(x, u, t)$. Therefore negative numbers for q are not accepted.

2. If $q = 0$, then the Hamiltonian is given by

$$\boldsymbol{H}(\boldsymbol{\lambda}, x, u) = 0 \times L(x, u, t) + \lambda^T f(x, u, t). \tag{3.116}$$

   which means that the function L does not affect the solution $u^*(t)$. This particular problem, in which $q = 0$, is known as "abnormal problem". Although these problems are valid, they are not considered in this work for their particularity.

3. If $q > 0$, then the Hamiltonian is given by

$$\boldsymbol{H}(\boldsymbol{\lambda}, x, u) = qL(x, u, t) + \lambda^T f(x, u, t). \tag{3.117}$$

   Notice, however, that by multiplying an objective function of an optimization problem by a scalar the solution does not change. So for any $q > 0$, it is possible to find a function $\widehat{L}$ for a $\widehat{q} = 1$ such that

$$\widehat{q}\widehat{L}(x, u, t) = qL(x, u, t). \tag{3.118}$$

   Therefore the case which $q > 0$ can be analyzed considering $q = 1$.

Therefore, we assume$\lambda_L(t) = 1$ for all $t \in [t_0, t_f]$. Let us define the Hamiltonian function in the most common form,

$$H(x, \lambda, u, t) = L(x, u, t) + \lambda^T f(x, u, t) \tag{3.119}$$

which is equivalent to the Hamiltonian of (3.111). Using the Hamiltonian H, the necessary condition can be written as

$$\dot{x} = \frac{\partial H}{\partial \lambda} = f(x, u, t), \tag{3.120a}$$

$$-\dot{\lambda} = \frac{\partial H}{\partial x} = \frac{\partial L}{\partial x}^T + \frac{\partial f}{\partial x}^T \lambda, \tag{3.120b}$$

$$0 = \frac{\partial H}{\partial u} = \frac{\partial L}{\partial u} + \lambda^T \frac{\partial f}{\partial u}. \tag{3.120c}$$

The DAE system given by the conditions (3.120) has $2N_x$ differential equations, the state and costate equations, and N_u algebraic equations which have to be satisfied in the interval $[t_0, t_f]$. The solution of the differential equations requires $2N_x$ boundary conditions. The initial condition $x(t_0) = x_0$ gives N_x of the required boundary conditions. So it is necessary to define the remaining N_x boundary conditions and an additional condition for the cases that t_f is not fixed. These conditions will be obtained from the terms left in J_a after the application of the necessary conditions of Theorem 6, from (3.106) we obtain

$$\begin{aligned}\delta J_a = &\Bigg[L(x^*(t_f), u^*(t_f), t_f) + \lambda^*(t_f)^T f(x^*(t_f), u^*(t_f), t_f) \\ &+ \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f)\Bigg] \delta t_f + \left[\frac{\partial \Phi}{\partial x}(x^*(t_f), t_f) - \lambda^*(t_f)^T\right] \delta x_f.\end{aligned} \tag{3.121}$$

The following two subsections are based in [17] and [18].

3.2.1 Problems with Fixed Final Time

This section will treat the particular case of the optimal control problem $\mathcal{P}_s$ for which the final time t_f is fixed. Therefore herein we will be looking for N_x equations to complement the conditions defined by (3.120).

Since the final time is fixed, the perturbation variable δt_f can only assume the value zero,

$$\delta t_f = 0, \tag{3.122}$$

applying it to terms left on the first variation of the augmented functional δJ_a (3.121), we obtain

$$\delta J_a = \left[\frac{\partial \Phi}{\partial x}(x(t_f), t_f) - \lambda^*(t_f)^T\right] \delta x_f. \tag{3.123}$$

This category can be split in three cases: with fixed final state, with free final state, with final state lying in some manifold. The conditions will be assessed for each case:

1. **Given Final State:** Since $x(t_f)$ and t_f are specified, the perturbations are given by $\delta t_f = 0$ and $\delta x_f = 0$. The additional boundary conditions are given by the N_x equations below

$$x^*(t_f) = x_f. \tag{3.124}$$

2. **Free Final State:** If the final state $x(t_f)$ is free but the final time t_f is fixed, then the perturbation $\delta x(t_f)$ is only affected by δx_f, while $\delta t_f = 0$. Therefore in order to make the value of δJ_a (3.123) to be zero, the following equation must be satisfied

$$\lambda^*(t_f) = \frac{\partial \Phi}{\partial x}(x^*(t_f), t_f)^T \tag{3.125}$$

which establishes N_x boundary conditions.

3. **Final state lying on a manifold:** Before going into the conditions, let us go through a brief review of manifolds.

Let h be a function from some region of the N_x-dimensional Euclidean space X to $\mathbb{R}$. Then a hypersurface S in space X can be defined by

$$S = \{\mathrm{x} \in X \mid h(\mathrm{x}) = 0\}. \tag{3.126}$$

Let the partial derivatives of h with respect to x be continuous. A point $\mathrm{x} \in S$ at which the condition

$$\frac{\partial h}{\partial \mathrm{x}_1}(\mathrm{x}) = \frac{\partial h}{\partial \mathrm{x}_2}(\mathrm{x}) = \ldots = \frac{\partial h}{\partial \mathrm{x}_{N_x}}(\mathrm{x}) = 0 \tag{3.127}$$

is satisfied is called a singular point in S. The other points of S, the points where all the partial derivatives do not vanish, are called nonsingular points. A hypersurface that is formed by a

continuously differentiable function and that has no singular point is called a smooth surface.

Let x_0 be an arbitrary point of the smooth surface S given by (3.126), then the vector $n(\mathrm{x}_0) = \frac{\partial h}{\partial \mathrm{x}}(\mathrm{x}_0)$ is called the normal vector of S at point x_0.

As an example, consider the particular case $X = \mathbb{R}^2$. The hypersurface formed by

$$h(\mathrm{x}_1, \mathrm{x}_2) = 0 \tag{3.128}$$

is the traditional notion of a curve in the $\mathbb{R}^2$ space, and h shall be differentiable with respect to x_1 and x_2.

Now consider the case that the hypersurface is linear. In this case, the hypersurface can be represented generically by

$$h(\mathrm{x}) = a_1\mathrm{x}_1 + a_2\mathrm{x}_2 + \ldots + a_{N_x}\mathrm{x}_{N_x} + b = 0. \tag{3.129}$$

where $a_i \in \mathbb{R}$ are the coefficients. This surface is smooth, has no singular point, if at least one of the coefficients a_i is nonzero. For a generic $\mathbb{R}^{N_x}$ space the surface formed by this linear functions is called as hyperplane. For the particular case where $N_x = 2$, the hyperplane is a straight line, and for the case where $N_x = 3$ the hyperplane is a traditional plane. Every hyperplane can be uniquely defined by a normal vector and a point in this hyperplane.

If S is a smooth hypersurface defined by some function h, and x_0 is one of its points, then the hyperplane T passing through x_0, having the vector $n(\mathrm{x}_0)$ as its normal, is called the tangent hyperplane of the hypersurface S at point x_0, as illustrated in Figure 3.5. Each vector emanating from x_0 is a tangent vector of S if and only if it is orthogonal to the normal vector $n(\mathrm{x}_0)$, which are illustrated in Figure 3.5 by the arrows originating from x_0.

Let $S_1, S_2, \ldots, S_K$ be smooth hypersurfaces in space X given by

$$\begin{aligned} h_1(\mathrm{x}) &= 0 \\ h_2(\mathrm{x}) &= 0 \\ &\vdots \\ h_K(\mathrm{x}) &= 0 \end{aligned} \tag{3.130}$$

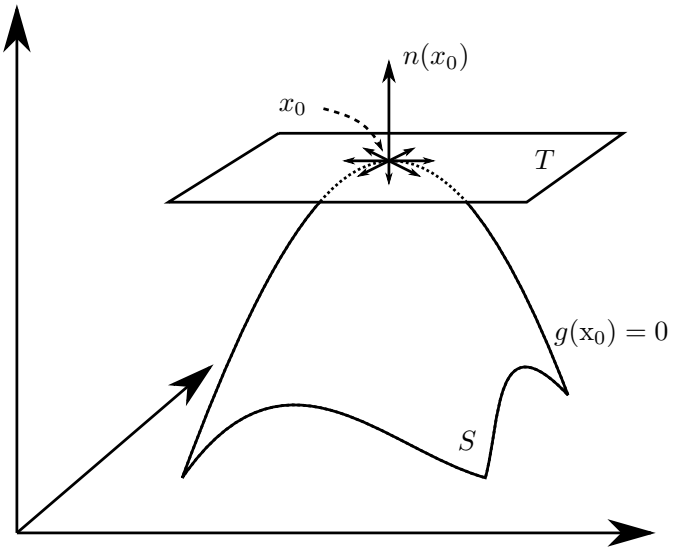

Figure 3.5: Illustration of the definition of the tangent hyperplane of a hypersurface in $X = \mathbb{R}^3$ at point x_0.

respectively.

The intersection M of all these hypersurfaces (given in (3.130)) is called a $(N_x - K)$-dimensional smooth manifold in X if for every $\mathrm{x} \in M$ the vector of partial derivatives,

$$\frac{\partial h_j}{\partial \mathrm{x}} \tag{3.131}$$

with $j = 1, \ldots, K$, are linearly independent.

The linear independence of the partial derivatives implies that the Jacobian of h with respect to x,

$$\frac{\partial h}{\partial \mathrm{x}} = \begin{bmatrix} \frac{\partial h_1}{\partial \mathrm{x}_1} & \frac{\partial h_1}{\partial \mathrm{x}_2} & \cdots & \frac{\partial h_1}{\partial \mathrm{x}_{N_x}} \\ \frac{\partial h_2}{\partial \mathrm{x}_1} & \frac{\partial h_2}{\partial \mathrm{x}_2} & \cdots & \frac{\partial h_2}{\partial \mathrm{x}_{N_x}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial h_K}{\partial h_1} & \frac{\partial h_K}{\partial x_2} & \cdots & \frac{\partial h_K}{\partial \mathrm{x}_{N_x}} \end{bmatrix} \tag{3.132}$$

has full rank.

A R-dimensional manifold in space X is defined by $(N_x - R)$ equations. For the case in which $R = N_x - 1$, the manifold is

given by only one equation and its definition coincides with the definition of a hypersurface.

Let M be a $(N_x - K)$-dimensional manifold in X formed by the intersection of $S_1, \ldots, S_K$. Let L_i be a tangent plane of the hypersurface S_i at point x_0. The intersection L of the $L_1, \ldots, L_K$ tangent planes is an $(N_x - K)$-dimensional plane which is called the tangent plane of M at the point x_0. Moreover, a vector emanating from $x_0 \in M$ is a tangent vector of M if it lies on all hyperplanes L, equivalently, if it is orthogonal to the normal vectors of $S_1, \ldots, S_K$ at point x_0.

Returning to the problem of interest, if it is desired that the final states lies in a $(N_x - 1)$-dimensional manifold (a surface) formed by $h(x(t_f)) = 0$, for instance,

$$h(x(t_f)) = [x_1(t_f) - 3]^2 + [x_2(t_f) - 4]^2 - 4 = 0, \quad (3.133)$$

then an admissible direction of $\delta x(t_f) = \delta x_f$ has to be a direction tangent to the circle formed by this manifold. Figure 3.6 depicts the manifold (3.133). Notice that if we choose a non-tangent direction, then a perturbation will move the point so it does not satisfy $h(x(t_f)) = 0$. On the other hand, an infinitely small perturbation tangent to the circle will lie sufficiently close to the circle.

The normal vector of a hypersurface at a point is equal to its partial derivatives at that given point, therefore we obtain the normal vector at point $x^*(t_f)$,

$$\frac{\partial h}{\partial x}(x^*(t_f)) = \begin{bmatrix} 2\,[x_1^*(t_f) - 3] \\ 2\,[x_2^*(t_f) - 4] \end{bmatrix}^T \quad (3.134)$$

and, as seen previously, the tangent vectors at a point are those emanating from the point and that are orthogonal to the normal at that given point, therefore we require that for all admissible δx_f

$$\begin{bmatrix} 2\,[x_1^*(t_f) - 3] \\ 2\,[x_2^*(t_f) - 4] \end{bmatrix}^T \delta x_f = 0 \quad (3.135)$$

which is generalized in the equation

$$\frac{\partial h}{\partial x}(x^*(t_f))\delta x_f = 0 \quad (3.136)$$

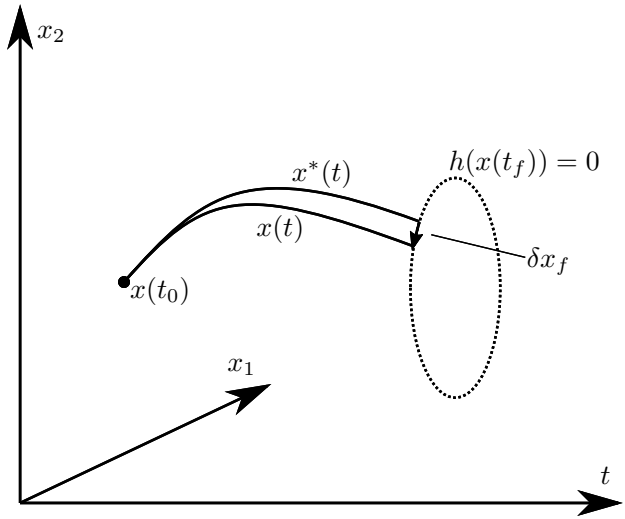

Figure 3.6: Illustration of the manifold formed by $h(x(t_f)) = 0$ and the direction of δx_f.

At the same time, according to (3.123), the following equation must be satisfied

$$\left[\frac{\partial \Phi}{\partial x}(x^*(t_f), t_f) - \lambda^*(t_f)^T\right] \delta x_f = 0 \tag{3.137}$$

for all δx_f that satisfy (3.136), which means that they are orthogonal.

So, if $\frac{\partial h}{\partial x}(x^*(t_f))$ is orthogonal to all δx_f, and $[\frac{\partial \Phi}{\partial x}(x^*(t_f), t_f) - \lambda^*(t_f)^T]$ is also orthogonal to all δx_f. Then $\frac{\partial h}{\partial x}(x^*(t_f))$ and $[\frac{\partial \Phi}{\partial x}(x^*(t_f), t_f) - \lambda^*(t_f)^T]$ are parallel. This can be visualized in Figure 3.5, where the normal vector $n(\mathrm{x}_0)$ is $\frac{\partial h}{\partial x}(x^*(t_f))$. Notice that all vectors orthogonal to the normal vector at point x_0 (which are the vectors δx_f) form a plane T. The only direction that another vector can take to be orthogonal to all vectors δx_f are either in the same direction of $n(\mathrm{x}_0)$ or in the opposite direction. Therefore $\frac{\partial h}{\partial x}(x^*(t_f))$ and $[\frac{\partial \Phi}{\partial x}(x^*(t_f), t_f) - \lambda^*(t_f)^T]$ are parallel.

Therefore, there exists a scalar $\hat{\eta}$ such that

$$\left[\frac{\partial \Phi}{\partial x}(x^*(t_f), t_f) - \lambda^*(t_f)^T\right] = \hat{\eta}^T \frac{\partial h}{\partial x}(x^*(t_f)) \tag{3.138}$$

In the case that h is not a hypersurface but a K-dimensional manifold, then δx has to be orthogonal to the normal vectors of every hypersurface

$$h_k(x(t_f)) = 0, \qquad \text{for all } k = 1, \ldots, K \tag{3.139}$$

at point $x^*(t_f)$, which means that (3.136) repeats for all h_k hypersurface,

$$\frac{\partial h_k}{\partial x}(x^*(t_f))\delta x_f = 0 \tag{3.140}$$

Hence, for every scalar $\alpha \neq 0$ there exists a vector $\hat{\eta} = [\hat{\eta}_1, \ldots, \hat{\eta}_K]$ such that

$$\alpha\left[\frac{\partial \Phi}{\partial x}(x^*(t_f), t_f) - \lambda^*(t_f)^T\right] = \hat{\eta}_1 \frac{\partial h_1}{\partial x}(x^*(t_f)) + \ldots + \hat{\eta}_K \frac{\partial h_K}{\partial x}(x^*(t_f)) \tag{3.141}$$

or, by defining $\eta = \alpha^{-1}\hat{\eta}$,

$$\lambda^*(t_f) = \frac{\partial \Phi}{\partial x}(x^*(t_f), t_f)^T - \frac{\partial h}{\partial x}(x^*(t_f))^T \eta \tag{3.142}$$

which establishes N_x relations. With the inclusion of the η vector, K additional relations are required. These can be obtained from the K equations of the manifold,

$$h(x(t_f)) = 0. \tag{3.143}$$

3.2.2 Problems with Free Final Time

This section considers the case where t_f is free to assume any value and δt_f as well. Then $N_x + 1$ boundary conditions are needed to complement the conditions (3.120). According to (3.121), the first variation of the augmented functional is given by

$$\delta J_a = \left[L(x^*(t_f), u^*(t_f), t_f) + \lambda^*(t_f)^T f(x^*(t_f), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f)\right]\delta t_f + \left[\frac{\partial \Phi}{\partial x}(x^*(t_f), t_f) - \lambda^*(t_f)^T\right]\delta x_f \tag{3.144}$$

or, in terms of H,

$$\delta J_a = \left[H(x^*(t_f), \lambda^*(t), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f) \right] \delta t_f + \left[\frac{\partial \Phi}{\partial x}(x(t_f), t_f) - \lambda^*(t_f)^T \right] \delta x_f. \quad (3.145)$$

An optimal control problem with free final time can come in different forms, the main possibilities being:

1. **Final State Fixed:** If the final state is fixed, then the equation

$$x^*(t_f) = x_f \quad (3.146)$$

should be satisfied, which gives N_x relations, and the perturbation δx_f is zero. To bring the first variation of J_a to zero the following equation must be satisfied

$$H(x^*(t_f), \lambda^*(t), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f) = 0 \quad (3.147)$$

which secures the remaining condition.

2. **Final State Free:** In the case that the final state and the final time are free, the condition to make δJ_a equal to zero is

$$H(x^*(t_f), \lambda^*(t), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f) = 0 \quad (3.148a)$$

$$\lambda^*(t_f) = \frac{\partial \Phi}{\partial x}(x^*(t_f), t_f)^T \quad (3.148b)$$

3. **Final State Lies on a Moving Point:** Consider that the optimal control problem is related to launching a satellite and the target is to park it close to another satellite already in orbit. Since the target satellite revolves around the Earth, its position is a function of time, so the final position of the launched satellite is also a function of time, namely $\theta(t)$.

So we are looking for the conditions of an optimal control problem that has its final position at a point in $\theta(t)$. Since the final state varies with time, the perturbation δx_f can be written in terms of the perturbation δt_f,

$$\delta x_f = \frac{\partial \theta}{\partial t}(t_f) \delta t_f \quad (3.149)$$

and making the substitutions in (3.145), yields the condition

$$H(x^*(t_f), \lambda^*(t), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f) + \left[\frac{\partial \Phi}{\partial x}(x(t_f), t_f) - \lambda^*(t_f)\right]^T \frac{\partial \theta}{\partial t}(t_f) = 0 \tag{3.150}$$

which imposes one boundary conditions. The remaining N_x conditions are obtained from the equation

$$x^*(t_f) = \theta(t_f) \tag{3.151}$$

4. **Final State Lies on a Manifold:** The case of final state lying on a manifold with free final time is similar to the case with fixed final time. The manifold is given by the equation

$$h(x(t_f)) = 0 \tag{3.152}$$

which does not depend on the final time explicitly.

Since the manifold function does not depend on t_f, the circle drawn in Figure 3.6 becomes a cylinder with a circle projected in the $x_1 \times x_2$ plane.

So, in order to make the first variation of the augmented functional to be zero (3.145), the term related to the perturbation δt_f has to be zero,

$$H(x^*(t_f), \lambda^*(t), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f) = 0 \tag{3.153}$$

which gives one of the $N_x + 1$ conditions.

The development of the remaining condition is the same as in case 3 of the fixed final time, Section 3.2.1. If $h(x(t_f)) = 0$ is a $(N_x - K)$-dimensional manifold, then by including an additional nonzero vector $\eta \in \mathbb{R}^K$ it is required $N_x + K$ equations, from which K comes from satisfying the manifold equation,

$$h(x(t_f)) = 0, \tag{3.154}$$

and N_x equations come from a perturbation in a parallel plane to the manifold,

$$\lambda^*(t_f) = \frac{\partial \Phi}{\partial x}(x(t_f), t_f)^T - \frac{\partial h}{\partial x}(x^*(t_f))^T \eta. \tag{3.155}$$

5. **Final State Lies on a Moving Manifold:** The last and more general case happens when the final state has to lie on a manifold that changes with time, which means that the equation of the manifold explicitly depends on the time variable,

$$h(x(t_f), t_f) = 0. \tag{3.156}$$

For instance consider a space $X = \mathbb{R}^2$ and a 1-dimensional manifold,

$$h_1(x(t_f)) = [x_1(t_f) - 3]^2 + [x_2(t_f) - 4 - t]^2 - 4 = 0, \tag{3.157}$$

which is depicted in Figure 3.7. The partial derivatives of h_1 are

$$\frac{\partial h_1}{\partial x} = \begin{bmatrix} 2[x_1(t_f) - 3] \\ 2[x_2(t_f) - 4 - t]. \end{bmatrix} \tag{3.158a}$$

$$\frac{\partial h_1}{\partial t} = 2[x_2(t_f) - 4] \tag{3.158b}$$

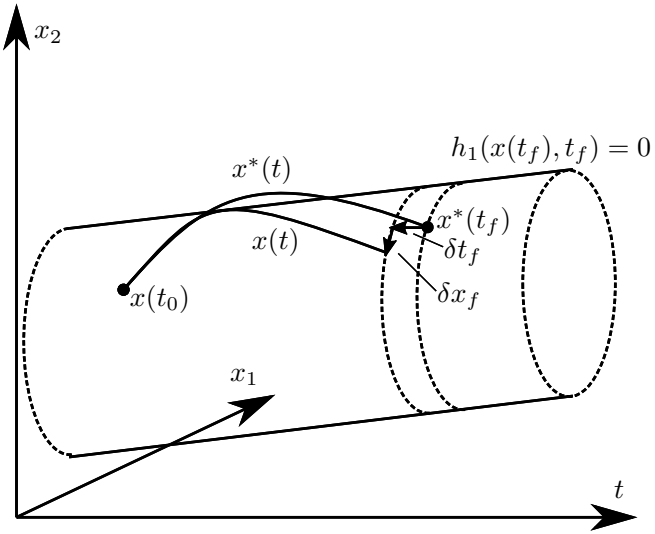

Figure 3.7: Illustration of the manifold formed by $h(x(t_f), t_f) = 0$ and the direction of δx_f and δt_f.

Notice that in order to the perturbations δx_f and δt_f to be admissible they should move in a plane tangent to the surface $h_1(x(t_f), t_f) = 0$. To lie in a tangent plane, all perturbations δx_f and δt_f have to be orthogonal to the normal vector of h_1 at the point $(x^*(t_f), t_f)$, which means that for δx_f the equation

$$\left[\frac{\partial h_1}{\partial x}\right] \delta x_f = 0 \tag{3.159}$$

must be satisfied, and the equivalent for the perturbation δt_f is

$$\left[\frac{\partial h_1}{\partial t}\right] \delta t_f = 0. \tag{3.160}$$

In order to (3.145) be satisfied for all δx_f and δt_f, the following equations has to be satisfied

$$\left[\frac{\partial \Phi}{\partial x}(x(t_f), t_f) - \lambda^*(t_f)^T\right] \delta x_f = 0 \tag{3.161a}$$

$$\left[H(x^*(t_f), \lambda^*(t), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f)\right] \delta t_f = 0 \tag{3.161b}$$

which means that $\left[\frac{\partial \Phi}{\partial x}(x(t_f), t_f) - \lambda^*(t_f)\right]$ is orthogonal to δx_f, and $\left[H(x^*(t_f), \lambda^*(t), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f)\right]$ is orthogonal to δt_f.

These conditions can only be satisfied if $\left[\frac{\partial \Phi}{\partial x}(x(t_f), t_f) - \lambda^*(t_f)^T\right]$ is parallel to $\left[\frac{\partial h_1}{\partial x}\right]$. Therefore, for some nonzero scalar η_1,

$$\left[\frac{\partial \Phi}{\partial x}(x(t_f), t_f) - \lambda^*(t_f)\right] = \eta_1 \left[\frac{\partial h_1}{\partial x}\right]. \tag{3.162}$$

With respect to the terms related to δt_f, the vector $\left[H(x^*(t_f), \lambda^*(t), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f)\right]$ has to be parallel to $\left[\frac{\partial h_1}{\partial t}\right]$, which gives

$$\left[H(x^*(t_f), \lambda^*(t), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f)\right] = \eta_1 \left[\frac{\partial h_1}{\partial t}\right] \tag{3.163}$$

with the same scalar η_1.

Consider now that $h : \mathbb{R}^{N_x+1} \to \mathbb{R}^K$ defines a $(N_x - K)$ manifold. Then the perturbation δx_f and δt_f have to be orthogonal to the normal of every hypersurface,

$$\left[\frac{\partial h_k}{\partial x}\right] \delta x_f = 0, \qquad k = 1, \ldots, K \tag{3.164a}$$

$$\left[\frac{\partial h_k}{\partial t}\right] \delta t_f = 0, \qquad k = 1, \ldots, K \tag{3.164b}$$

therefore, the term $\left[\frac{\partial \Phi}{\partial x}(x(t_f), t_f) - \lambda^*(t_f)\right]$ can be written by a linear combinations of the normals of all h,

$$\left[\frac{\partial \Phi}{\partial x}(x(t_f), t_f) - \lambda^*(t_f)^T\right] = \eta^T \frac{\partial h}{\partial x}, \tag{3.165}$$

for some K-dimensional nonzero vector η, giving N_x conditions. However with the introduction of the vector $\eta \in \mathbb{R}^K$, it is necessary K additional equations, those are obtained from the manifold equation

$$h(x(t_f), t_f) = 0. \tag{3.166}$$

The last equation is obtained with the terms related to the perturbation δt_f,

$$\left[H(x^*(t_f), \lambda^*(t), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f)\right] = \eta^T \left[\frac{\partial h_1}{\partial t}\right]. \tag{3.167}$$

3.2.3 More General Optimal Control Problem

Using the conditions defined in Sections 3.2.1 and 3.2.2, we are able to solve a problem that is more general than the optimal control problem $\mathcal{P}_s$ (3.87). By using those conditions it is possible to solve a problem in the form

$$\min_{u, t_f} \quad \Phi(x(t_f), t_f) + \int_{t_0}^{t_f} L(x, u, t)\, dt \tag{3.168a}$$

$$\text{s.t.:} \quad \dot{x} = f(x, u, t) \tag{3.168b}$$

$$x(t_0) = x_0 \tag{3.168c}$$

$$h(x(t_f), t_f) = 0 \tag{3.168d}$$

Notice that with the introduction of the constraint (3.168d) it is possible to solve some positioning problems and a vast number of problems with constraints in the final condition. However the problem (3.168) does still assume that the controls and states can assume any value, which is not the case in most of the applications. In real world applications, control and state variables are subject to constraint.

3.3 PONTRYAGIN'S MINIMUM PRINCIPLE[4]

This section will investigate the case in which the controls are piecewise continuous functions with a finite number of discontinuities and which should be kept inside a set U_B,

$$U_B = \left\{ \mathrm{u} \in U = \mathbb{R}^{N_u} \,|\, u_L \leq \mathrm{u} \leq u_U \right\} \tag{3.169}$$

where u_L is a lower bound for the control, and u_U is an upper bound. Let us call the controls that satisfy these requirements *admissible controls*.

The optimal control problem addressed here is similar to those addressed in Section 3.2, with the additional constraint that the control profiles lie in some set U_B. The problem to be addressed in this section is defined by

$$\begin{aligned}
\mathcal{P}_P: \quad & \min_{u,t_f} \quad \Phi(x(t_f),t_f) + \int_{t_0}^{t_f} L(x,u,t)\,dt && \text{(3.170a)}\\
& \text{s.t.:} \quad \dot{x} = f(x,u,t) && \text{(3.170b)}\\
& \qquad x(t_0) = x_0 && \text{(3.170c)}\\
& \qquad h(x(t_f),t_f) = 0 && \text{(3.170d)}\\
& \qquad u(t) \in U_B \qquad \forall t \in [t_0,t_f] && \text{(3.170e)}
\end{aligned}$$

Consider that we were solving problem $\mathcal{P}_P$ (3.170) with an unconstrained control ($u(t) \in U$) and the solution u^*_{unb} lies in the interior region of U_B. In this case, the optimality conditions of Theorem 6 and those developed in the Sections 3.2.1 and 3.2.2 are valid.

Intuitively it follows that if for all $t \in [t_0,t_f]$, the solution of $\mathcal{P}_P$ (with $u(t) \in U_B$) given by u^* is in the interior of U_B, then

[4]This section was written following [17], but with references to [18, 19]

conditions of Theorem 6 and the other conditions developed on Sections 3.2.1 and 3.2.2 shall hold, because solving with or without the bound constraints in the controls leads to the results.

The challenging case occurs when for at least one subinterval $[t_1, t_2] \subseteq [t_0, t_f]$ with $t_1 < t_2$ the optimal control u^* lies on the boundary of the set U_B, that is

$$u^*(t) = u_L \qquad \forall t \in [t_1, t_2] \tag{3.171}$$

or

$$u^*(t) = u_U \qquad \forall t \in [t_1, t_2]. \tag{3.172}$$

For this case, the condition for the optimal control u^* given in (3.120c),

$$\frac{\partial H}{\partial u}(x^*, \lambda^*, u^*, t) = 0 \qquad \forall t \in [t_0, t_f] \tag{3.173}$$

cannot be applied, since there is no guarantee that the control that satisfies (3.173) satisfies $u^*(t) \in U_B$. Furthermore, the developed idea of a perturbation δu cannot be applied since the control $u^*(t) - \delta u(t)$ might not be an admissible control, given that $u^*(t) - \delta u(t) \notin U_B$. Figure 3.8 illustrates a case in which the controls obtained from (3.173) do not lie inside U_B, and how the perturbation δu is an admissible perturbation, while $-\delta u$ is not. Notice that when the control u^* is saturated, the perturbation control $u^* - \delta u$ violates the constraint. For this case, a new condition has to be defined.

The following "demonstration" of Pontryagin's Minimum Principle follows a heuristic approach, being more intuitive than the mathematical demonstration. For a rigorous demonstration the reader is referred to Pontryagin's book [18].

The control profile u^* is a local minimum to the functional J if

$$\Delta J(u) = J(u) - J(u^*) \geq 0 \tag{3.174}$$

for all admissible control u close to u^*. By defining $u = u^* + \delta u$, the increment in J can be expressed by

$$\Delta J(u^*, \delta u) = \delta J(u^*, \delta u) + \text{higher-order variations.} \tag{3.175}$$

From (3.104) we have the first variation of the functional J_a, assuming that the dynamic equation holds we have $J_a = J$.

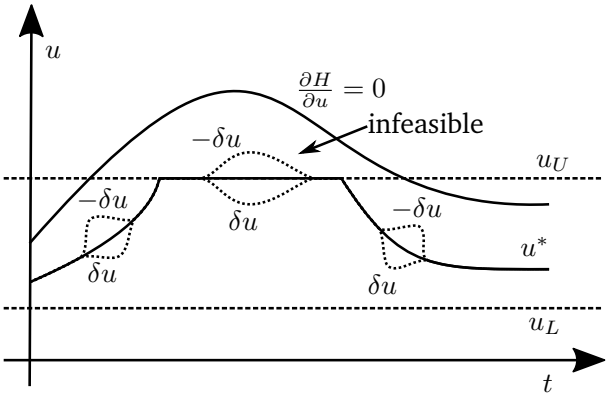

Figure 3.8: Illustration of a control profile defined by $\frac{\partial H}{\partial u} = 0$ and bounded control.

Therefore, in term of the Hamiltonian H, δJ is given by

$$\begin{aligned}\delta J = &\left[H(x^*(t_f), \lambda^*(t_f), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f)\right]\delta t_f \\ &+ \left[\frac{\partial \Phi}{\partial x}(x^*(t_f), t_f) - \lambda^*(t_f)^T\right]\delta x_f \\ &+ \int_{t_0}^{t_f} \left\{ \left[\frac{\partial H}{\partial x}(x^*, \lambda^*, u^*, t) + \dot{\lambda}^{*T}\right]\delta x \right. \\ &\left. + \left[\frac{\partial H}{\partial u}(x^*, \lambda^*, u^*, t)\right]\delta u + \left[\frac{\partial H}{\partial \lambda}(x^*, \lambda^*, u^*, t) - \dot{x}^{*T}\right]\delta\lambda \right\} dt.\end{aligned} \tag{3.176}$$

If the state equations are satisfied, and λ^* is chosen so the terms related to δx are zero to make the first variation be zero (Theorem 1), and the boundary conditions are satisfied, then (3.175) can be written as

$$\begin{aligned}\Delta J = \int_{t_0}^{t_f} &\left[\frac{\partial H}{\partial u}(x^*(t), \lambda^*(t), u^*(t), t)\right]\delta u \, dt \\ &+ \text{higher-order variations}\end{aligned} \tag{3.177}$$

by the definition of the first variation,

$$\left[\frac{\partial H}{\partial u}(x^*(t), \lambda^*(t), u^*(t), t)\right] \delta u = H(x^*(t), \lambda^*(t), u^*(t) + \delta u(t), t) - H(x^*(t), \lambda^*(t), u^*(t), t) \tag{3.178}$$

therefore,

$$\Delta J = \int_{t_0}^{t_f} \Big[H(x^*(t), \lambda^*(t), u^*(t) + \delta u(t), t) - H(x^*(t), \lambda^*(t), u^*(t), t)\Big] dt + \text{higher-order variations} \tag{3.179}$$

given that $u^* + \delta u$ is sufficiently small, in the neighborhood of u^*, the higher terms are small and the integral dominates the value of ΔJ. Thus, for u^* to be a solution it is necessary that

$$\int_{t_0}^{t_f} \left[H(x^*(t_f), u^*(t_f) + \delta u(t), t_f) - H(x^*(t_f), u^*(t_f), t_f)\right] dt \geq 0 \tag{3.180}$$

for all admissible δu given that $u^* + \delta u$ is close enough to u^*. In order for (3.180) to be true for every admissible δu satisfying the neighborhood demand, it is necessary that

$$H(x^*(t), \lambda^*(t), u^*(t), t) \leq H(x^*(t), \lambda^*(t), u^*(t) + \delta u(t), t) \tag{3.181}$$

for all $t \in [t_0, t_f]$ and for all admissible δu.

Notice that if u is piecewise continuous with a finite number of discontinuities, then the state x and the costate λ are piecewise differentiable. This is a consequence of the dynamic equations, for instance

$$\dot{x} = f(x, u, t) \tag{3.182}$$

where the right-hand side can be discontinuous where u is discontinuous, and by consequence the first derivative of x are discontinuous at the discontinuity points.

Using these ideas, the necessary optimality condition for constrained OCP can be stated. These conditions are known as the Pontryagin's Minimum Principle.

Theorem 7 (Pontryagin's Minimum Principle [18]). *Let $u^*(t)$ for $t \in [t_0, t_f]$ be an admissible control such that the corresponding trajectory x^* is defined by*

$$\dot{x}^* = \frac{\partial H}{\partial \lambda} = f(x^*, u^*, t) \tag{3.183}$$

with the boundary condition $x^*(t_0) = x_0$. In order for $u^*(t)$ and $x^*(t)$ to be optimal it is necessary that there exists a nonzero continuous vector function $\lambda^* : [t_0, t_f] \to \mathbb{R}^{N_x}$ corresponding to $u^*(t)$ and $x^*(t)$ through the ODE such that

$$-\dot{\lambda}^* = \frac{\partial H}{\partial x} = \frac{\partial L}{\partial x}(x^*, u^*, t) + \frac{\partial f}{\partial x}(x^*, u^*, t), \tag{3.184}$$

with the boundary conditions given by the conditions stated in Sections 3.2.1 and 3.2.2, accordingly to the problem characteristics, and that

1. *The function $H(x^*(t), \lambda^*(t), \mathrm{u}, t)$, with $\mathrm{u} \in U_B$, attains its minimum at the point $\mathrm{u} = u^*(t)$ for all $t \in [t_0, t_f]$:*

$$H(x^*(t), \lambda^*(t), u^*(t), t) = \inf_{\mathrm{u} \in U_B} H(x^*(t), \lambda^*(t), \mathrm{u}, t) \tag{3.185}$$

   or, equivalently, for all $\mathrm{u} \in U_B$ *and* for all $t \in [t_0, t_f]$*:*

$$H(x^*(t), \lambda^*(t), u^*(t), t) \leq H(x^*(t), \lambda^*(t), \mathrm{u}, t). \tag{3.186}$$

2. *If the final time is fixed and the Hamiltonian does not depend explicitly on the time variable, then the Hamiltonian must be equal to a constant c_1 for all $t \in [t_0, t_f]$,*

$$H(x^*(t), \lambda^*(t), u^*(t), t) = c_1, \quad \forall t \in [t_0, t_f]. \tag{3.187}$$

3. *If the final time is free and the Hamiltonian does not depend explicitly on the time variable, then the Hamiltonian must be zero for all $t \in [t_0, t_f]$,*

$$H(x^*(t), \lambda^*(t), u^*(t), t) = 0, \quad \forall t \in [t_0, t_f]. \tag{3.188}$$

Proof. The proof of this Theorem can be found in [18, Chapter 2]. □

The first premise of Pontryagin's minimum principle has already been intuitively developed. The second and third statements can be verified in [19] using the Hamilton-Bellman-Jacobi equation (which will be presented in the next section), or in [18, Chapter 2] which has a more mathematical flavor. For the particular case in which the controls lie inside the interior of U_B for all $t \in [t_0, t_f]$ and if time does not appear in the Hamiltonian explicitly, then the control u^* satisfies $\frac{\partial H}{\partial u} = 0$, and the time derivative of the Hamiltonian can be obtained using the chain rule,

$$\begin{aligned}\frac{dH}{dt} &= \frac{\partial H}{\partial x}\dot{x}^* + \frac{\partial H}{\partial \lambda}\dot{\lambda}^* + \frac{\partial H}{\partial u}\frac{du^*}{dt} + \frac{\partial H}{\partial t} \\ &= \left[-\dot{\lambda}\right]\dot{x} + [\dot{x}]\,\dot{\lambda} + 0 \times \frac{du^*}{dt} + 0 = 0 \end{aligned} \tag{3.189}$$

Notice that if the optimum control u^* lies in the interior of U_B for all t, then the minimization of u^* in (3.185) 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 $\frac{\partial H}{\partial u} = 0$ 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 $u_{opt}(t)$ that lies in the interior of U_B, then $u^*(t) = u_{opt}(t)$ at time t. Otherwise, if $u_{opt}(t)$ 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 [17]. For instance, if $u_{opt}(t) \in \mathbb{R}$ is above u_U at some time $t = t_1$, then $u^*(t_1) = u_U$. Therefore, given that u_{opt} is the solution to $\frac{\partial H}{\partial u} = 0$, (3.185) can be written for a convex Hamiltonian as:

$$u^*(t) = \begin{cases} u_U, & \text{if } u_U \leq \hat{u}, \\ \hat{u} & \text{if } u_L < \hat{u} < u_U, \\ u_L, & \text{if } \hat{u} \leq u_L \end{cases} \tag{3.190}$$

From a practical standpoint, when the convexity of the Hamiltonian cannot ascertained, then the procedure (3.190) can be seen as a best effort strategy.

In the following section, the Hamilton-Jacobi-Bellman (HJB) equation will be presented. The HJB equations give a necessary and sufficient condition of optimality for an OCP.

3.4 HAMILTON-JACOBI-BELLMAN EQUATION[5]

The conditions presented so far are *necessary* for optimality. However stronger conditions are lacking. In this section, sufficient and necessary conditions are presented in the form of the Hamilton-Jacobi-Bellman equation (HJB equation). These conditions are in fact stronger than the necessary conditions presented previously, however in practice they are not applicable in a general form giving only a theoretical support for the field of optimal control.

For convenience, let us define an OCP of ODE with bounded controls that will be used for the developments in this section,

$$\mathcal{P}_{HJB}: \quad \min_{u} \quad J(x,u) = \Phi(x(t_f),t_f) + \int_{t_0}^{t_f} L(x,u,t)\,dt \tag{3.191a}$$

$$\text{s.t.:} \quad \dot{x} = f(x,u,t) \tag{3.191b}$$

$$x(t_0) = x_0 \tag{3.191c}$$

$$u(t) \in U_B \tag{3.191d}$$

$$t \in [t_0, t_f] \tag{3.191e}$$

where the control profile u is a piecewise continuous function such that

$$U_B = \left\{ \mathrm{u} \in U = \mathbb{R}^{N_u} \,|\, u_L \leq \mathrm{u} \leq u_U \right\}, \tag{3.192}$$

and that t_0 and t_f are fixed.

3.4.1 Optimality Principle

Before stating the HJB equation, an important concept has to be introduced. Let us call the following theorem the optimality principle.

Theorem 8 (Optimality Principle [**?**, 17])**.** *Let u^* be an optimal control for $\mathcal{P}_{HJB}$* (3.191)*, which induces a state profile x^* given by the system of equations* (3.191b) *and the initial conditions* (3.191c)*.*

Given a $t_1 \in [t_0, t_f)$, then the control $u^*(t)$ with $t \in [t_1, t_f]$ is a minimizer of the functional

$$J_{[t_1,t_f]}(x,u) = \Phi(x(t_f),t_f) + \int_{t_1}^{t_f} L(x,u,t)\,dt \tag{3.193}$$

[5]This section was written based on [17, 19, **?**]

subject to

$$\dot{x} = f(x, u, t) \tag{3.194a}$$
$$x(t_1) = x^*(t_1) \tag{3.194b}$$

Moreover,

$$\min_u \mathcal{P}_{HJB} = \min_{\substack{\dot{x}=f(x,u,t) \\ u\in U_B, x(t_0)=x_0}} \left\{ \int_{t_0}^{t_1} L(x, u, t)\, dt + J_{[t_1,t_f]}(x, u) \right\} \tag{3.195}$$

Proof. Assume that $\widehat{u} \in U_B$ satisfies

$$J_{[t_1,t_f]}(\widehat{x}, \widehat{u}) < J_{[t_1,t_f]}(x^*, u^*) \tag{3.196}$$

where $\widehat{x}$ is the solution of (3.191b) with initial conditions (3.191c) and controls given by

$$u(t) = \begin{cases} u^*(t) & t \in [t_0, t_1) \\ \widehat{u}(t) & t \in [t_1, t_f] \end{cases} \tag{3.197}$$

Since the dynamic system has an unique solution,

$$\widehat{x}(t) = x^*(t) \qquad t \in [t_0, t_1) \tag{3.198}$$

Using a fundamental integral property, we have

$$J(x, u) = \Phi(x(t_f), t_f) + \int_{t_0}^{t_f} L(x, u, t)\, dt \tag{3.199a}$$
$$= \Phi(x(t_f), t_f) + \int_{t_0}^{t_1} L(x, u, t)\, dt + \int_{t_1}^{t_f} L(x, u, t)\, dt \tag{3.199b}$$

and applying for $\widehat{x}$ and $\widehat{u}$

$$J(\widehat{x},\widehat{u}) = \Phi(\widehat{x}(t_f),t_f) + \int_{t_0}^{t_1} L(\widehat{x},\widehat{u},t)\,dt + \int_{t_1}^{t_f} L(\widehat{x},\widehat{u},t)\,dt \tag{3.200a}$$

$$= \Phi(\widehat{x}(t_f),t_f) + \int_{t_0}^{t_1} L(x^*,u^*,t)\,dt + \int_{t_1}^{t_f} L(\widehat{x},\widehat{u},t)\,dt \tag{3.200b}$$

$$= \int_{t_0}^{t_1} L(x^*,u^*,t)\,dt + J_{[t_1,t_f]}(\widehat{x},\widehat{u}) \tag{3.200c}$$

$$< \int_{t_0}^{t_1} L(x^*,u^*,t)\,dt + J_{[t_1,t_f]}(x^*,u^*) \tag{3.200d}$$

$$= \Phi(x^*(t_f),t_f) + \int_{t_0}^{t_1} L(x^*,u^*,t)\,dt + \int_{t_1}^{t_f} L(x^*,u^*,t)\,dt \tag{3.200e}$$

$$= J(x^*,u^*) \tag{3.200f}$$

which contradicts the optimality of x^* and u^*. □

The idea developed in Theorem 8 shares similarities with Dynamic Programming. The theorem shows that if you slice the interval at some time t_1, the control profile $u^*(t)$ with $t \in [t_1,t_f]$ is optimal in sense of taking the system from $x(t_1)$ and moving it to some state $x(t_f)$ in order to minimize the functional $J_{[t_1,t_f]}$, which is the functional J sliced at t_1. In other words, it has been shown that

$$\min_{\substack{\dot{x}=f(x,u,t)\\ u\in U_B, x(t_1)=x^*(t_1)}} J_{[t_1,t_f]}(x,u) = J_{[t_1,t_f]}\left(x^*|_{t\in[t_1,t_f]}\,,\,u^*|_{t\in[t_1,t_f]}\right) \tag{3.201}$$

which allows us to conclude that if you are on an optimal trajectory, the best you can do is to stay on the trajectory.

3.4.2 HJB Equation

In order to obtain the HJB equation, let us define a *value* function V, which is the cost induced by a control profile u in a time interval from an initial time $t \in [t_0,t_f]$ to the final time t_f,

where the initial condition is $x(t) \in \mathbb{R}^{N_x}$,

$$V(x(t), t, u) = \Phi(x(t_f), t_f) + \int_t^{t_f} L(x(\tau), u(\tau), \tau)\, d\tau \tag{3.202}$$

where state x meets the system dynamics for all $\tau \in [t, t_f]$, and u is an admissible control profile for all $\tau \in [t, t_f]$.

The control profile that minimizes (3.202) is unknown, but for all possible $x(t)$ and t the optimal cost can be represented by

$$V^*(x(t), t) = \min_{u(t) \in U_B} \left\{ \Phi(x(t_f), t_f) + \int_t^{t_f} L(x(\tau), u(\tau), \tau)\, d\tau \right\} \tag{3.203}$$

Using a fundamental integral property, the integration can be decomposed into

$$V^*(x(t), t) = \min_u \left\{ \Phi(x(t_f), t_f) + \int_t^{t+\varepsilon} L(x(\tau), u(\tau), \tau)\, d\tau + \int_{t+\varepsilon}^{t_f} L(x(\tau), u(\tau), \tau)\, d\tau \right\} \tag{3.204}$$

for some $\varepsilon < t_f - t$.

Using the optimality principle, (3.204) can be rewritten as

$$V^*(x(t), t) = \min_u \left\{ \int_t^{t+\varepsilon} L(x(\tau), u(\tau), \tau)\, d\tau + V^*(x(t+\varepsilon), t+\varepsilon) \right\} \tag{3.205}$$

where $V^*(x(t+\varepsilon), t+\varepsilon)$ is the minimum cost for the time $t+\varepsilon \leq \tau \leq t_f$ with the initial condition $x(t+\varepsilon)$.

Notice that V^* is a function not a functional, if we assume that V^* has second partial derivatives and those are bounded, we can expand $V^*(x(t+\varepsilon), t+\varepsilon)$ at the point $(x(t), t)$ which results in

$$V^*(x(t), t) = \min_u \left\{ \int_t^{t+\varepsilon} L(x(\tau), u(\tau), \tau)\, d\tau + V^*(x(t), t) + V_t^*(x(t), t)\varepsilon + [V_x^*(x(t), t)]\,[x(t+\varepsilon) - x(t)] + \text{ term of higher order} \right\} \tag{3.206}$$

where V_t^* and V_x^* are shorthands for the partial derivatives,

$$V_t^*(x(t),t) = \frac{\partial V^*}{\partial t}(x(t),t) \tag{3.207a}$$

$$V_x^*(x(t),t) = \frac{\partial V^*}{\partial x}(x(t),t) \tag{3.207b}$$

For a small ε, equation (3.206) can be rewritten as

$$V^*(x(t),t) = \min_{u(t)\in U_B} \Big\{ L(x(t),u(t),t)\varepsilon + V^*(x(t),t) + V_t^*(x(t),t)\varepsilon + [V_x^*(x(t),t)]\, f(x(t),u(t),t)\varepsilon + o(\varepsilon) \Big\} \tag{3.208}$$

where $o(\varepsilon)$ denotes the terms with ε^2 and the terms from the approximation of the integral and the truncation of the Taylor expansion. Eliminating the $V^*(x(t),t)$ terms and removing V_t^* from the minimization, since it does not depend on u, we obtain

$$0 = V_t^*(x(t),t)\varepsilon + \min_{u(t)\in U_B} \Big\{ L(x(t),u(t),t)\varepsilon + [V_x^*(x(t),t)]\, f(x(t),u(t),t)\varepsilon + o(\varepsilon) \Big\} \tag{3.209}$$

Dividing (3.209) by ε and taking the limit $\varepsilon \to 0$, we obtain the Hamilton-Jacobi-Bellman equation

$$-V_t^*(x(t),t) = \min_{u(t)\in U_B} \Big\{ L(x(t),u(t),t) + [V_x^*(x(t),t)]\, f(x(t),u(t),t) \Big\} \tag{3.210}$$

with the truncation error vanishing since $\frac{o(\varepsilon)}{\varepsilon} \to 0$ as $\varepsilon \to 0$.

The boundary condition to this partial differential equation (PDE) can be obtained by setting $t = t_f$, from (3.203) we obtain

$$V^*(x(t_f),t_f) = \Phi(x(t_f),t_f) \tag{3.211}$$

In the sequence, a theorem is developed that shows a necessary and *sufficient* condition for optimality of a control profile.

Theorem 9 (Sufficient Optimality Conditions [19, 21])**.** *Let V^* be a function of $x(t)$ and t that satisfies the HJB equation*

$$-V_t^*(x(t),t) = \min_{u(t)\in U_B} \Big\{ L(x(t),u(t),t) + [V_x^*(x(t),t)]\, f(x(t),u(t),t) \Big\} \tag{3.212a}$$

$$V^*(x(t_f),t_f) = \Phi(x(t_f),t_f) \tag{3.212b}$$

and the functions V, f, L, and Φ be continuously differentiable with respect to their arguments. Suppose that the piecewise continuous function $\mu^*(x(t),t) \in U_B$ minimizes (3.212a) *for all possible $x(t)$ and t. Let x^* be the state profile when the applied controls are $u^*(t) = \mu^*(x^*(t),t)$ for all $t \in [t_0,t_f]$, being defined through the ODE*

$$\dot{x}^* = f(x^*,u^*,t) \tag{3.213}$$

with the initial conditions $x^*(t_0) = x_0$.

Then $V^*(x_0,t_0)$ is equal to minimum of (3.191)*, and furthermore u^* is optimal.*

Proof. Let $\widehat{u} \in U_B$ be any admissible control trajectory, and $\widehat{x}$ be its corresponding state profile with an initial condition $\widehat{x}(t_0) = x_0$.

From (3.212a) we have

$$0 \le L(\widehat{x}(t),\widehat{u}(t),t) + V_t^*(\widehat{x}(t),t) + [V_x^*(\widehat{x}(t),t)]\, f(\widehat{x}(t),\widehat{u}(t),t) \tag{3.214}$$

which the right-hand side can be substituted with

$$L(\widehat{x}(t),\widehat{u}(t),t) + \frac{dV^*}{dt}(\widehat{x}(t),t) \tag{3.215}$$

and be integrated over the time interval $[t_0,t_f]$, resulting in

$$0 \le \int_{t_0}^{t_f} L(\widehat{x}(t),\widehat{u}(t),t)\,dt + V^*(\widehat{x}(t_f),t_f) - V^*(\widehat{x}(t_0),t_0) \tag{3.216}$$

Using the initial condition $\widehat{x}(t_0) = x_0$ and the boundary condition $V^*(\widehat{x}(t_f),t_f) = \Phi(\widehat{x}(t_f),t_f)$, we have

$$V^*(x_0,t_0) \le \Phi(\widehat{x}(t_f),t_f) + \int_{t_0}^{t_f} L(\widehat{x}(t),\widehat{u}(t),t)\,dt \tag{3.217}$$

If x^* and u^* in the place of $\widehat{x}$ and $\widehat{u}$ in (3.214), then the inequality becomes the equality

$$0 = L(x^*(t), u^*(t), t) + V_t^*(x^*(t), t) + [V_x^*(x^*(t), t)]\, f(x^*(t), u^*(t), t) \quad (3.218)$$

and the subsequential inequalities become equalities that leads to

$$V^*(x_0, t_0) = \Phi(x^*(t_f), t_f) + \int_{t_0}^{t_f} L(x^*(t), u^*(t), t)\, dt \quad (3.219)$$

Therefore the cost corresponding to u^* is $V^*(x_0, t_0)$, shown in (3.219), and is the minimum between all admissible controls, shown in (3.217). □

Remark ([21, 17])**.** *Regarding the Proof of Theorem 3.212, the objective functional of $\widehat{x}$ and $\widehat{u}$ is*

$$J(\widehat{x}, \widehat{u}) = \Phi(\widehat{x}(t_f), t_f) + \int_{t_0}^{t_f} L(\widehat{x}(t), \widehat{u}(t), t)\, dt \quad (3.220)$$

Using the definition of the integral, we have

$$V^*(\widehat{x}(t_0), t_0) - V^*(\widehat{x}(t_f), t_f) + \int_{t_0}^{t_f} \frac{dV^*}{dt}(\widehat{x}(t), t)\, dt = 0 \quad (3.221)$$

assuming that it holds for all $u(t) \in U_B$ for all t.

Adding up both equation,

$$J(\widehat{x}, \widehat{u}) = V^*(x_0, t_0) + \Phi(\widehat{x}(t_f), t_f) - V^*(\widehat{x}(t_f), t_f) + \int_{t_0}^{t_f} L(\widehat{x}(t), \widehat{u}(t), t) + \frac{dV^*}{dt}(\widehat{x}(t), t)\, dt \quad (3.222)$$

Using the boundary condition and substituting the total derivative with the partial derivative,

$$J(\widehat{x}, \widehat{u}) = V^*(x_0, t_0) + \int_{t_0}^{t_f} \Big\{ L(\widehat{x}(t), \widehat{u}(t), t) + V_t^*(x(t), t) + [V_x^*(x(t), t)]\, f(x(t), u(t), t) \Big\}\, dt \quad (3.223)$$

Since V^* satisfy (3.210) *where $\mu(\widehat{x},t)$ is the minimum, we can replace $V^*(\widehat{x}(t),t)$ in the previous equation to obtain*

$$J(\widehat{x},\widehat{u}) = V^*(x_0,t_0) + \int_{t_0}^{t_f} \Big\{ L(\widehat{x}(t),\widehat{u}(t),t) - L(\widehat{x}(t),\mu(\widehat{x}(t),t),t) + \left[V_x^*(\widehat{x}(t),t)\right] f(\widehat{x}(t),\widehat{u}(t),t) - \left[V_x^*(\widehat{x}(t),t)\right] f(\widehat{x}(t),\mu(\widehat{x}(t),t),t) \Big\}\, dt \quad (3.224)$$

Let us define a Hamiltonian function

$$H_{HJB}\big(x(t),V_x(x(t),t),u(t),t\big) = L(x(t),u(t),t) + \left[V_x^*(x(t),t)\right] f(x(t),u(t),t) \quad (3.225)$$

then (3.224) *can be rewritten as*

$$J(\widehat{x},\widehat{u}) = V^*(x_0,t_0) + \int_{t_0}^{t_f} \Big\{ H_{HJB}\big(\widehat{x}(t),V_x^*(\widehat{x}(t),t),\widehat{u}(t),t\big) - H_{HJB}\big(\widehat{x}(t),V_x^*(\widehat{x}(t),t),\mu(\widehat{x}(t),t),t\big) \Big\}\, dt \quad (3.226)$$

Notice that the integral in (3.226) *is taken along a nonoptimal $\widehat{x}$ path induced by $\widehat{u}$. Equation* (3.226) *can be rearranged to obtain*

$$\Delta J = J(\widehat{x},\widehat{u}) - V^*(x_0,t_0) \quad (3.227a)$$

$$= \int_{t_0}^{t_f} \Big\{ H_{HJB}\big(\widehat{x}(t),V_x^*(\widehat{x}(t),t),\widehat{u}(t),t\big) - H_{HJB}\big(\widehat{x}(t),V_x^*(\widehat{x}(t),t),\mu(\widehat{x}(t),t),t\big) \Big\}\, dt \geq 0 \quad (3.227b)$$

which represents the change in cost away from the optimal path. Notice that if $\widehat{u} = u^*$ and $\widehat{x} = x^*$, then the integral vanishes and $\Delta J = 0$. For any nonoptimal $\widehat{u}$, the value of ΔJ will be positive.

3.5 OPTIMALITY CONDITIONS FOR DAE SYSTEMS

The theory described so far handles only optimal control problems of ODE systems. In this section, the developments of Sections 3.2 and 3.3 are extended to handle DAE system.

Let us define a standard form for an OPC of a DAE system, namely $\mathcal{P}_{DAE}$, which is given by

$$\begin{aligned}
\mathcal{P}_{DAE}: \quad \min \;& \Phi(x(t_f), t_f) + \int_{t_0}^{t_f} L(x, y, u, t)\, dt && \text{(3.228a)}\\
\text{s.t.:} \;& \dot{x} = f(x, y, u, t) && \text{(3.228b)}\\
& g(x, y, u, t) = 0 && \text{(3.228c)}\\
& h(x(t_f), t_f) = 0 && \text{(3.228d)}\\
& x(0) = x_0 && \text{(3.228e)}\\
& t \in [t_0, t_f] && \text{(3.228f)}
\end{aligned}$$

where $x(t) \in \mathbb{R}^{N_x}$ is the state vector, $y(t) \in \mathbb{R}^{N_y}$ is the algebraic vector, and $u \in \mathbb{R}^{N_u}$ is the control vector. The function f describes the system dynamics, g characterizes the algebraic variables, and h is the final time constraint. The final time cost is given by Φ, and L is the integral cost. Let us assume that (3.228b) and (3.228c) form a semi-explicit DAE system, and that f, g, and h are continuously differentiable with respect to their arguments.

For developing the optimality conditions for the OCP, the adjoint variable $\lambda : [t_0, t_f] \to \mathbb{R}^{N_x}$ was introduced. In addition to λ, for OCP of DAE systems a new multiplier has to be introduced for the algebraic equations. Let $\nu : [t_0, t_f] \to \mathbb{R}^{N_y}$ be a multiplier of (3.228c).

The multiplier $\nu(t)$ shares a meaning similar to the multiplier η introduced in Section (3.2.1) for the final time constraint (final state lying in a manifold). The algebraic equation (3.228c) can be understood as a manner to express that the algebraic variable y belongs to a manifold, however, differently from the final state constraint (3.228d), the algebraic variable has to belong to the manifold for all $t \in [t_0, t_f]$ and, therefore, ν is a time dependent variable.

In order to restate Theorem 5 with the inclusion of the algebraic equation, let us define the Hamiltonian for OCP of DAE systems,

$$\begin{aligned}
H_{DAE}(x, \lambda, y, \nu, u, t) = L(x, y, u, t) + \lambda^T f(x, y, u, t) \\
+ \nu^T g(x, y, u, t) \quad \text{(3.229)}
\end{aligned}$$

Which allows us to state necessary conditions for optimality of OCPs of DAE systems.

Theorem 10 ([11]). *Consider an OCP of DAE in the standard form $\mathcal{P}_{DAE}$. If the control u^*, which induces the states x^* by* (3.228b) *and the algebraic variables y^* by* (3.228c) *in the time interval $t \in [t_0, t_f]$, is a minimum of $\mathcal{P}_{DAE}$, then there exist a continuous differentiable function $\lambda^* : [t_0, t_f] \to \mathbb{R}^{N_x}$ and a function $\nu^* : [t_0, t_f] \to \mathbb{R}^{N_y}$, such that*

$$\frac{\partial H_{DAE}}{\partial x} = -\dot{\lambda}^{*T} = \frac{\partial L}{\partial x}(x^*, y^*, u^*, t) + \lambda^{*T}\frac{\partial f}{\partial x}(x^*, y^*, u^*, t) + \nu^{*T}\frac{\partial g}{\partial x}(x^*, y^*, u^*, t) \quad (3.230a)$$

$$\frac{\partial H_{DAE}}{\partial y} = 0 = \frac{\partial L}{\partial y}(x^*, y^*, u^*, t) + \lambda^{*T}\frac{\partial f}{\partial y}(x^*, y^*, u^*, t) + \nu^{*T}\frac{\partial g}{\partial y}(x^*, y^*, u^*, t) \quad (3.230b)$$

$$\frac{\partial H_{DAE}}{\partial u} = 0 = \frac{\partial L}{\partial u}(x^*, y^*, u^*, t) + \lambda^{*T}\frac{\partial f}{\partial u}(x^*, y^*, u^*, t) + \nu^{*T}\frac{\partial g}{\partial u}(x^*, y^*, u^*, t) \quad (3.230c)$$

Proof. The proof follows the same steps of the proof of Theorem 5, therefore a shortened proof is presented here.

Let us define the an augmented functional J_a, given by

$$J_a(x, \lambda, y, \nu, u) = \Phi(x(t_f), t_f) + \int_{t_0}^{t_f} \Big\{ L(x, y, u, t) + \lambda^T \left[f(x, y, u, t) - \dot{x} \right] + \nu^T g(x, y, u, t) \Big\} dt \quad (3.231)$$

Let δx be the perturbation on the state, δy be the perturbation on the algebraic variable, and δu be the perturbation on the control variable. Then we can perform the same process used to obtain the first variation of δJ_a for the ODE case (3.104). By doing so, the first

variation of J_a at $(x^*, \lambda^*, y^*, \nu^*, u^*)$ is given by

$$\begin{aligned}
\delta J_a = &\Bigg[L(x^*(t_f), y^*(t_f), u^*(t_f), t_f) + \lambda^*(t_f)^T f(x^*(t_f), u^*(t_f), t_f) \\
&+ \nu^*(t_f)^T g(x^*(t_f), y^*(t_f), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f)\Bigg]\delta t_f \\
&+ \left[\frac{\partial \Phi}{\partial x}(x^*(t_f), t_f) - \lambda^*(t_f)^T\right]\delta x_f \\
&+ \int_{t_0}^{t_f} \Bigg\{ \left[\frac{\partial L}{\partial x} + \lambda^{*T}\frac{\partial f}{\partial x} + \nu^{*T}\frac{\partial g}{\partial x} + \dot{\lambda}^{*T}\right]\delta x \\
&+ \left[\frac{\partial L}{\partial y} + \lambda^{*T}\frac{\partial f}{\partial y} + \nu^{*T}\frac{\partial g}{\partial y}\right]\delta y \\
&+ \left[\frac{\partial L}{\partial u} + \lambda^{*T}\frac{\partial f}{\partial u} + \nu^{*T}\frac{\partial g}{\partial u}\right]\delta u \\
&+ \left[f(x^*, y^*, u^*, t) - \dot{x}^*\right]^T \delta\lambda + \left[g(x^*, y^*, u^*, t)\right]^T \delta\nu \Bigg\} dt \qquad (3.232)
\end{aligned}$$

where the arguments of partial derivatives were suppressed for a better readability.

If $(x^*, \lambda^*, y^*, \nu^*, u^*)$ minimizes $\mathcal{P}_{DAE}$, then by Theorem 1 the first variation has to be zero. By the fundamental lemma of the calculus of variations (Lemma 1) to J_a to be zero, the following system of equation has to be satisfied for all $t \in [t_0, t_f]$

$$-\dot{\lambda}^{*T} = \frac{\partial L}{\partial x} + \lambda^{*T}\frac{\partial f}{\partial x} + \nu^{*T}\frac{\partial g}{\partial x} \qquad (3.233a)$$

$$0 = \frac{\partial L}{\partial y} + \lambda^{*T}\frac{\partial f}{\partial y} + \nu^{*T}\frac{\partial g}{\partial y} \qquad (3.233b)$$

$$0 = \frac{\partial L}{\partial u} + \lambda^{*T}\frac{\partial f}{\partial u} + \nu^{*T}\frac{\partial g}{\partial u} \qquad (3.233c)$$

$$\dot{x}^* = f(x^*, y^*, u^*, t) \qquad (3.233d)$$

$$g(x^*, y^*, u^*, t) = 0 \qquad (3.233e)$$

□

Since we did not assume that the final time (t_f) and the final state (x_f) are free or fixed, the conditions have to be developed. Those conditions follow the same form of the conditions developed in Sections 3.2.1 and 3.2.2 for the OCP of ODE. In this work, all

these conditions will not be developed. It is attained for the particular case in which the final state is free and the final time is fixed, which represents the majority of cases of system control.

If the conditions of Theorem 10 are satisfied, then the perturbation of the augmented function (3.231) becomes

$$\delta J_a = \Big[L(x^*(t_f), y^*(t_f), u^*(t_f), t_f) + \lambda^*(t_f)^T f(x^*(t_f), u^*(t_f), t_f) + \nu^*(t_f)^T g(x^*(t_f), y^*(t_f), u^*(t_f), t_f) + \frac{\partial \Phi}{\partial t_f}(x^*(t_f), t_f)\Big]\delta t_f + \left[\frac{\partial \Phi}{\partial x}(x^*(t_f), t_f) - \lambda^*(t_f)^T\right]\delta x_f \quad (3.234)$$

If the final time is fixed then the perturbation δt_f is zero. Assuming that the final state is free, then the perturbation δx_f can take any value. In order to make the first variation zero, the following equation has to be satisfied,

$$\lambda^*(t_f) = \frac{\partial \Phi}{\partial x}(x^*(t_f), t_f)^T \quad (3.235)$$

Given an OCP of the form (3.228), with the addition restriction that the control function u has to be in the set U_B for all $t \in [t_0, t_f]$, where

$$U_B = \{\mathrm{u} \in U \,|\, u_L \leq \mathrm{u} \leq u_U\} \quad (3.236)$$

The optimality condition for this problem can be obtained by adapting the Pontryagin's minimum principle, Theorem 7.

Using the Hamiltonian given by (3.229), the Pontryagin's mininum principle can be recast in the following theorem.

Theorem 11. *Let the optimal control $u^*(t)$ satisfy $u^*(t) \in U_B$ for all $t \in [t_0, t_f]$, such that the corresponding trajectory x^* and y^* are defined by*

$$\frac{\partial H_{DAE}}{\partial \lambda}^T = \dot{x}^* = f(x^*, y^*, u^*, t) \quad (3.237a)$$

$$\frac{\partial H_{DAE}}{\partial \nu}^T = g(x^*, y^*, u^*, t) = 0 \quad (3.237b)$$

with the boundary condition $x^*(t_0) = x_0$.

In order for $x^*(t)$, $y^*(t)$, and $u^*(t)$ to induce an optimal solution to $\mathcal{P}_{DAE}$, there must exist a nonzero continuous vector function $\lambda^* : [t_0, t_f] \to \mathbb{R}^{N_x}$ corresponding to $x^*(t)$, $y^*(t)$, and $u^*(t)$ through the DAE

$$\frac{\partial H_{DAE}}{\partial x} = -\dot{\lambda}^{*T} = \frac{\partial L}{\partial x}(x^*, y^*, u^*, t) + \lambda^{*T}\frac{\partial f}{\partial x}(x^*, y^*, u^*, t) + \nu^{*T}\frac{\partial g}{\partial x}(x^*, y^*, u^*, t) \quad (3.238a)$$

$$\frac{\partial H_{DAE}}{\partial y} = \frac{\partial L}{\partial y}(x^*, y^*, u^*, t) + \lambda^{*T}\frac{\partial f}{\partial y}(x^*, y^*, u^*, t) + \nu^{*T}\frac{\partial g}{\partial y}(x^*, y^*, u^*, t) = 0 \quad (3.238b)$$

with the boundary conditions given by the conditions stated in Section 3.2.1 and 3.2.2 accordingly, and that

1. *The function $H(x^*(t), \lambda^*(t), y^*(t), \nu^*(t), \mathrm{u}, t)$ with $\mathrm{u} \in U_B$ attains its minimum at the point $\mathrm{u} = u^*(t)$ for all $t \in [t_0, t_f]$:*

$$H_{DAE}(x^*(t), \lambda^*(t), y^*(t), \nu^*(t), u^*(t), t) = \inf_{\mathrm{u} \in U_B} H(x^*(t), \lambda^*(t), y^*(t), \nu^*(t), \mathrm{u}, t) \quad (3.239)$$

   or, equivalently, for all $\mathrm{u} \in U_B$ and for all $t \in [t_0, t_f]$:

$$H(x^*(t), \lambda^*(t), y^*(t), \nu^*(t), u^*(t), t) \leq H(x^*(t), \lambda^*(t), y^*(t), \nu^*(t), \mathrm{u}, t) \quad (3.240)$$

2. *If the final is fixed and the Hamiltonian does not depend explicitly on the time variable, then the Hamiltonian must be equal to a constant c_1 for all $t \in [t_0, t_f]$,*

$$H(x^*(t), \lambda^*(t), y^*(t), \nu^*(t), u^*(t), t) = c_1 \quad \forall t \in [t_0, t_f] \quad (3.241)$$

3. *If the final time is free and the Hamiltonian does not depend explicitly on the time variable, then the Hamiltonian must be zero for all $t \in [t_0, t_f]$,*

$$H(x^*(t), \lambda^*(t), y^*(t), \nu^*(t), u^*(t), t) = 0 \quad \forall t \in [t_0, t_f] \quad (3.242)$$

Theorems 10 and 11 show a relation between the necessary conditions for optimality of OCP of ODE and the condition of OCP of DAE.

In the following section, the conditions developed so far will be used to obtain the controls that minimize an optimal control problem.

3.6 INDIRECT METHODS

The methods that obtain an optimal trajectory by solving the boundary value problem (BVP) that rises from the necessary optimal conditions are known as the indirect methods. On the other hand, the methods that obtain an optimal trajectory by applying gradient descent on the objective are known as the direct methods.

Each of these classes can be split in implicit and explicit methods. Implicit methods make use of black-box tools to solve the underlying ODE/DAE systems, among which the most representative method is the multiple shooting. In contrast, the explicit methods express the solution of the ODE/DAE systems as a set of nonlinear equations, for instance using the collocation method.

Herein, the indirect methods will be discussed and followed by an example. The various classes of OCPs are applied to the Van der Pol oscillator, which is described in the sequence.

3.6.1 Van der Pol Oscillator

The Van der Pol oscillator [20] is a dynamic system that has an unstable equilibrium at 0 and an attractive limit cycle. For this reason, the oscillator is a common benchmark for nonlinear control. The Van der Pol oscillator can be modeled in the form of an ODE system

$$\dot{x}_1 = (1 - x_2^2)x_1 - x_2 + u \tag{3.243a}$$
$$\dot{x}_2 = x_1 \tag{3.243b}$$

For the purpose of demonstration, the same system can be modeled by a DAE system

$$\dot{x}_1 = y + u \tag{3.244a}$$
$$\dot{x}_2 = x_1 \tag{3.244b}$$
$$y = (1 - x_2^2)x_1 - x_2 \tag{3.244c}$$

Considering the objective of driving the system to the unstable equilibrium, the objective functional is defined by

$$J = \int_{t_0}^{t_f} \left[x_1^2 + x_2^2 + u^2\right] \, dt. \tag{3.245}$$

The initial time is $t_0 = 0$, the final time is $t_f = 5$ seconds, and the initial conditions are $x(0) = [0, 1]^T$.

Using the objective (3.245) and the ODE system (3.243), let us define the OCP $\mathcal{P}^V_{ODE}$, given by

$$\begin{aligned}
\mathcal{P}^V_{ODE}: \quad \min_{x,y,u} \; J &= \int_{t_0}^{t_f} \left[x_1^2 + x_2^2 + u^2\right] \, dt && \text{(3.246a)}\\
\text{s.t.:} \quad \dot{x}_1 &= (1 - x_2^2)x_1 - x_2 + u && \text{(3.246b)}\\
\dot{x}_2 &= x_1 && \text{(3.246c)}\\
x(0) &= x_0, \qquad t \in [t_0, t_f] && \text{(3.246d)}
\end{aligned}$$

Let us define the OCP $\mathcal{P}^V_{DAE}$ using the objective (3.245) and the DAE system (3.244), which can be expressed by

$$\begin{aligned}
\mathcal{P}^V_{DAE}: \quad \min_{x,y,u} \; J &= \int_{t_0}^{t_f} \left[x_1^2 + x_2^2 + u^2\right] \, dt && \text{(3.247a)}\\
\text{s.t.:} \quad \dot{x}_1 &= y + u && \text{(3.247b)}\\
\dot{x}_2 &= x_1 && \text{(3.247c)}\\
y &= (1 - x_2^2)x_1 - x_2 && \text{(3.247d)}\\
x(0) &= x_0, \qquad t \in [t_0, t_f] && \text{(3.247e)}
\end{aligned}$$

For comparison, Figure 3.9 shows the open loop response with the control $u = 0$.

3.6.2 BVP for an OCP of ODE system

A local optimal control for $\mathcal{P}^V_{ODE}$ can be obtained through the necessary conditions developed in Section 3.2, under the assumption that there are no inflection points. The sufficient conditions (HJB equation) are more difficult to work, and often disfavored. Although a solution for the necessary conditions are often considered sufficient with additional information on the problem.

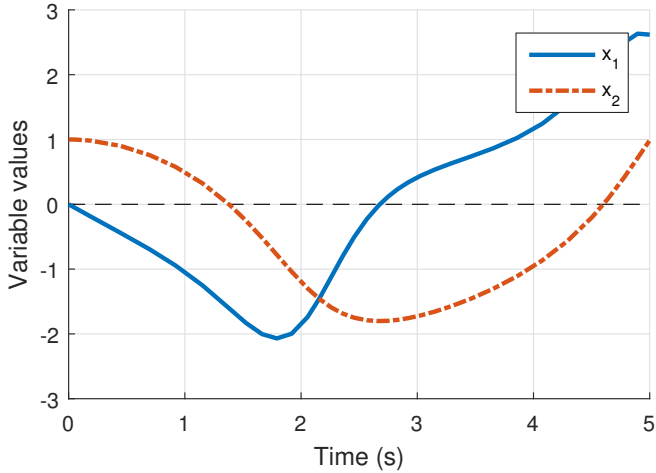

Figure 3.9: The open loop response of the Van der Pol oscillator.

Let us define the Hamiltonian for the problem $\mathcal{P}^V_{ODE}$,

$$H^V_{ODE}(x_1, x_2, u) = \left(x_1^2 + x_2^2 + u^2\right) + \lambda_1 \left[(1 - x_2^2)x_1 - x_2 + u\right] + \lambda_2 x_1 \quad (3.248)$$

Using the necessary conditions of Theorem 6 we obtain the following system of equations:

$$\frac{\partial H^V_{ODE}}{\partial x_1} = -\dot{\lambda}_1 = 2x_1 + \lambda_1(1 - x_2^2) + \lambda_2 \quad (3.249a)$$

$$\frac{\partial H^V_{ODE}}{\partial x_2} = -\dot{\lambda}_2 = 2x_2 + \lambda_1(-2x_2x_1 - 1) \quad (3.249b)$$

$$\frac{\partial H^V_{ODE}}{\partial u} = 2u + \lambda_1 = 0 \quad (3.249c)$$

$$\frac{\partial H^V_{ODE}}{\partial \lambda_1} = \dot{x}_1 = (1 - x_2^2)x_1 - x_2 + u \quad (3.249d)$$

$$\frac{\partial H^V_{ODE}}{\partial \lambda_2} = \dot{x}_2 = x_1, \quad x(0) = [1, 0]^T, \quad \lambda(t_f) = 0 \quad (3.249e)$$

Equation (3.249c) infer the optimal control law

$$u^* = -\frac{\lambda_1}{2} \quad (3.250)$$

By substituting (3.250) into (3.249), we can formulate a BVP to obtain the optimal control,

$$\dot{x}_1 = (1 - x_2^2)x_1 - x_2 - \frac{\lambda_1}{2} \qquad \dot{x}_2 = x_2 \tag{3.251a}$$

$$\dot{\lambda}_1 = -2x_1 - \lambda_1(1 - x_2^2) - \lambda_2 \qquad \dot{\lambda}_2 = -2x_2 - \lambda_1(-2x_2x_1 - 1) \tag{3.251b}$$

$$x(0) = [1, 0]^T \qquad \lambda(t_f) = 0 \tag{3.251c}$$

for which there is an initial condition for the state x and a terminal condition for the costate λ.

3.6.3 BVP for an OCP of DAE system

The same process applied for the problem with the ODE system can be applied for the problem $\mathcal{P}_{DAE}^V$. Let the Hamiltonian of the problem $\mathcal{P}_{DAE}^V$ be given by

$$H_{DAE}^V = \left(x_1^2 + x_2^2 + u^2\right) + \lambda_1 (y + u) + \lambda_2 x_1 + \nu \left[(1 - x_2^2)x_1 - x_2 - y\right] \tag{3.252}$$

Apply the necessary optimality conditions we achieve the system of equations:

$$\frac{\partial H_{DAE}^V}{\partial x_1} = -\dot{\lambda}_1 = 2x_1 + \lambda_2 + \nu(1 - x_2^2) \tag{3.253a}$$

$$\frac{\partial H_{DAE}^V}{\partial x_2} = -\dot{\lambda}_2 = 2x_2 + \nu(-2x_2x_1 - 1) \tag{3.253b}$$

$$\frac{\partial H_{DAE}^V}{\partial y} = \lambda_1 - \nu = 0, \qquad \frac{\partial H_{DAE}^V}{\partial u} = 2u + \lambda_1 = 0 \tag{3.253c}$$

$$\frac{\partial H_{DAE}^V}{\partial \lambda_1} = \dot{x}_1 = y + u, \qquad \frac{\partial H_{DAE}^V}{\partial \lambda_2} = \dot{x}_2 = x_1 \tag{3.253d}$$

$$\frac{\partial H_{DAE}^V}{\partial \nu} = (1 - x_2^2)x_1 - x_2 - y = 0 \tag{3.253e}$$

$$x(0) = [1, 0]^T, \qquad \lambda(t_f) = 0 \tag{3.253f}$$

From (3.253c) we can imply the optimal control law

$$u^* = -\frac{\lambda_1}{2} \tag{3.254}$$

which can be replaced on (3.253) to obtain the BVP

$$\dot{x}_1 = y - \frac{\lambda}{2}, \qquad \dot{x}_2 = x_1, \tag{3.255a}$$

$$\dot{\lambda}_1 = -2x_1 - \lambda_2 + \lambda_1(1 - x_2^2), \tag{3.255b}$$

$$\dot{\lambda}_2 = -2x_2 + \lambda_1(-2x_2x_1 - 1), \tag{3.255c}$$

$$(1 - x_2^2)x_1 - x_2 - y = 0, \qquad \lambda_1 - \nu = 0, \tag{3.255d}$$

$$x(0) = [1, 0]^T, \qquad \lambda(t_f) = 0 \tag{3.255e}$$

Notice that the algebraic equations (3.255d) have an explicit solution for y and for ν. We could eliminate these variables and reduce the system to an ODE system.

3.6.4 Indirect Multiple Shooting Method

To solve the BVPs (3.251) and (3.255), one of the possible approaches is the multiple shooting described in Section 2.4.

The shooting methods see the solution of an ODE/DAE system as a black-box only regarding the boundary values (x_0 and x_f) of the system, making use of numerical tools for solving the IVP.

First, lets us consider solving the BVP of the OCP of the ODE system (3.251). Let the state and costate be represented by $\widehat{x} = [x, \lambda]$, where $\widehat{x}_0$ and $\widehat{x}_f$ are the boundary conditions. Using Definition 4, the BVP of $\mathcal{P}^V_{ODE}$ has the following functions

$$F^V_{ODE}(\widehat{x}_0, T) = \left\{ \widehat{x}_f \in \mathbb{R}^{2N_x} \;\middle|\; \begin{array}{l} \dot{x}_1 = (1 - x_2^2)x_1 - x_2 - \frac{\lambda_1}{2} \\ \dot{x}_2 = x_2 \\ \dot{\lambda}_1 = -2x_1 - \lambda_1(1 - x_2^2) - \lambda_2 \\ \dot{\lambda}_2 = -2x_2 - \lambda_1(-2x_2x_1 - 1) \\ [x_1(t_0), x_2(t_0), \lambda_1(t_0), \lambda_2(t_0)] = \widehat{x}_0 \\ \widehat{x}_f = [x_1(t_f), x_2(t_f), \lambda_1(t_f), \lambda_2(t_f)] \\ t \in T \end{array} \right. \tag{3.256a}$$

$$G^V_{ODE}(\widehat{x}_0, \widehat{x}_f) = \begin{bmatrix} \widehat{x}_{0,1} - x_{0,1} \\ \widehat{x}_{0,2} - x_{0,2} \\ \widehat{x}_{f,3} \\ \widehat{x}_{f,4} \end{bmatrix} = 0 \tag{3.256b}$$

where T is a time interval. The function $F^V_{ODE}(\widehat{x}_0, T)$ solves an IVP of the underlying ODE system with initial conditions $\widehat{x}_0$ and integration interval T, and returns the states at the final time. The function

G^V_{ODE} gathers the boundary conditions, this function has the property that when the boundary conditions are satisfied $G^V_{ODE} = 0$.

Let us split the time interval $[t_0, t_f]$ into N subintervals, such that each subinterval T_i is defined by

$$T_i = [t_{i-1}, t_i] \tag{3.257}$$

where $\delta t = \frac{t_f - t_0}{N}$ and $t_i = t_{i-1} + \delta t$ for $i = 1, \ldots, N$, in particular for $i = N$ we have $t_N = t_f$.

Let $\widehat{x}^i_0$ and $\widehat{x}^i_f$ be the boundary values at the interval T_i. The functions presented in (3.256) can be used to formulate a nonlinear system of equations

$$\widehat{x}^i_f = F^V_{ODE}(x^i_0, T_i) \qquad i = 1, \ldots, N \tag{3.258a}$$

$$x^{i-1}_f = x^i_0 \qquad i = 2, \ldots, N \tag{3.258b}$$

$$0 = G^V_{ODE}(x^1_0, x^N_f) \tag{3.258c}$$

whose derivatives can be obtained using the sensitivity calculations, Section 2.5.

Using the optimization solver IPOPT [22] to solve (3.258) and the numerical integrator Sundials' CVODES [23], within CasADi framework [24], to evaluate the function and the derivatives of F^V_{ODE}, the values obtained for the initial conditions is

$$\widehat{x}^1_0 = \begin{bmatrix} 0 \\ 1 \\ 0.8234 \\ 4.8742 \end{bmatrix}, \tag{3.259}$$

with optimal objective 2.86697. Figure 3.10 shows the states and controls during the interval $[t_0, t_f]$.

For solving the OCP for the DAE through the BVP (3.255), let us define $\widehat{x} = [x, \lambda]$, with $\widehat{x}_0$ and $\widehat{x}_f$ being the initial and final

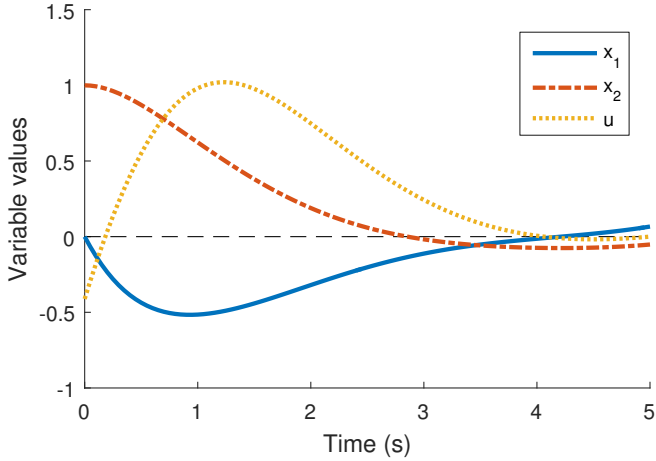

Figure 3.10: Optimal trajectory for x_1, x_2, and u obtained with indirect multiple shooting of the Van der Pol oscillator modeled with ODE.

conditions, respectively. According to Definition 4, let us define

$$F^V_{DAE}(\widehat{x}^i_0, T_i) = \left\{ \widehat{x}_f \in \mathbb{R}^{2N_x} \;\middle|\; \begin{array}{l} \dot{x}_1 = y - \frac{\lambda}{2} \\ \dot{x}_2 = x_1 \\ \dot{\lambda}_1 = -2x_1 - \lambda_2 + \lambda_1(1 - x_2^2) \\ \dot{\lambda}_2 = -2x_2 + \lambda_1(-2x_2x_1 - 1) \\ (1 - x_2^2)x_1 - x_2 - y = 0 \\ \lambda_1 + \nu = 0 \\ \left[x_1(t^i_0), x_2(t^i_0), \lambda_1(t^i_0), \lambda_2(t^i_0)\right] = \widehat{x}^i_0 \\ \widehat{x}_f = \left[x_1(t^i_f), x_2(t^i_f), \lambda_1(t^i_f), \lambda_2(t^i_f)\right] \\ t \in T_i = [t^i_0, t^i_f] \end{array} \right\} \tag{3.260a}$$

$$G^V_{DAE}(\widehat{x}_0, \widehat{x}_f) = \begin{bmatrix} \widehat{x}_{0,1} - x_{0,1} \\ \widehat{x}_{0,2} - x_{0,2} \\ \widehat{x}_{f,3} \\ \widehat{x}_{f,4} \end{bmatrix} = 0 \tag{3.260b}$$

where F^V_{DAE} is a function that solves an IVP of the DAE system with initial condition $\widehat{x}^i_0$ and integration interval T_i, which starts at t^i_0 and ends at t^i_f, and returns the states at the final time. By defining the

subinterval T_i as given in (3.257), for all $i = 1, \ldots, N$, a nonlinear system of equations can be formulated as

$$\widehat{x}_f^i = F_{DAE}^V(x_0^i, T_i) \qquad i = 1, \ldots, N \tag{3.261a}$$

$$x_f^{i-1} = x_0^i \qquad i = 2, \ldots, N \tag{3.261b}$$

$$0 = G_{DAE}^V(x_0^1, x_f^N) \tag{3.261c}$$

where $\widehat{x}_0^i$ and $\widehat{x}_f^i$ correspond to the initial and final conditions in the subinterval T_i.

Assuming $N = 10$ and using IPOPT to solve (3.261), where the function F_{DAE}^V had its values and derivatives evaluated using the numerical integrator Sundial's IDAS [23], all implemented in the CasADi framework [24]. The initial conditions for x and λ are $x_0 = [0, 1]$ and $\lambda_0 = [0.8234, 4.8742]$. Figure 3.11 shows the behavior of the states x_1 and x_2, the control u, and the algebraic variable y.

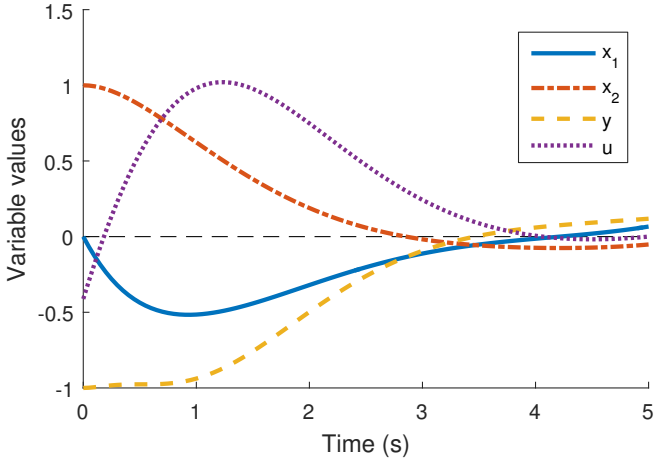

Figure 3.11: Optimal trajectory for x_1, x_2, y, and u obtained with indirect multiple shooting of the Van der Pol oscillator modeled with DAE.

3.6.5 Indirect Collocation Method

An alternative manner to solve the BVP, which arises from the necessary optimality conditions, is the collocation method detailed in Section 2.6.

Herein, the collocation method will be applied to solve the BVP obtained from the conditions of the OCP of the DAE system (3.255). The approach can be applied to obtain the solution for the OCP of the ODE system as well.

Let us define the variables $\widehat{x} = [x, \lambda]$ and $\widehat{y} = [y, \nu]$, and the functions

$$\widehat{f} = \begin{bmatrix} y - \frac{\lambda}{2} \\ x_1 \\ -2x_1 - \lambda_2 + \lambda_1(1 - x_2^2) \\ -2x_2 + \lambda_1(-2x_2x_1 - 1) \end{bmatrix} \tag{3.262a}$$

$$\widehat{g} = \begin{bmatrix} (1 - x_2^2)x_1 - x_2 - y \\ \lambda_1 - \nu \end{bmatrix} \tag{3.262b}$$

which allows us to state the DAE in the compact form

$$\dot{\widehat{x}} = \widehat{f}(\widehat{x}, \widehat{y}, u, t) \tag{3.263a}$$

$$\widehat{g}(\widehat{x}, \widehat{y}, u, t) = 0 \tag{3.263b}$$

The collocation method requires two Lagrangean polynomial basis. According to (2.111), we define the polynomial basis for the states

$$\ell_j(\tau) = \prod_{k=0, \neq j}^{K} \frac{(\tau - \tau_k)}{(\tau_j - \tau_k)} \tag{3.264}$$

and (2.121) gives the basis for the algebraic and control variables, which is

$$\widehat{\ell}_j(\tau) = \prod_{k=1, \neq j}^{K} \frac{(\tau - \tau_k)}{(\tau_j - \tau_k)} \tag{3.265}$$

Consider that the time interval is split into N subintervals, where in each subinterval the states are approximated by a K-th order polynomial, while the algebraic and control variables are approximated by a $K-1$-th order polynomial. From equations (2.112), (2.122), and (2.125), for all $i = 1, \ldots, N$ with $t \in [t_{i-1}, t_i]$ we have

$$\widehat{x}(t) = \sum_{j=0}^{K} \ell_j(\tau)\widehat{x}_{ij} \tag{3.266a}$$

$$\widehat{y}_i(t) = \sum_{j=1}^{K} \widehat{\ell}_j(\tau)\widehat{y}_{ij} \tag{3.266b}$$

where

- τ is the normalized time variable in the subinterval T_i, where $\tau = \frac{t_0^i + t}{h_i}$ with h_i being the length and t_0^i the initial time of the interval.

- $\widehat{x}_{ij} \in \mathbb{R}^{N_x}$ is the state in the subinterval T_i at the collocation point j.

- $\widehat{y}_{ij} \in \mathbb{R}^{N_y}$ is the algebraic variable in the subinterval T_i at the collocation point j.

Since a control law was used to replace the control variable u, the indirect collocation method does not have to approximate the controls.

In order to make the representation (3.266) correspond to the dynamics of the DAE system, we have to enforce the equations (2.117) and (2.119) for the state variables. This is accomplished by stating for all $i = 1, \ldots, N$ and $k = 1, \ldots, K$:

$$\sum_{j=0}^{K} \widehat{x}_{ij} \frac{d\ell_j}{d\tau}(\tau_k) = h_i \widehat{f}(\widehat{x}_{ik}, \widehat{y}_{ik}, u_{ik}, t_{ik}) \tag{3.267a}$$

and, to satisfy continuity, for all $i = 1, \ldots, N-1$:

$$\widehat{x}_{i+1,0} = \sum_{j=0}^{K} \ell_j(1) \widehat{x}_{ij} \tag{3.267b}$$

For the algebraic variables, replicating (2.124), for all $i = 1, \ldots, N$ and for all $j = 1, \ldots, K$

$$\widehat{g}(\widehat{x}_{ij}, \widehat{y}_{ij}, \widehat{u}_{ij}, t_{ij}) = 0 \tag{3.268}$$

Using equations (3.267) and (3.268) together with the bound-

ary conditions, the nonlinear system is formulated

$$\sum_{j=0}^{K} \widehat{x}_{ij} \frac{d\ell_j}{d\tau}(\tau_k) = h_i \widehat{f}(\widehat{x}_{ik}, \widehat{y}_{ik}, t_{ik}) \qquad \forall i, \forall k \tag{3.269a}$$

$$\widehat{x}_{i+1,0} = \sum_{j=0}^{K} \ell_j(1) \widehat{x}_{ij} \qquad i = 1, \ldots, N-1 \tag{3.269b}$$

$$\widehat{g}(\widehat{x}_{ij}, \widehat{y}_{ij}, t_{ij}) = 0, \qquad \forall i, \forall j \tag{3.269c}$$

$$\begin{bmatrix} \widehat{x}_{ij,1} - x_{0,1} \\ \widehat{x}_{ij,2} - x_{0,2} \end{bmatrix} = 0, \qquad i = 1, j = 1 \tag{3.269d}$$

$$\begin{bmatrix} \widehat{x}_{ij,3} \\ \widehat{x}_{ij,4} \end{bmatrix} = 0, \qquad i = N, j = K \tag{3.269e}$$

where (3.269d) is the initial condition, and (3.269e) is the final condition, both stated in (3.255e).

If we use $N = 10$ and $K = 4$, and use IPOPT to solve (3.269) the initial conditions found is

$$\widehat{x} = \begin{bmatrix} 0 \\ 1 \\ 0.8227 \\ 4.8739 \end{bmatrix} \tag{3.270}$$

which induces the objective equal to 2.86695. The trajectories found are very similar to the ones obtained with the indirect multiple shooting, presented in Figure 3.11.

3.7 DIRECT METHODS

While most of the chapter was centered around the indirect methods, the direct methods are vastly used and more easily formulated. In this section, the shooting and the collocation methods will be applied so solve the OCP *directly*. The solution is direct in the sense that the optimization process iteratively produces a solution that minimizes an objective function, rather than a solution that satisfies the optimality conditions.

To illustrate how these methods work, the same Van der Pol Oscillator, described in Section 3.6.1, will be used.

The direct methods do not handle objectives with integrals, however Theorem 5 can be used to recast the OCP with no integral

in the objective. Let x_c be the integral cost state, whose initial condition is zero, and the vector of all states be $\widehat{x} = [x_1, x_2, x_c]$. Therefore, in this section, we refer to $\mathcal{P}^V_{ODE}$ as

$$\begin{aligned}
\mathcal{P}^V_{ODE}: \quad \min_{x,u} \; J =& x_c(t_f) && \text{(3.271a)}\\
\text{s.t.:} \quad & \dot{x}_1 = (1 - x_2^2)x_1 - x_2 + u && \text{(3.271b)}\\
& \dot{x}_2 = x_1 && \text{(3.271c)}\\
& \dot{x}_c = x_1^2 + x_2^2 + u^2 && \text{(3.271d)}\\
& x(0) = x_0, && \text{(3.271e)}\\
& x_c(0) = 0 && \text{(3.271f)}\\
& t \in [t_0, t_f] && \text{(3.271g)}
\end{aligned}$$

and the problem $\mathcal{P}^V_{DAE}$ is recast as

$$\begin{aligned}
\mathcal{P}^V_{DAE}: \quad \min_{x,y,u} \; J =& x_c(t_f) && \text{(3.272a)}\\
\text{s.t.:} \quad & \dot{x}_1 = y + u && \text{(3.272b)}\\
& \dot{x}_2 = x_1 && \text{(3.272c)}\\
& \dot{x}_c = x_1^2 + x_2^2 + u^2 && \text{(3.272d)}\\
& y = (1 - x_2^2)x_1 - x_2 && \text{(3.272e)}\\
& x_c(0) = 0 && \text{(3.272f)}\\
& x(0) = x_0, \qquad t \in [t_0, t_f] && \text{(3.272g)}
\end{aligned}$$

The following subsections will show how to transform the optimal control problems (3.271) and (3.272) into mathematical optimization problems, which can be solved with standard nonlinear optimization solvers.

3.7.1 Direct Multiple-Shooting

To solve the OCPs (3.271) and (3.272) with multiple shooting in a direct manner, we will use the structure developed in Section 2.4.

Since the procedures to solve an OCP of an ODE system and an OCP of a DAE system do not differ, herein we follow with the solution of the problem $\mathcal{P}^V_{DAE}$ (3.272).

Recall that for the indirect methods, a optimal control was obtained as a consequence of the necessary conditions. For the direct methods, there is no such optimal control law. What is done in

the case of the multiple shooting is to parametrize the control variable. In practice, one chooses a piecewise constant control profile, assuming that at each shooting subinterval the control is equal to some variable, whose value will be determined by the optimization problem. That is, for $t \in T_i$ and $i \in 1, \ldots, N$

$$u(t) = u_i \tag{3.273}$$

where $u_i \in \mathbb{R}^{N_u}$.

However, in order to obtain the same solution of the indirect methods, the control needs a good approximation. Such approximation can be achieved with polynomial interpolations, which is conveniently done with Lagrangian interpolation polynomials, the same used for collocation methods. Considering that a K-th order polynomial is used, for $t \in T_i$

$$u(t) = \sum_{j=1}^{K} \widehat{\ell}_j(\tau) u_{ij} \tag{3.274}$$

where $u_{ij} \in \mathbb{R}^{N_u}$ with $i = 1, \ldots, N$ and $j = 1, \ldots, K$ are parameters that define the control profile at the collocation τ_j in the subinterval T_i, and $\widehat{\ell}$ is defined in (2.121). To obtain results comparable with the indirect methods, the latter approximation is chosen.

Let us define the vector

$$\theta_i = [u_{ij} : j = 1, \ldots, K] \tag{3.275}$$

and the functions

$$F^V_{DAE}(\widehat{x}^i_0, \theta_i, T_i) = \left\{ \widehat{x}_f \in \mathbb{R}^{N_x+1} \;\middle|\; \begin{array}{l} \dot{x}_1 = y + u \\ \dot{x}_2 = x_1 \\ \dot{x}_c = x_1^2 + x_2^2 + u^2 \\ (1 - x_2^2)x_1 - x_2 - y = 0 \\ u(t) = \sum_{j=1}^{K} \widehat{\ell}_j(\tau) u_{ij} \\ \left[x_1(t^i_0), x_2(t^i_0), \lambda_1(t^i_0), \lambda_2(t^i_0)\right] = \widehat{x}^i_0 \\ \widehat{x}_f = \left[x_1(t^i_f), x_2(t^i_f), \lambda_1(t^i_f), \lambda_2(t^i_f)\right] \\ t \in T_i = [t^i_0, t^i_f] \end{array} \right\} \tag{3.276a}$$

$$G^V_{DAE}(\widehat{x}_0, \widehat{x}_f) = \begin{bmatrix} \widehat{x}_{0,1} - x_{0,1} \\ \widehat{x}_{0,2} - x_{0,2} \\ \widehat{x}_{0,c} \\ \widehat{x}_{f,c} - J \end{bmatrix} = 0 \tag{3.276b}$$

where F^V_{DAE} is a function that solves an IVP of the DAE system with initial condition $\widehat{x}^i_0$ and integration interval T_i, which starts at t^i_0 and ends at t^i_f, and returns the states at the final time.

Let $\widehat{x}^i_0$ and $\widehat{x}^i_f$ be initial and final condition in the subinterval T_i, respectively. Using the functions described in (3.276), we can state the nonlinear programming (NLP) problem

$$\min_{\theta_i, \widehat{x}^i_0, \widehat{x}^i_f} \quad J \tag{3.277a}$$

$$\text{s.t.:} \quad \widehat{x}^i_f = F^V_{DAE}(\widehat{x}^i_0, \theta_i, T_i) \quad i = 1, \ldots, N \tag{3.277b}$$

$$\widehat{x}^{i-1}_f = \widehat{x}^i_0 \qquad i = 2, \ldots, N \tag{3.277c}$$

$$G^V_{DAE}(x^1_0, x^N_f) = 0 \tag{3.277d}$$

Assuming $N = 10$ and $K = 3$, and using IPOPT to solve the NLP the values obtained for the objective function is 2.86695. The dynamic of the states, algebraic, and control variables are identical to those obtained with the indirect multiple shooting method, depicted in Figure 3.11.

One of the advantages of using direct methods is the ease to include constraint on the states, algebraic, and control variables. The state constraint

$$x_L \leq x(t) \leq x_U \tag{3.278}$$

can be implemented by including the constraints

$$x_L \leq \widehat{x}^i_0 \leq x_U \qquad i = 1, \ldots, N \tag{3.279a}$$

$$x_L \leq \widehat{x}^i_f \leq x_U \qquad i = 1, \ldots, N \tag{3.279b}$$

in the NLP problem (3.277). Notice however, that the constraint will only be satisfied at the beginning and end of the subintervals, but by making a refined discretization of the interval, a satisfactory result can be obtained. A similar approach can be used to include constraints in the algebraic variables.

In the case of control constraints, the constraints can be applied directly to the parametrization variables. For example, given a constraint

$$u_L \leq u(t) \leq u_U \tag{3.280}$$

the inclusion of the constraint

$$u_L \leq u_{ij} \leq u_U \qquad i = 1, \ldots, N, j = 1, \ldots, K \tag{3.281}$$

in the NLP problem (3.277) ensures that, at the interpolation points, the constraint will not be violated.

3.7.2 Direct Collocation Method

The direct collocation use the discretization structure developed in Section 2.6 to formulate the NLP problem. The objective function of the optimization problem is the last collocation point of the cost state at the last subinterval.

Just like the direct multiple shooting, the controls have do be discretized. For the collocation, a piecewise constant control is applicable, however less common since the structure of the problem already makes use of polynomial interpolations, using polynomials for discretizing the control is more convenient.

To apply the collocation, let us define the extended state

$$\widehat{x} = [x_1, x_2, x_c] \tag{3.282}$$

and the variables $\widehat{x}_{ij} \in \mathbb{R}^{N_x}$, $y_{ij} \in \mathbb{R}^{N_y}$, $u_{ij} \in \mathbb{R}^{N_u}$ as the state, algebraic, and control variables in the interval T_i at the collocation point j. Therefore, for $t \in T_i$, the states, algebraic, and control variables are given by

$$\widehat{x}(t) = \sum_{j=0}^{K} \ell_j(\tau)\widehat{x}_{ij} \tag{3.283a}$$

$$y_i(t) = \sum_{j=1}^{K} \widehat{\ell}_j(\tau) y_{ij} \tag{3.283b}$$

$$u_i(t) = \sum_{j=1}^{K} \widehat{\ell}_j(\tau) u_{ij} \tag{3.283c}$$

and the function

$$\widehat{f} = \begin{bmatrix} y + u \\ x_1 \\ x_1^2 + x_2^2 + u^2 \end{bmatrix} \tag{3.284}$$

which allows us to create the nonlinear system of equations that

describe the dynamics of the system

$$\sum_{j=0}^{K} \widehat{x}_{ij} \frac{d\ell_j}{d\tau}(\tau_k) = h_i \widehat{f}(\widehat{x}_{ik}, y_{ik}, u_{ik}, t_{ik}) \qquad \forall i, \forall k \tag{3.285a}$$

$$\widehat{x}_{i+1,0} = \sum_{j=0}^{K} \ell_j(1) \widehat{x}_{ij} \qquad i = 1, \ldots, N-1 \tag{3.285b}$$

$$g(\widehat{x}_{ij}, y_{ij}, u_{ij}, t_{ij}) = 0, \qquad \forall i, \forall j \tag{3.285c}$$

By using the system approximation (3.285) with the initial conditions, we can formulate the NLP problem

$$\min_{\widehat{x}_{ij}, y_{ij}, u_{ij}} \quad J \tag{3.286a}$$

$$\text{s.t.:} \quad \sum_{j=0}^{K} \widehat{x}_{ij} \frac{d\ell_j}{d\tau}(\tau_k) = h_i \widehat{f}(\widehat{x}_{ik}, y_{ik}, u_{ik}, t_{ik}) \qquad \forall i, \forall k \tag{3.286b}$$

$$\widehat{x}_{i+1,0} = \sum_{j=0}^{K} \ell_j(1) \widehat{x}_{ij} \qquad i = 1, \ldots, N-1 \tag{3.286c}$$

$$g(\widehat{x}_{ij}, y_{ij}, u_{ij}, t_{ij}) = 0, \qquad \forall i, \forall j \tag{3.286d}$$

$$\widehat{x}_{ij} = \begin{bmatrix} x_0 \\ 0 \end{bmatrix} \qquad i = 1, j = 0 \tag{3.286e}$$

$$J = \widehat{x}_{ij,c} \qquad i = N, j = K \tag{3.286f}$$

The NLP problem (3.286) can be solved with a standard NLP solver, as the IPOPT. Assuming $N = 10$ and $K = 4$, the result obtained is $J = 2.8669$. The method also converges to the same solution of the other approaches, therefore have a dynamic almost identical to those obtained in Figure 3.11.

Constraints on the states can be easily implemented by constraining the parameters of the approximation. For instance, given the constraint

$$x_L \leq x(t) \leq x_U, \tag{3.287}$$

the constraint can be implemented by including in the NLP problem (3.286) the constraint

$$x_L \leq \widehat{x}_{ij} \leq x_U \qquad i = 1, \ldots, N, j = 0, \ldots, K \tag{3.288}$$

For the algebraic and control variables, the approach is very similar. Given the constraints

$$y_L \leq y(t) \leq y_U, \text{ and } u_L \leq u(t) \leq u_U \tag{3.289a}$$

the equivalent in nonlinear programming will be

$$y_L \leq y_{ij} \leq y_U, \text{ and } u_L \leq u_{ij} \leq u_U \tag{3.290a}$$

for $i = 1, \ldots, N$ and $j = 1, \ldots, K$.

3.8 SUMMARY

With the content of this chapter, we are able to solve optimal control problems of ODE and DAE systems. These OCPs can have bound constraints in the state, algebraic, and control variables. In addition, with the development of Sections 3.2.1 and 3.2.2, we are able to solve problems with free or fixed final state, fixed or final time, and with final time constraints. Combining the collocation method and the shooting methods presented in Chapter 2 with the theory developed in this chapter, we were able to solve an OCP using indirect and direct approaches for a practical case.

4 ALGORITHM DEVELOPMENT

This chapter presents the main contributions of this work, which is an algorithm for solving optimal control problems (OCP) of systems of differential-algebraic equations (DAE). The proposed algorithm is based on the augmented Lagrange [1] for constrained optimization, a brief exposition of the algorithm is given in Appendix B. The original augmented Lagrange algorithm solves problems in vector spaces, therefore cannot be applied for solving OCPs, which are problems in function spaces.

For the proposed algorithm some mathematical properties are developed, including global convergence, local convergence, and convergence of sub-optimal solutions. The algorithm and the properties are verified with numerical experiments using the Van der Pol oscillator.

4.1 PROBLEM DEFINITION

Consider an optimal control problem (OCP) for a system of differential-algebraic equations (DAE) in the form:

$$\mathcal{P}_0: \quad \min J(x,y,u) = \int_{t_0}^{t_f} L(x,y,u,t)\,dt \tag{4.1a}$$

$$\text{s.t.:} \quad \dot{x} = f(x,y,u,t) \tag{4.1b}$$

$$g(x,y,u,t) = 0 \tag{4.1c}$$

$$u(t) \in U_B \tag{4.1d}$$

$$x(0) = x_0 \tag{4.1e}$$

$$t \in [t_0, t_f] \tag{4.1f}$$

with

$$U_B = \{\mathrm{u} \in U \,|\, u_L \leq \mathrm{u} \leq u_U\} \tag{4.2}$$

where $x(t) \in X = \mathbb{R}^{N_x}$ is the state variable, $y(t) \in Y = \mathbb{R}^{N_y}$ is the algebraic variable, $u(t) \in U_V \subset U = \mathbb{R}^{N_u}$ is the control variable, and t is the time variable. The function of dynamics f, the function of algebraic relations g, and the function of dynamic cost L are assumed to be continuously differentiable with respect to their arguments. The DAE system formed by (4.1b) and (4.1c) are assumed to be in the semi-explicit form, which means that it is solvable for y and the partial derivative $\frac{\partial g}{\partial y}$ is invertible (Section 2.2). Problems with the format $\mathcal{P}_0$ but that include a final cost function $\Phi(x(t_f), t_f)$

can be adapted to this approach by transforming the objective using Theorem 5.

4.2 ALGORITHM SUMMARY

Given an optimal control problem in the form $\mathcal{P}_0$, the algorithm proposed in this work solves the OCP by relaxing (4.1c) and introducing a new objective functional,

$$J_\mu(x, y, u, \nu) = \int_{t_0}^{t_f} \mathcal{L}_\mu(x, y, u, \nu, t)\, dt \tag{4.3}$$

where the function $\mathcal{L}_\mu$ is defined by

$$\mathcal{L}_\mu(x, y, u, \nu, t) = L(x, y, u, t) + \nu(t)^T \left[g(x, y, u, t)\right] + \frac{\mu}{2} \left\|g(x, y, u, t)\right\|^2, \tag{4.4}$$

where $\mu > 0$ is a scalar, and the function $\nu : [t_0, t_f] \to \mathbb{R}^{N_y}$ is an approximation of the multiplier function ν^* that satisfies the optimality conditions (Theorem 11) of problem $\mathcal{P}_0$ given by (4.1).

The functional (4.3) is the objective of the auxiliary optimal control problem solved by the algorithm at each iteration k, which is given by

$$\mathcal{P}_\mathcal{L}(\mu_k, \nu_k): \quad \min_{y,u} J_{\mu_k} = \int_{t_0}^{t_f} \mathcal{L}_{\mu_k}(x, y, u, \nu_k, t)\, dt \tag{4.5a}$$

$$\text{s.t.:} \quad \dot{x} = f(x, y, u, t) \tag{4.5b}$$

$$x(0) = x_0 \tag{4.5c}$$

$$u \in U_B \tag{4.5d}$$

$$t \in [t_0, t_f] \tag{4.5e}$$

Notice that without an algebraic equation, the variable y is free to be optimized. In this sense, the algebraic variable plays the same role as the control variable u. Therefore, we define an extended control variable $\widehat{u} = [u, y]$, where $\widehat{u} \in \widehat{U} = U_B \times Y$. Using $\widehat{u}$, the problem $\mathcal{P}_\mathcal{L}$ meets the standard form of an OCP of ODE (3.87), and the optimality conditions of Theorem 6 apply.

The algorithm steps are given by Algorithm 1. Therein the parameter μ_0 is the initial value of the sequence $\{\mu_k\}$, ν_0 is the initial function for the sequence $\{\nu_k\}$, and ε_g is the tolerance on the violation of the algebraic constraint.

Some features of this algorithm should be pointed:

Algorithm 1 Augmented Lagrangian for Optimal Control

Require: μ_0, ν_0, and ε_g:
1: **for** $k = 1, 2, \ldots$ **do**
2: $(J_k, x_k, y_k, u_k) \leftarrow solve\{\mathcal{P}_{\mathcal{L}}(\mu_k, \nu_k)\}$
3: $\nu_{k+1} \leftarrow \nu_k + \mu_k g(x_k, y_k, u_k)$
4: $\mu_{k+1} \leftarrow update\_mu\{\mu_k\}$
5: **if** $\|g(x_k, y_k, u_k)\| < \varepsilon_g$, $\forall t \in [t_0, t_f]$ **then**
6: **return** u_k
7: **end if**
8: **end for**

1. In line 2, the pseudo-function $solve$ can use any suitable method to obtain the solution of the OCP $\mathcal{P}_{\mathcal{L}}$, for instance using one of the direct or indirect methods presented in Chapter 3. An important condition is that if y_k^* is parametrized, *e.g.* as a polynomial in direct methods, then it should be a sufficient good approximation to prevent the approximation error from hindering the algorithm. To speed up the algorithm iteration, the solution of the previous iteration can be used as an initial guess to compute the next solution.

2. In line 3, it is performed the update of the function ν_k. Since generic functions cannot be stored in computers, the function ν_k is approximated by a parametrized function, this subject is discussed further in Section 4.4.

3. In line 4, the pseudo-function $update\_mu$ performs an increment in the penalty parameter μ_k. While doing the theoretical analysis, we adopt the most commonly applied rule for updating the penalty parameter of the augmented Lagrange for constrained optimization (B.3), that is

$$\mu_{k+1} = \beta \mu_k \tag{4.6}$$

where β is a scalar greater than 1, with its values usually ranging from 2 to 10. However if $\mu_k \to \infty$ then the computational problem becomes ill conditioned. Therefore, for using the augmented Lagrange for constrained optimization, the alternative update rule

$$\mu_{k+1} = \begin{cases} \beta \mu_k & \text{if } \beta \mu_k < \mu_{\max} \\ \mu_{\max} & \text{otherwise} \end{cases} \tag{4.7}$$

is preferred for practical application.

4. Regarding the initial conditions μ_0 and ν_0. The initial penalization μ_0 cannot be 0. Small values will make the problem lose its structure and become hard to solve since the algebraic constraint g is not taken into account. Large values make the problem difficult to solve. A reasonable value to start is $\mu_0 = 1$. The initial multiplier ν_0 is an approximation of the multiplier, so if there is any information on the multiplier it should be used. On the other hand, if there is no information a good start is $\nu_0 = 0$, then the algorithm will take into account only the quadratic penalization on the algebraic equation.

The highlights of the algorithm proposed in this work are:

1. By relaxing the algebraic equations, the algorithm transforms the DAE system into an ODE system. This reduction makes optimal control more accessible, given that ODE solvers are widely available and have reduced computational cost.

2. The algorithm solves an OCP with the form (3.170), which allows the inclusion of bounds in the control variables. In addition, the algorithm makes it easy to handle bound constraints on algebraic and state variables, that is $y(t) \in Y_B$ and $x(t) \in X_B$, with

$$Y_B = \{\mathrm{y} \in Y \,|\, y_L \leq \mathrm{y} \leq y_U\} \tag{4.8a}$$

$$X_B = \{\mathrm{x} \in X \,|\, x_L \leq \mathrm{x} \leq x_U\} \tag{4.8b}$$

This approach benefits indirect methods and is further discussed in Section 4.5.

4.3 MATHEMATICAL DEMONSTRATIONS[1]

In the last section an algorithm was proposed for solving problems in the form $\mathcal{P}_0$ (4.1). For such class of problems, some properties can be shown for the algorithm. Under the assumption that $\langle x_k, y_k, u_k \rangle$ is the solution of $\mathcal{P}_\mathcal{L}$ at the k-iteration of the algorithm and that the sequence $\{\langle x_k, y_k, u_k \rangle\}$ converges uniformly, these properties can be informally summarized:

[1]The development here follows the structure presented in [1] for demonstration of the augemented Lagrange method for constrained optimization.

1. If problem $\mathcal{P}_{\mathcal{L}}(\mu_k, \nu_k)$ is solved to global optimality at each iteration, then the algorithm produces a sequence $\{\langle x_k, y_k, u_k\rangle\}$ that converges to $\langle x^*, y^*, u^*\rangle$, which is a global optimum of $\mathcal{P}_0$.

2. If problem $\mathcal{P}_{\mathcal{L}}(\mu_k, \nu_k)$ is solved to local optimality at each iteration, then the algorithm produces a sequence $\{\langle x_k, y_k, u_k\rangle\}$ that converges to $\langle x^*, y^*, u^*\rangle$ that is a local optimum of $\mathcal{P}_0$.

3. The sequence $\{\nu_k + \mu_k g(x_k, y_k, u_k, t)\}$ converges to a limiting function ν^*. In particular if one defines $\nu_{k+1} = \nu_k + \mu_k g(x_k, y_k, u_k, t)$, then $\{\nu_k\} \to \nu^*$.

4. The sequences $\{x_k\}$, $\{y_k\}$, and $\{u_k\}$ are uniform convergent sequences.

Through this section, these properties will be precisely stated and the proofs given. Although the properties and proofs follow the same sequence of [1], the demonstrations here presented are original contributions of this work.

For the sake of organization, let us state the auxiliary OCP that the algorithm uses at each iteration. Let $J_\mu(x, u, \lambda)$ be the objective functional, which is

$$J_\mu(x, y, u, \nu) = \int_{t_0}^{t_f} \mathcal{L}_\mu(x, y, u, \nu, t)\, dt \tag{4.9}$$

where the function $\mathcal{L}_\mu$ is defined by

$$\mathcal{L}_\mu(x, y, u, \nu, t) = L(x, y, u, t) + \nu(t)^T \left[g(x, y, u, t)\right] + \frac{\mu}{2} \left\|g(x, y, u, t)\right\|^2 \tag{4.10}$$

The method consists of solving a sequence of problems of the form

$$\mathcal{P}_{\mathcal{L}}(\mu_k, \nu_k): \quad \min_{x,y,u} J_{\mu_k} = \int_{t_0}^{t_f} \mathcal{L}_{\mu_k}(x, y, u, \nu_k, t)\, dt \tag{4.11a}$$

$$\text{s.t.: } \dot{x} = f(x, y, u) \qquad \forall t \in [t_0, t_f] \tag{4.11b}$$

$$u(t) \in U_B \qquad \forall t \in [t_0, t_f] \tag{4.11c}$$

$$x(0) = x_0 \tag{4.11d}$$

$$t \in [t_0, t_f] \tag{4.11e}$$

where $\nu_k \in \mathcal{C}^1[t_0, t_f]$ is an approximation of the multiplier with $\{\nu_k\}$ being a bounded sequence[2], and $\mu_k \in \mathbb{R}$ is the penalty parameter where $\{\mu_k\}$ is a penalty parameter sequence satisfying:

$$0 < \mu_k < \mu_{k+1} \qquad \forall k \tag{4.12a}$$

$$\mu_k \to \infty \quad \text{as} \quad k \to \infty \tag{4.12b}$$

If we would follow a penalty method only, we would make $\nu = 0$ in (4.3) and by making the penalty parameter μ go to infinity we would force $g(x, u, t)$ to go to zero. By doing so we need to iterate indefinitely to reduce the value of the algebraic equation. On the other hand, by including an approximation of the multiplier ν_k we can have better properties which allow in practice to obtain a lower error for the relaxation of the algebraic equation.

Before going into the mathematical properties of the method, it is important to understand what the method is attempting to achieve. At first we develop the properties from the penalty perspective, which means we only consider the effects of making $\mu_k \to \infty$.

Let $J^g_{\mu_k}$ be a functional of terms of J_{μ_k} that depends on the algebraic function g, that is

$$J^g_{\mu_k}(x, y, u, \nu_k) = \int_{t_0}^{t_f} \nu_k^T g(x, y, u, t) + \frac{\mu_k}{2} \|g(x, y, u, t)\|^2 \, dt. \tag{4.13}$$

Seen as a penalty method, intuitively it follows that if $\{\nu_k\}$ is a sequence of bounded functions, and g is continuous on its domain, then by making $\mu_k \to \infty$ the functional $J^g_{\mu_k}$ goes to zero if $g(x, y, u, t) = 0$ for all $t \in [t_0, t_f]$, otherwise $J^g_{\mu_k}$ goes to infinity.

Let us define a functional $\widehat{J}(x, y, u)$ such that

$$\widehat{J}(x, y, u) \triangleq \begin{cases} J(x, y, u) & \text{if } g(x, y, u, t) = 0 \quad \forall t \in [0, t_f] \\ \infty & \text{otherwise} \end{cases} \tag{4.14}$$

where J is the objective functional defined in (4.1a). Letting J^* the optimal value for the problem $\mathcal{P}_0$ (4.1), the following equality holds

$$\begin{aligned} J^* = \inf_{x,y,u} \mathcal{P}_0 &= \inf_{\substack{\dot{x} = f(x,y,u,t) \\ u \in U_B}} \widehat{J}(x, y, u) \\ &= \inf_{\substack{\dot{x} = f(x,y,u,t) \\ u \in U_B}} \lim_{k \to \infty} J_{\mu_k}(x, y, u, \nu_k) \end{aligned} \tag{4.15}$$

[2]with respect to $\mathcal{C}^1[t_0, t_f]$-norm

The first equality holds by definition. The second equality holds because if there are feasible x, y, and u such that $g(x, y, u, t) = 0$ for all $t \in [t_0, t_f]$ then $\widehat{J}(x, y, u) = J(x, y, u)$. The third equality holds because in the limit $k \to \infty$ we have $J_{\mu_k} = \widehat{J}(x, y, u)$.

On the other hand, the proposed algorithm performs a sequence of minimizations of (4.5), namely

$$\bar{J} = \lim_{k\to\infty} \inf_{\substack{\dot{x}=f(x,y,u,t) \\ u\in U_B}} J_{\mu_k}(x, y, u, \nu_k) \tag{4.16}$$

For the penalty method to be correct, problem $\mathcal{P}_0$ (4.1) must allow the interchange of the limit and infimum operators in equation (4.15) and (4.16). The following theorem guarantees the validity of this interchange, under some assumptions, and gives the convergence resultant of the penalty method. However before giving the theorem, some general assumptions will be made.

Assumption 2 (Regularity)**.** *To ensure that problem* $\mathcal{P}_0$ (4.1) *and* $\mathcal{P}_{\mathcal{L}}(\mu_k, \nu_k)$ (4.5) *are well-conditioned we assume that*

1. $x : [t_0, t_f] \to \mathbb{R}^{N_x}$ *is a continuously differentiable function,* $y : [t_0, t_f] \to \mathbb{R}^{N_y}$ *and* $\nu_k : [t_0, t_f] \to \mathbb{R}^{N_y}$ *are continuous functions, and the control* $u : [t_0, t_f] \to U_B$ *is a piecewise continuous function such that* $U_B = \{u \in \mathbb{R}^{N_u} | u_L \leq u \leq u_U\}$

2. *L, g, and f are continuously differentiable with respect to all the arguments,*

3. *the matrix of partial derivatives of* $g(x(t), y(t), u(t), t)$ *with respect to* y *has full rank for all* $x(t) \in X$, $y(t) \in Y$, $u(t) \in U_B$, *and* $t \in [t_0, t_f]$,

4. *the sequence* $\{\mu_k\}$ *has the property that* $0 < \mu_k < \mu_{k+1}$ *for all* k, *and* $\mu_k \to \infty$ *as* $k \to \infty$,

5. *the problem* $\mathcal{P}_0$ (4.1) *is solvable.*

Let us define and uniform convergence of a sequence of functions.

Definition 10. *Let* $f_k : [t_0, t_f] \to \mathbb{R}^N$ *be a function for every* $k \in \mathbb{N}$. *The sequence of functions* $\{f_k\}$ *converges uniformly to the limiting function* $f^* : [t_0, t_f] \to \mathbb{R}^N$ *if for every* $\varepsilon > 0$ *there exists a number* $N \in \mathbb{N}$ *such that for all* $t \in [t_0, t_f]$ *and all* $k \geq N$, *we have* $\|f_k(t) - f^*(t)\| < \varepsilon$.

Using the definition of uniform convergence, we can state the first theorem.

Theorem 12. *Let the paths x_k, y_k, and u_k be global minima of the problem $\mathcal{P}_{\mathcal{L}}(\mu_k, \nu_k)$ (Eq. 4.5) at each iteration k. In addition, let $\{\nu_k\}$ be an uniformly convergent sequence.*

Then, under Assumption 2, every limiting function of the sequences $\{x_k\}$, $\{y_k\}$, and $\{u_k\}$ are global minimizers of problem $\mathcal{P}_0$ (4.1) and the sequence $\{J_{\mu_k}(x_k, y_k, u_k, \nu_k)\}$ converges to the optimum objective of $\mathcal{P}_0$.

Proof. Under Assumption 2, the sequences $\{x_k\}$, $\{y_k\}$, and $\{u_k\}$ converges uniformly. Let x^*, y^*, and u^* be limiting functions of the sequences $\{x_k\}$, $\{y_k\}$, and $\{u_k\}$, respectively. We have by definition of x_k, y_k, and u_k that for a given k

$$J_{\mu_k}(x_k, y_k, u_k, \nu_k) \leq J_{\mu_k}(x, y, u, \nu_k) \tag{4.17}$$

for all feasible x, y, and u.

Let J^* denote the optimal value of $\mathcal{P}_0$. We have that

$$J^* = \inf_{x,y,u} \mathcal{P}_0 = \inf_{\substack{x,y,u \\ g(x,y,u,t)=0}} \mathcal{P}_{\mathcal{L}}(\mu_k, \nu_k) \tag{4.18}$$

where the last term implies the minimization of the problem $\mathcal{P}_{\mathcal{L}}$ on x, y, and u with the additional equation $g(x, y, u, t) = 0$. The first equality holds by definition. The second equality holds because $\mathcal{P}_0$ and $\mathcal{P}_{\mathcal{L}}$ are equivalent when the equation $g(x, y, u, t) = 0$ is included.

The inequality (4.17) holds for any x, y, and u, including a minimizer of (4.18). Therefore, we can substitute the optimum value on the right-hand side, and on the left-hand side we substitute $J_{\mu_k}(x_k, y_k, u_k, \nu_k)$ with its definition to obtain

$$\int_{t_0}^{t_f} L(x_k, y_k, u_k, t) + \nu_k^T \left[g(x_k, y_k, u_k, t)\right] + \frac{\mu_k}{2} \left\|g(x_k, y_k, u_k, t)\right\|^2 dt \leq J^* \tag{4.19}$$

Given that the sequence $\{\nu_k\}$ is uniformly convergent, it has a limiting function ν^*. By taking the limit with $k \to \infty$ in the in-

equality (4.19) we obtain

$$\int_{t_0}^{t_f} \left[L(x^*, y^*, u^*, t) + \nu^{*T} g(x^*, y^*, u^*, t)\right] dt + \lim_{k\to\infty} \frac{\mu_k}{2} \int_{t_0}^{t_f} \|g(x_k, y_k, u_k, t)\|^2 \, dt \leq J^* \quad (4.20)$$

Since $\|g(x_k, y_k, u_k, t)\|^2 \geq 0$ and $\mu_k \to \infty$, it follows that we must have $g(x_k, y_k, u_k, t) \to 0$ and

$$g(x^*, y^*, u^*, t) = 0 \qquad \forall t \in [0, t_f] \quad (4.21)$$

otherwise the limit on the left-hand side of (4.20) would be $+\infty$ which does not hold since J^* is finite. Therefore,

$$J(x^*, y^*, u^*) = \int_{t_0}^{t_f} L(x^*, y^*, u^*, t) \, dt \leq J^* \quad (4.22)$$

Any solution to the problem $\mathcal{P}_\mathcal{L}$ satisfies all of the constraints of $\mathcal{P}_0$ except the relaxed algebraic equations. However (4.21) gives that the limiting functions x^*, y^*, and u^* do satisfy the algebraic equation. By definition, J^* is less or equal to any feasible point for the problem $\mathcal{P}_0$, therefore we have

$$J^* \leq J(x^*, y^*, u^*) \quad (4.23)$$

Using (4.23) and (4.22), we obtain

$$J^* \leq J(x^*, y^*, u^*) \leq J^* \quad (4.24)$$

from which we conclude that

$$J^* = J(x^*, y^*, u^*) \quad (4.25)$$

which proves that the limiting functions x^*, y^*, and u^* are the global minimizers for problem $\mathcal{P}_0$.

From (4.25) we have that the sequence

$$\left\{ \int_{t_0}^{t_f} L(x_k, y_k, u_k, t) \, dt \right\} \to J^* \quad (4.26)$$

Therefore for all $\varepsilon/2 > 0$ there exists an $N_1 \in \mathbb{N}$ such that for all $k \geq N_1$,

$$\left| \int_{t_0}^{t_f} L(x_k, y_k, u_k, t) \, dt - J^* \right| < \frac{\varepsilon}{2} \quad (4.27)$$

Similarly, having $g(x_k, y_k, u_k, t) \to 0$, $\mu_k \|g(x_k, y_k, u_k, t)\|^2 \to 0$, and ν_k bounded. Then, for all $\varepsilon/2 > 0$ there exists an $N_2 \in \mathbb{N}$ such that for all $k \geq N_2$,

$$\left| \int_{t_0}^{t_f} \nu_k^T \left[g(x_k, y_k, u_k, t) \right] + \frac{\mu_k}{2} \left\| g(x_k, y_k, u_k, t) \right\|^2 \, dt \right| < \frac{\varepsilon}{2} \quad (4.28)$$

Therefore, for all $\varepsilon > 0$, there exists an $N_3 \in \mathbb{N}$ with $N_3 = \max\{N_1, N_2\}$ such that for all $k \geq N_3$,

$$\left| \int_{t_0}^{t_f} L(x_k, y_k, u_k, t) \, dt - J^* \right| + \left| \int_{t_0}^{t_f} \nu_k^T \left[g(x_k, y_k, u_k, t) \right] + \frac{\mu_k}{2} \left\| g(x_k, y_k, u_k, t) \right\|^2 \, dt \right| < \varepsilon \quad (4.29)$$

Using the subadditivity property,

$$|J_{\mu_k}(x_k, y_k, u_k, \nu_k) - J^*| \leq \left| \int_{t_0}^{t_f} L(x_k, y_k, u_k, t) \, dt - J^* \right| + \left| \int_{t_0}^{t_f} \nu_k^T \left[g(x_k, y_k, u_k, t) \right] + \frac{\mu_k}{2} \left\| g(x_k, y_k, u_k, t) \right\|^2 \, dt \right| \quad (4.30)$$

which results in

$$|J_{\mu_k}(x_k, y_k, u_k, \nu_k) - J^*| < \varepsilon \quad (4.31)$$

and therefore we conclude that $\{J_{\mu_k}(x_k, y_k, u_k, \nu_k)\}$ converges to J^*. □

Theorem 12 assumes that the original $\mathcal{P}_0$ and the augmented problems $\mathcal{P}_\mathcal{L}$ are solved to global optimality. The next theorem shows that the sequence of problems $\mathcal{P}_\mathcal{L}$ that reach a local minimum converges to a local minimum of problem $\mathcal{P}_0$. To make the notation more compact we adopt the notation of tuples, represented by $\langle \cdot \rangle$. Therefore, the solution of $\mathcal{P}_\mathcal{L}(\nu_k, \mu_k)$ is given by $\langle x_k, y_k, u_k \rangle$ and a limiting function obtained by the sequence is $\langle x^*, y^*, u^* \rangle$.

Definition 11. *Let $\mathcal{V}$ be a function space, then a nonempty set $\mathcal{V}^* \subset \mathcal{V}$ is said to be an isolated set of local minima of problem $\mathcal{P}_0$ if each trajectory $v^* \in \mathcal{V}^*$ is a local minimum of problem $\mathcal{P}_0$ and, for some $\varepsilon > 0$, the set*

$$\mathcal{V}_\varepsilon^* = \{v \in \mathcal{V} : \|v - v^*\| \leq \varepsilon \text{ for some } v^* \in \mathcal{V}^*\} \quad (4.32)$$

contains no local minima of problem $\mathcal{P}_0$ but the functions of $\mathcal{V}^*$, where the norm is the $\mathcal{C}^1[t_0, t_f]$ norm.

A strict local minimum may be viewed as an isolated set of local minima consisting of a single path.

Theorem 13. *Suppose that regularity Assumption 2 holds, and that $\mathcal{V}^*$ is an compact and isolated set of local minima of problem $\mathcal{P}_0$. Then there exists a subsequence $\{\langle x_k, y_k, u_k\rangle\}_K$ converging to a limiting function $\langle x^*, y^*, u^*\rangle \in \mathcal{V}^*$ such that $\langle x_k, y_k, u_k\rangle$ is a local minimizer for the problem $\mathcal{P}_\mathcal{L}$ for each k. Furthermore, if $\mathcal{V}^*$ consists of a single point $\langle x^*, y^*, u^*\rangle$, then there exists a sequence $\{\langle x_k, y_k, u_k\rangle\}$ such that $\{\langle x_k, y_k, u_k\rangle\} \to \langle x^*, y^*, u^*\rangle$.*

Proof. Consider the set

$$\mathcal{V}^*_{\tilde{\varepsilon}} = \{v \in \mathcal{V} : \|v - v^*\| \leq \tilde{\varepsilon} \text{ for some } v^* \in \mathcal{V}^*\} \tag{4.33}$$

where $\mathcal{V}$ is the set of feasible functions of $\mathcal{P}_\mathcal{L}$, $0 < \tilde{\varepsilon} < \varepsilon$, and ε is as in (4.32). The compactness of $\mathcal{V}^*$ implies that $\mathcal{V}^*_{\tilde{\varepsilon}}$ is also compact, and hence the problem

$$\min_{x,y,u} J_{\mu_k} = \int_{t_0}^{t_f} \mathcal{L}_{\mu_k}(x, y, u, \nu_k, t)\, dt \tag{4.34a}$$

$$\text{s.t.: } \dot{x} = f(x, y, u, t) \qquad \forall t \in [t_0, t_f] \tag{4.34b}$$

$$u(t) \in U_B \qquad \forall t \in [t_0, t_f] \tag{4.34c}$$

$$x(0) = x_0 \tag{4.34d}$$

$$\langle x, y, u\rangle \in \mathcal{V}^*_{\tilde{\varepsilon}} \tag{4.34e}$$

has a global minimum $\langle x_k, y_k, u_k\rangle \in \mathcal{V}^*_{\tilde{\varepsilon}}$. By Theorem 12, every limiting function $\langle x^*, y^*, u^*\rangle$ of the sequence $\{\langle x_k, y_k, u_k\rangle\}$ is a global minimum of the problem

$$\min_{x,y,u} J = \int_{t_0}^{t_f} L(x, y, u, t)\, dt \tag{4.35a}$$

$$\text{s.t.: } \dot{x} = f(x, y, u, t) \qquad \forall t \in [t_0, t_f] \tag{4.35b}$$

$$g(x, y, u, t) = 0 \qquad \forall t \in [t_0, t_f] \tag{4.35c}$$

$$u(t) \in U_B \qquad \forall t \in [t_0, t_f] \tag{4.35d}$$

$$x(0) = x_0 \tag{4.35e}$$

$$\langle x, y, u\rangle \in \mathcal{V}^*_{\tilde{\varepsilon}} \tag{4.35f}$$

Furthermore, each global minimum of the problem above must belong to $\mathcal{V}^*$ by the definition of $\mathcal{V}^*_{\tilde{\varepsilon}}$. Therefore there is a subsequence $\{\langle x_k, y_k, u_k\rangle\}_K$ converging to $\langle x^*, y^*, u^*\rangle \in \mathcal{V}^*$, such that $K = \{k :$

$\|\langle x_k, y_k, u_k\rangle - \langle x^*, y^*, u^*\rangle\| < \tilde{\varepsilon}\}$ for the given $\langle x^*, y^*, u^*\rangle \in \mathcal{V}^*$. If $\mathcal{V}^*$ contains only one local optimal path then all the subsequences will lead to this local optimal path once it is close to it, meaning that there exists an $N \in \mathbb{N}$ such that for all $k \geq N$

$$\|\langle x_k, y_k, u_k\rangle - \langle x^*, y^*, u^*\rangle\| < \tilde{\varepsilon} \tag{4.36}$$

Since $\langle x^*, y^*, u^*\rangle$ is unique, one has $\{\langle x_k, y_k, u_k\rangle\} \to \langle x^*, y^*, u^*\rangle$. □

Both Theorems 12 and 13 assume implicitly that a method can find a local or global minimum for the augmented Lagrange problem at each iteration. On the other hand, numerical methods terminate when optimality conditions are within a specified tolerance, but not necessarily zero. In particular, given the problem

$$\min_{x,y,u} J_{\mu_k} = \int_{t_0}^{t_f} \mathcal{L}_{\mu_k}(x, y, u, \nu_k, t)\, dt \tag{4.37a}$$

$$\text{s.t.: } \dot{x} = f(x, y, u, t) \qquad \forall t \in [t_0, t_f] \tag{4.37b}$$

$$u(t) \in U_B \qquad \forall t \in [t_0, t_f] \tag{4.37c}$$

$$x(0) = x_0 \tag{4.37d}$$

we expect numerical methods to terminate when the optimality conditions of Theorem 7 are almost satisfied, that is, for a small scalar $\varepsilon_k > 0$ the conditions are

$$\|f(x_k, y_k, u_k, t) - \dot{x}\| < \varepsilon_k, \tag{4.38a}$$

$$\left\| \frac{\partial \mathcal{L}_{\mu_k}}{\partial x}(x_k, y_k, u_k, \nu_k, t)^T + \frac{\partial f}{\partial x}(x_k, y_k, u_k, t)^T \lambda_k + \dot{\lambda}_k \right\| < \varepsilon_k, \tag{4.38b}$$

$$\left\| u_k(t) - \arg \inf_{\mathrm{u} \in U_B} H(x_k(t), \lambda_k(t), y_k, \mathrm{u}, t) \right\| < \varepsilon_k \tag{4.38c}$$

$$\left\| \frac{\partial \mathcal{L}_{\mu_k}}{\partial y}(x_k, y_k, u_k, \nu_k, t)^T + \frac{\partial f}{\partial y}(x_k, y_k, u_k, t)^T \lambda_k \right\| < \varepsilon_k. \tag{4.38d}$$

such that $\varepsilon_k \to 0$ as $k \to \infty$ and λ_k is the costate at the k-th iteration. The norm used in (4.38) is the $\mathcal{C}[t_0, t_f]$ norm of functions (Section 3.1.1), which is given by

$$\|f\| = \max_{t \in [t_0, t_f]} \|f(t)\|_\infty \tag{4.39}$$

where $f : [t_0, t_f] \to \mathbb{R}^d$ is a continuous function in the d dimensional space. This norm gives the maximum absolute value between all components of f during the interval $[t_0, t_f]$.

The following theorem shows that a sequence of suboptimal solutions of the auxiliary problem $\mathcal{P}_{\mathcal{L}}$ is able to converge to the optimal value. The theorem also justifies an update rule for the multiplier approximation ν_k.

Theorem 14. *Suppose that Assumption 2 holds and let $\langle x_k, y_k, u_k \rangle$ be a suboptimal solution obtained for $\mathcal{P}_{\mathcal{L}}(\mu_k, \nu_k)$ such that the violation of the optimality conditions are given by* (4.38)*, for which the following inequality is fundamental*

$$\left\| \frac{\partial \mathcal{L}_{\mu_k}}{\partial y}(x_k, y_k, u_k, \nu_k, t) + \lambda_k^T \frac{\partial f}{\partial y}(x_k, y_k, u_k, t) \right\| \leq \varepsilon_k \tag{4.40}$$

where $0 \leq \varepsilon_k$, and $\varepsilon_k \to 0$ as $k \to \infty$, $\{\nu_k\}$ is a uniform convergent sequence, and λ_k is the costate at the k-th algorithm iteration. Assume that a subsequence $\{\langle x_k, y_k, u_k \rangle\}_K$ converges to $\langle x^*, y^*, u^* \rangle$ such that $\frac{\partial g}{\partial y}(x^*, y^*, u^*, t)$ has full rank and is bounded for all $t \in [t_0, t_f]$.

Then the sequence $\{\nu_k + \mu_k g(x_k, y_k, u_k, t)\}_K$ converges uniformly to $\widetilde{\nu}^*$, such that the necessary optimality condition of problem $\mathcal{P}_{\mathcal{L}}$ with respect to y

$$\frac{\partial L}{\partial y}(x^*, y^*, u^*, t) + \lambda^{*T} \frac{\partial f}{\partial y}(x^*, y^*, u^*, t) + \widetilde{\nu}^{*T} \frac{\partial g}{\partial y}(x^*, y^*, u^*, t) = 0 \tag{4.41}$$

and with respect to x, u, and λ are

$$-\dot{\lambda}^* = \frac{\partial L}{\partial x}(x^*, y^*, u^*, t)^T + \frac{\partial f}{\partial x}(x^*, y^*, u^*, t)^T \lambda^* + \frac{\partial g}{\partial x}(x^*, y^*, u^*, t)^T \widetilde{\nu}^* \tag{4.42a}$$

$$u^*(t) = \arg \inf_{\mathrm{u} \in U_B} H(x^*(t), \lambda^*(t), y^*, \mathrm{u}, t) \tag{4.42b}$$

$$\dot{x}^* = f(x^*, y^*, u^*, t) \tag{4.42c}$$

Proof. Define for all k

$$\widetilde{\nu}_k = \nu_k + \mu_k g(x_k, y_k, u_k, t) \tag{4.43}$$

Then

$$\frac{\partial \mathcal{L}_{\mu_k}}{\partial y}(x_k, y_k, u_k, \nu_k, t) = \frac{\partial L}{\partial y}(x_k, y_k, u_k, t) + [\nu_k + \mu_k g(x_k, y_k, u_k, t)]^T \frac{\partial g}{\partial y}(x_k, y_k, u_k, t) \quad (4.44)$$

replacing with $\widetilde{\nu}_k$,

$$\frac{\partial \mathcal{L}_{\mu_k}}{\partial y}(x_k, y_k, u_k, \nu_k, t) = \frac{\partial L}{\partial y}(x_k, y_k, u_k, t) + \widetilde{\nu}_k^T \frac{\partial g}{\partial y}(x_k, y_k, u_k, t) \quad (4.45)$$

Since $\frac{\partial g}{\partial y}$ is invertible, we can derive the following expression for $\widetilde{\nu}_k$,

$$\widetilde{\nu}_k = \left[\frac{\partial g}{\partial y}(x_k, y_k, u_k, t)^T\right]^{-1} \left[\frac{\partial \mathcal{L}_{\mu_k}}{\partial y}(x_k, y_k, u_k, \nu_k, t)^T - \frac{\partial L}{\partial y}(x_k, y_k, u_k, t)^T\right] \quad (4.46)$$

From (4.46) we can say that there exists an F such that

$$\widetilde{\nu}_k = F(x_k, y_k, u_k, \nu_k) \quad (4.47)$$

and since all the functions in (4.46) are continuous, we know that F is continuous. Given that a subsequence $\{\langle x_k, y_k, u_k\rangle\}_K$ converges to a $\langle x^*, y^*, u^*\rangle$ and $\{\nu_k\}$ converges to ν^*, we can invoke Theorem 18 from Appendix A, to conclude that

$$\{\widetilde{\nu}_k = F(x_k, y_k, u_k, \nu_k)\}_K \to F(x^*, y^*, u^*, \nu^*) = \widetilde{\nu}^* \quad (4.48)$$

where $\widetilde{\nu}^*$ is given by

$$\widetilde{\nu}^* = \left[\frac{\partial g}{\partial y}(x^*, y^*, u^*, t)^T\right]^{-1} \left[\frac{\partial \mathcal{L}_{\mu_k}}{\partial y}(x^*, y^*, u^*, \nu^*, t)^T - \frac{\partial L}{\partial y}(x^*, y^*, u^*, t)^T\right] \quad (4.49)$$

Considering the optimality conditions for y, we have

$$\left\| \frac{\partial \mathcal{L}_{\mu_k}}{\partial y}(x_k, y_k, u_k, \nu_k, t) + \lambda_k^T \frac{\partial f}{\partial y}(x_k, y_k, u_k, t) \right\| \leq \varepsilon_k \quad (4.50)$$

if we take the limit $k \to \infty$, we obtain

$$\frac{\partial \mathcal{L}_{\mu_k}}{\partial y}(x_k, y_k, u_k, \nu_k, t) = -\lambda_k^T \frac{\partial f}{\partial y}(x_k, y_k, u_k, t) \tag{4.51}$$

which can be substituted into (4.49) to obtain

$$\widetilde{\nu}^* = \left[\frac{\partial g}{\partial y}(x^*, y^*, u^*, t)^T\right]^{-1} \left[-\frac{\partial L}{\partial y}(x^*, y^*, u^*, t)^T - \lambda_k^T \frac{\partial f}{\partial y}(x_k, y_k, u_k, t)\right] \tag{4.52}$$

which can be related to necessary conditions of the original OCP $\mathcal{P}_0$ (4.1).

If the sequence $\{\nu_k\}$ is bounded and $\{\nu_k + \mu_k g(x_k, y_k, u_k, t)\}_K \to \widetilde{\nu}^*$, it follows that $\{\mu_k g(x_k, y_k, u_k, t)\}_K$ is bounded. Given that $\mu_k \to \infty$ we must have $g(x_k, y_k, u_k, t) \to 0$ with $g(x^*, y^*, u^*, t) = 0$ for all t.

We can replace with (4.45) into the necessary conditions to obtain

$$\left\| \frac{\partial L}{\partial y}(x_k, y_k, u_k, t) + \widetilde{\nu}_k^T \frac{\partial g}{\partial y}(x_k, y_k, u_k, t) + \lambda_k^T \frac{\partial f}{\partial y}(x_k, y_k, u_k, t) \right\| \leq \varepsilon_k \tag{4.53}$$

by making $k \to \infty$, we obtain

$$\frac{\partial F}{\partial y}(x^*, y^*, u^*, t) + \widetilde{\nu}^{*T} \frac{\partial g}{\partial y}(x^*, y^*, u^*, t) + \lambda^{*T} \frac{\partial f}{\partial y}(x^*, y^*, u^*, t) = 0 \tag{4.54}$$

The same approach can be used to obtain the conditions for x, u, and λ.

□

Notice that so far we have said nothing with respect to $\{\nu_k\}$ except that it is a uniformly convergent series. In the following we define an iteration rule.

Corollary 3. *By defining $\nu_{k+1} = \nu_k + \mu_k g(x_k, y_k, u_k, t)$ we have that $\{\nu_k\} \to \widetilde{\nu}^*$ and $\{\mu_k g(x_k, y_k, u_k, t)\} \to 0$.*

Proof. Theorem 14 gives that for any uniformly convergent sequence $\{\nu_k\}$, we have $\{\nu_k + \mu_k g(x_k, y_k, u_k, t)\} \rightarrow \widetilde{\nu}^*$. Therefore, we can define $\nu_{k+1} = \nu_k + \mu_k g(x_k, y_k, u_k, t)$, which makes the sequence become $\{\nu_{k+1}\} \rightarrow \widetilde{\nu}^*$. For $k+1$ the sequence $\{\nu_{k+1} + \mu_{k+1} g(x_{k+1}, y_{k+1}, u_{k+1}, t)\}$ also converge to $\widetilde{\nu}^*$, therefore $\{\mu_{k+1} g(x_{k+1}, y_{k+1}, u_{k+1}, t)\}$ must go to 0. □

4.4 MULTIPLIER INTERPOLATION

The proposed algorithm depends on the penalization parameter μ_k and multiplier approximation ν_k to formulate the auxiliary problem $\mathcal{P}_{\mathcal{L}}(\mu_k, \nu_k)$. While the penalization parameter is a scalar and easy to store in a computational platform, the multiplier is a function that can assume any form. To overcome this implementational problem, the multiplier is approximated by a parametrized function, which can be easily stored in a computer. A polynomial approximation is particularly interesting because, in general, these approximation are easy to evaluate and update. In addition, the Stone–
Weierstrass theorem establishes that for every continuous function there is a polynomial that is a good enough approximation.

Theorem 15 (Stone–Weierstrass Theorem [25])**.** *Suppose f is a continuous real-valued function defined on the real interval $[t_0, t_f]$. For every $\varepsilon > 0$, there exists a polynomial $p(t)$ such that for all t in $[t_0, t_f]$, we have $|f(t) - p(t)| < \varepsilon$.*

Although Stone–Weierstrass theorem ensures the existence of a polynomial that makes the approximation error $\varepsilon \rightarrow 0$ as the polynomial order increases, the theorem does not specify which polynomial has such a property. In addition, it has been proved that there is no general formula for a polynomial that satisfies Stone– Weierstrass Theorem [26].

Chebyshev polynomials usually yield a good approximation [26], and splines are very flexible for approximation of general functions. However in this work, for convenience, a collocation type approach is chosen, where the approximation is done by a family of Lagrange polynomials with Legendre-Gauss-Radau points, being the same used for the collocation method, Section 2.6.

In order to use the collocation scheme for approximating the multiplier, let us split the integration interval $[t_0, t_f]$ into N subin-

tervals. Then at each subinterval T_i, for $i = 1, \ldots, N$,

$$\nu_k = \sum_{j=0}^{K} \widehat{\ell}_j(\tau)\nu_{k,ij} \tag{4.55}$$

where $\widehat{\ell}_j$ is a Lagrange polynomial such that

$$\widehat{\ell}_j(\tau) = \prod_{k=1,\neq j}^{K} \frac{(\tau - \tau_k)}{(\tau_j - \tau_k)} \tag{4.56}$$

where τ is the normalized time in the interval T_i, τ_i and τ_j are the collocation points, and K is the order of the interpolation polynomial.

Let x_k, u_k, and y_k, be the solution to $\mathcal{P}_{\mathcal{L}}(\mu_k, \nu_k)$ at iteration k of the Algorithm 1. Then the update rule for the multiplier given in Line 3 of the Algorithm 1 can be replaced by

$$\nu_{k+1,ij} = \nu_{k,ij} + \mu_k g(x_k(\tau_j), y_k(\tau_j), u_k(\tau_j), t_{ij}) \tag{4.57}$$

for $i = 1, \ldots, N$, and $j = 1, \ldots, K$, where t_{ij} is the time at the normalized time τ_j in the subinterval T_i.

When using multiple shooting methods, for solve the auxiliary problem $\mathcal{P}_{\mathcal{L}}$, the collocation points and the degree of the polynomial, are parameters that have to be chosen. Notice that using a refined approximation does not burden the optimization process other than with the cost of evaluating the approximation function. However when using collocation methods for solving the OCP $\mathcal{P}_{\mathcal{L}}$, it does not make sense to use a polynomial approximation with different degree and collocation points from those used for solving the OCP.

In the future, other approximation could be studied, for instance B-splines, Chebyshev polynomials, or a different approach for the Lagrangian polynomial, as the barycentric Lagrangian polynomial [27].

4.5 BOUNDED ALGEBRAIC AND STATE VARIABLES

In the previous sections, it has been shown that the proposed algorithm is able to solve the OCP in the form $\mathcal{P}_0$ (4.1), which represents a wide range of applications. On the other hand, a much wider range of application could be developed if we could handle

OCPs in the form $\mathcal{P}_0$ but with bounded state and algebraic variables. That is, the OCP with the form

$$\begin{aligned}
\mathcal{P}_{0,B}: \qquad & \min \; J = \int_{t_0}^{t_f} L(x,y,u,t)\,dt && \text{(4.58a)}\\
& \text{s.t.: } \dot{x} = f(x,y,u,t) && \text{(4.58b)}\\
& \qquad g(x,y,u,t) = 0 && \text{(4.58c)}\\
& \qquad x(0) = x_0 && \text{(4.58d)}\\
& \qquad x(t) \in X_B,\; y(t) \in Y_B,\; u(t) \in U_B && \text{(4.58e)}\\
& \qquad t \in [t_0, t_f] && \text{(4.58f)}
\end{aligned}$$

where

$$\begin{aligned}
X_B &= \{\mathrm{x} \in X \,|\, x_L \leq \mathrm{x} \leq x_U\} && \text{(4.59a)}\\
Y_B &= \{\mathrm{y} \in Y \,|\, y_L \leq \mathrm{y} \leq y_U\} && \text{(4.59b)}\\
U_B &= \{\mathrm{u} \in U \,|\, u_L \leq \mathrm{u} \leq u_U\} && \text{(4.59c)}
\end{aligned}$$

As said before, the auxiliary OCP $\mathcal{P}_\mathcal{L}$ can be put in a form of an OCP of ODE if we consider an extended control variable $\widehat{u} = [u, y]$. Notice that if we define the domain of $\widehat{u}$ such that $\widehat{U} = U_B \times Y_B$, then the Pontryagin's minimum principle for OCP of ODE, Theorem 7, can be applied to obtain a solution that satisfies $y(t) \in Y_B$.

The first two theorems developed for Algorithm 1, Theorems 12 and 13, do not impose any condition on y other the optimality with respect to $\mathcal{P}_\mathcal{L}$. On the contrary, under the assumption that there exists a feasible y such that $y(t) \in Y_B$ and that satisfies $g(x,y,u,t) = 0$, Theorem 12 shows that the sequence of solutions obtained by the algorithm will converge to a global minimizer $y^*(t) \in Y_B$ (local minimizer, in the case of Theorem 13).

This property can be extended to state variables by including in the problem $\mathcal{P}_{0,B}$ (4.58) an algebraic variable $y_x : [t_0, t_f] \to \mathbb{R}^{N_x}$ such the respective algebraic equations are

$$y_x = x \tag{4.60}$$

with $y_x(t) \in X_B$, and disregarding that $x(t) \in X_B$. In this way, the auxiliary OCP solved by the proposed algorithm has the extended control $\widehat{u} = [u, y, y_x] \in \widehat{U}$ with $\widehat{U} = U_B \times Y_B \times X_B$. Therefore, if there is a feasible $\widehat{u}_k \in \widehat{U}$ that satisfies $g(x_k, y_k, u_k, t) = 0$ and $y_{x,k} = x_k$, Theorem 12 ensures that the sequence of solutions given

by the algorithm will converge to a minimizer $\widehat{u}^*$ such that $u^*(t) \in U_B$, $y^* \in Y_B$, and $y_x^* \in X_B$, $g(x^*, y^*, u^*, t) = 0$ and $y_x^* = x^*$.

This property allows us to use indirect methods to solve OCPs with bounded state, algebraic, and control variables, that are in the form of $\mathcal{P}_{0,B}$ (4.58). Notice that with the theory presented in Chapter 3, this task is not possible. Although there exists an extension of the presented optimal control theory to handle such constraints, they are of difficult implementation and usually require hand-tailored equations for each problem. On the other hand, the proposed algorithm allows us to solve $\mathcal{P}_{0,B}$ (4.58) with a straight forward approach. In particular, because at each algorithm iteration the optimal extended control can be obtained with Pontryagin's minimum principle (Theorem 7), that is

$$\widehat{u}_k(t) = \arg \inf_{\hat{\mathrm{u}} \in \widehat{U}} H(x_k(t), \lambda_k(t), \hat{\mathrm{u}}, t). \tag{4.61a}$$

Cases with and without bound constraints will be demonstrated in the following sections, where experiments with the Van der Pol oscillator are performed.

In the next section, the Van der Pol oscillator will be used to confirm the properties of the algorithm, including problems with bound constraint.

4.6 APPLICATION: VAN DER POL OSCILLATOR

To demonstrate the details of implementation and the results obtained with the algorithm, the Van der Pol oscillator, that was used to illustrate the direct and indirect mehtods, Section 3.6 and 3.7, will also be used here. At the end, an analysis of the obtained results and the computational cost is performed.

4.6.1 Problem Formulation

The system description of the Van der Pol oscillator was presented in Section 3.6.1. However to facilitate the reading, the OCP

of the DAE system is restated here,

$$\mathcal{P}^V_{DAE}: \quad \min_{x,y,u} \; J = \int_{t_0}^{t_f} \left[x_1^2 + x_2^2 + u^2\right] \, dt \tag{4.62a}$$

$$\text{s.t.:} \quad \dot{x}_1 = y + u \tag{4.62b}$$

$$\dot{x}_2 = x_1 \tag{4.62c}$$

$$y = (1 - x_2^2)x_1 - x_2 \tag{4.62d}$$

$$x(0) = x_0, \qquad t \in [t_0, t_f] \tag{4.62e}$$

where $x_0 = [0, 1]$, $t_0 = 0$, and $t_f = 5$.

To put $\mathcal{P}^V_{DAE}$ in the form of (4.1), the functions f, g, and L are

$$f = \begin{bmatrix} y + u \\ x_1 \end{bmatrix} \tag{4.63a}$$

$$g = \left[(1 - x_2^2)x_1 - x_2 - y\right] \tag{4.63b}$$

$$L = \left[x_1^2 + x_2^2 + u^2\right] \tag{4.63c}$$

To investigate the properties of the proposed algorithm, three cases are considered for the optimal control problems:

- Case 1, the OCP $\mathcal{P}^V_{DAE}$ is solved as stated in (4.62).
- Case 2, the OCP $\mathcal{P}^V_{DAE}$ (4.62) is solved with the control variable bounded by $-0.3 \leq u(t) \leq 1$, that is $u_L = -0.3$ and $u_U = 1$.
- Case 3, solves $\mathcal{P}^V_{DAE}$ (4.62) subject to the bound constraints $-0.3 \leq u(t) \leq 1$ on the control variables and the constraint $-0.4 \leq x_1(t)$ on the state x_1, that is $u_L = -0.3$, $u_U = 1$, $x_L = [-0.4 \; -\infty]^T$, and $x_U = [\infty \; \infty]^T$.

For each of the three cases, the algorithm's underlying OCP will be solved with direct and indirect methods applying the multiple shooting and the collocation method. At the end of this section an analysis is done comparing the results of the algorithm and the traditional approach, obtained in Sections 3.6 and 3.7.

4.6.2 Augmented Lagrange Relaxation

To apply the algorithm for Case 1, let us define the function

$$\mathcal{L}_{\mu}^{V} = \left(x_1^2 + x_2^2 + u^2\right) + \nu\left[(1-x_2^2)x_1 - x_2 - y\right] + \frac{\mu}{2}\left\|(1-x_2^2)x_1 - x_2 - y\right\|^2 \quad (4.64)$$

which allows us to formulate the auxiliary problem

$$\begin{aligned}
\mathcal{P}_{\mathcal{L}}^{V}(\mu_k, \nu_k): \qquad \min\ & J_{\mu_k} = \int_{t_0}^{t_f} \mathcal{L}_{\mu_k}^{V}(x, y, u, t)\, dt && (4.65a)\\
\text{s.t.:}\ & \dot{x}_1 = y + u && (4.65b)\\
& \dot{x}_2 = x_1 && (4.65c)\\
& x(0) = x_0 && (4.65d)\\
& x(t) \in X,\ y(t) \in Y,\ u(t) \in U && (4.65e)\\
& t \in [t_0, t_f] && (4.65f)
\end{aligned}$$

and the update of ν_k and μ_k are done by

$$\begin{aligned}
\nu_{k+1} &= \nu_k + \mu_k\left[(1-x_2^2)x_1 - x_2 - y\right] && (4.66a)\\
\mu_{k+1} &= \beta\mu_k && (4.66b)
\end{aligned}$$

with the parameters $\beta = 8$, $\mu_0 = 2$, and $\nu_0 = 0$ for all $t \in [t_0, t_f]$. The number of subintervals $N = 10$ and a 3-th order polynomial was used to describe ν_k.

For Case 2, the auxiliary problem is the same from Case 1, just changing from $u(t) \in U$ to $u(t) \in U_B$.

For Case 3, the direct and indirect method have different auxiliary problems. The direct methods are able to handle the bound constraints and therefore the auxiliary problem is equal to Case 1 (4.65), with the inclusion of the bounds $x(t) \in X_B$ and $u(t) \in U_B$. On the other hand, the indirect methods require the inclusion of additional algebraic variables as discussed in Section 4.5, the formulation of this problem is given in the following section.

4.6.3 Solution with Indirect Methods

Let us consider the solution of each case individually:

1. **Case 1 – Indirect Multiple Shooting:** For this case the auxiliary OCP to be solved is given by problem $\mathcal{P}_{\mathcal{L}}^{V}$ (4.65), the

Hamiltonian function for the k-th algorithm iteration is given by,

$$H_k^V = \left(x_1^2 + x_2^2 + u^2\right) + \nu_k \left[(1 - x_2^2)x_1 - x_2 - y\right] + \frac{\mu_k}{2} \left\|(1 - x_2^2)x_1 - x_2 - y\right\|^2 + \lambda_1 (y + u) + \lambda_2 x_1 \quad (4.67)$$

Since $\mathcal{P}_{\mathcal{L}}^V$ (4.65) is in the form of OCP of an ODE system we can apply Theorem 6, which is more compactly represented with the Hamiltonian function (3.120). Then the necessary conditions for optimality of the problem $\mathcal{P}_{\mathcal{L}}^V$ (4.65) are

$$\frac{\partial H_k^V}{\partial x_1} = -\dot{\lambda}_1 = 2x_1 + \mu_k \left[(1 - x_2^2)x_1 - x_2 - y\right](1 - x_2^2) + \nu_k(1 - x_2^2) + \lambda_2 \quad (4.68a)$$

$$\frac{\partial H_k^V}{\partial x_2} = -\dot{\lambda}_2 = 2x_2 + \nu_k(-2x_2x_1 - 1) + \mu_k \left[(1 - x_2^2)x_1 - x_2 - y\right](-2x_2x_1 - 1) \quad (4.68b)$$

$$\frac{\partial H_k^V}{\partial y} = -\nu_k - \mu_k \left[(1 - x_2^2)x_1 - x_2 - y\right] + \lambda_1 = 0 \quad (4.68c)$$

$$\frac{\partial H_k^V}{\partial u} = 2u + \lambda_1 = 0 \quad (4.68d)$$

$$\frac{\partial H_k^V}{\partial \lambda_1} = \dot{x}_1 = y + u \quad (4.68e)$$

$$\frac{\partial H_k^V}{\partial \lambda_2} = \dot{x}_2 = x_1 \quad (4.68f)$$

$$x(0) = [1, 0]^T \qquad \lambda(t_f) = 0 \quad (4.68g)$$

From equations (4.68c)-(4.68d), the optimal control rule is

$$y_{opt} = \frac{\nu_k - \lambda_1}{\mu_k} + ((1 - x_2^2)x_1 - x_2) \quad (4.69a)$$

$$u_{opt} = -\frac{\lambda_1}{2} \quad (4.69b)$$

which allows us to draw two conclusions:

a) As $\mu_k \to \infty$ we have $(1 - x_2^2)x_1 - x_2 - y_{opt} \to 0$.
b) As $(1 - x_2^2)x_1 - x_2 - y_{opt} \to 0$ we have $\nu_k \to \lambda_1$, which agrees with the necessary conditions for the original problem (3.253c).

Since u and y are unbounded, we can use the optimal control laws to define u_k and y_k,

$$y_k(t) = y_{opt}(t) \tag{4.70a}$$
$$u_k(t) = u_{opt}(t) \tag{4.70b}$$

Then, the necessary conditions presented in (4.68) yield the following BVP

$$\dot{\lambda}_1 = -2x_1 - \mu_k \left[(1 - x_2^2)x_1 - x_2 - y_{opt}\right](1 - x_2^2) - \nu_k(1 - x_2^2) - \lambda_2 \tag{4.71a}$$
$$\dot{\lambda}_2 = -2x_2 - \nu_k(-2x_2x_1 - 1) - \mu_k \left[(1 - x_2^2)x_1 - x_2 - y_{opt}\right](-2x_2x_1 - 1) \tag{4.71b}$$
$$\dot{x}_1 = y_{opt} + u_{opt}, \qquad \dot{x}_2 = x_1 \tag{4.71c}$$
$$x(0) = [1, 0]^T, \qquad \lambda(t_f) = 0 \tag{4.71d}$$

for which, the free variables u_{opt} and y_{opt} can be substituted by the optimal control law (4.69) to obtain a solution for problem $\mathcal{P}_{\mathcal{L}}^V$ (4.65). This solution can be obtained by two methods, the collocation method or the multiple shooting method.

If we consider the multiple shooting method, the time horizon has to be split into N subintervals, where each subinterval T_i begins at time t_0^i and ends at t_f^i. Further, we define the extended state variable $\widehat{x} = [x_1, x_2, \lambda_1, \lambda_2]$, where $\widehat{x}_0$ and $\widehat{x}_f$ are the boundary conditions. Then, using Definition 4, the BVP (4.71) can be represented by following functions

$$F_{\mathcal{L},\mu_k}^V(\widehat{x}_0^i, \nu_k, T_i) = \left\{ \widehat{x}_f^i \in \mathbb{R}^4 \;\middle|\; \begin{array}{l} \widehat{x} \text{ satisfying (4.71a)-(4.71c)} \\ \left[x_1(t_0^i), x_2(t_0^i), \lambda_1(t_0^i), \lambda_2(t_0^i)\right] = \widehat{x}_0 \\ \widehat{x}_f^i = \left[x_1(t_f^i), x_2(t_f^i), \lambda_1(t_f^i), \lambda_2(t_f^{T_i})\right] \\ t \in T_i = \left[t_0^i, t_f^i\right] \end{array} \right. \tag{4.72a}$$

$$G_{\mathcal{L}}^{V}(\widehat{x}_0, \widehat{x}_f) = \begin{bmatrix} \widehat{x}_{0,1} - x_{0,1} \\ \widehat{x}_{0,2} - x_{0,2} \\ \widehat{x}_{f,3} \\ \widehat{x}_{f,4} \end{bmatrix} = 0 \tag{4.72b}$$

where, for each subinterval T_i, $\widehat{x}_0^i$ and $\widehat{x}_f^i$ are the boundary values. Then the functions presented in (4.72) can be used to formulate a nonlinear system of equations

$$\widehat{x}_f^i = F_{\mathcal{L},\mu_k}^{V}(\widehat{x}_0^i, \nu_k, T_i) \qquad i = 1, \ldots, N \tag{4.73a}$$

$$\widehat{x}_f^{i-1} = \widehat{x}_0^i \qquad i = 2, \ldots, N \tag{4.73b}$$

$$0 = G_{\mathcal{L}}^{V}(\widehat{x}_0^1, \widehat{x}_f^N) \tag{4.73c}$$

At each algorithm iteration, the nonlinear system (4.73) is solved, and the parameters ν_{k+1} and μ_{k+1} are calculated afterwards.

The algorithm obtains a solution within 4 iterations. The objective value obtained was 2.86695. Figure 4.1 shows the profile obtained for the states and control variables.

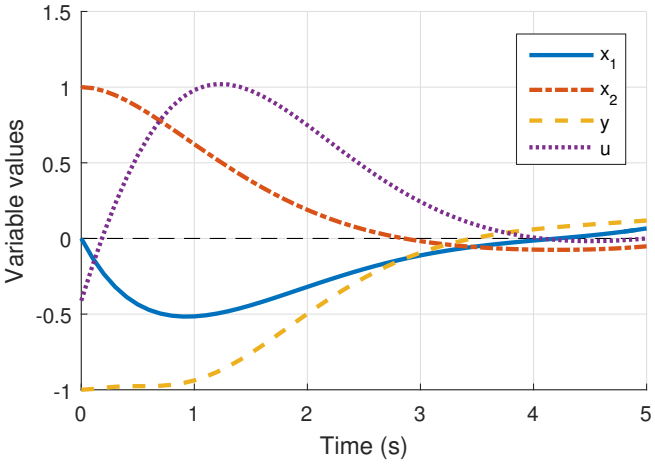

Figure 4.1: Plot of the optimal trajectories for Case 1, following the indirect method solved with the collocation method.

2. **Case 1 – Indirect Collocation Method:** For applying the collocation method, we also consider the extended state $\widehat{x} =$

$[x_1, x_2, \lambda_1, \lambda_2]$, where $\widehat{x}_0$ and $\widehat{x}_f$ are the boundary conditions. We define the function

$$\widehat{f}_{\mu_k} = \begin{bmatrix} y_{opt} + u_{opt}, \\ x_1 \\ -2x_1 - \left[\nu_k + \mu_k \left[(1-x_2^2)x_1 - x_2 - y_{opt}\right]\right](1-x_2^2) - \lambda_2 \\ -2x_2 - \left[\nu_k + \mu_k \left[(1-x_2^2)x_1 - x_2 - y_{opt}\right]\right](-2x_2x_1 - 1) \end{bmatrix} \tag{4.74}$$

If we substitute the optimal control law (4.69) for y_{opt} and u_{opt}, we can define the dynamic with

$$\dot{\widehat{x}} = \widehat{f}_{\mu_k}(\widehat{x}, \nu_k, t) \tag{4.75}$$

because y_{opt} and u_{opt} depends on the state variable, the time variable, and the algorithm parameters.

According to (2.111), we define the basis for the states as

$$\ell_j(\tau) = \prod_{k=0, \neq j}^{K} \frac{(\tau - \tau_k)}{(\tau_j - \tau_k)} \tag{4.76}$$

We split the time interval into N subintervals. In each subinterval, the states are approximated by a K-th order polynomial. From equations (2.112), for all $i = 1, \ldots, N$ with $t \in [t_{i-1}, t_i]$, we have

$$\widehat{x}(t) = \sum_{j=0}^{K} \ell_j(\tau)\widehat{x}_{ij} \tag{4.77}$$

where τ is the normalized time variable in the subinterval T_i, and $\widehat{x}_{ij} \in \mathbb{R}^{N_x}$ is the state in the subinterval T_i at the collocation point j. Since a control law was used to substitute the control variable $\widehat{u}$, the indirect collocation method does not have to approximate the controls.

In order to make the representation (4.77) to correspond to the dynamics of the DAE system, we have to enforce the equations (2.117) and (2.119) for the state variables. Which is accomplished by stating for all $i = 1, \ldots, N$ and $m = 1, \ldots, K$:

$$\sum_{j=0}^{K} \widehat{x}_{ij} \frac{d\ell_j}{d\tau}(\tau_m) = h_i \widehat{f}_{\mu_k}(\widehat{x}_{im}, \nu_k, t_{im}) \tag{4.78a}$$

and, to satisfy the state continuity, for all $i = 1, \ldots, N-1$:

$$\widehat{x}_{i+1,0} = \sum_{j=0}^{K} \ell_j(1)\widehat{x}_{ij} \tag{4.78b}$$

Therefore we can formulate a nonlinear system of equations to be solved at each iteration

$$\sum_{j=0}^{K} \widehat{x}_{ij} \frac{d\ell_j}{d\tau}(\tau_m) = h_i \widehat{f}_{\mu_k}(\widehat{x}_{im}, \nu_k, t_{im}) \qquad \forall i, \forall m \tag{4.79a}$$

$$\widehat{x}_{i+1,0} = \sum_{j=0}^{K} \ell_j(1)\widehat{x}_{ij} \qquad i = 1, \ldots, N-1 \tag{4.79b}$$

$$\begin{bmatrix} \widehat{x}_{ij,1} - x_{0,1} \\ \widehat{x}_{ij,2} - x_{0,2} \end{bmatrix} = 0, \qquad i = 1, j = 1 \tag{4.79c}$$

$$\begin{bmatrix} \widehat{x}_{ij,3} \\ \widehat{x}_{ij,4} \end{bmatrix} = 0, \qquad i = N, j = K \tag{4.79d}$$

Using the number of subintervals $N = 10$ and 3-th order polynomials, the algorithm obtained a solution after 4 iterations. The objective value obtained was 2.86696. The optimal trajectories obtained were identical to those obtained with the indirect multiple shooting, shown in Figure 4.1.

3. **Case 2 – Indirect Multiple Shooting:** For Case 2, the indirect multiple shooting solves the auxiliary problem $\mathcal{P}_{\mathcal{L}}^{V}$ (4.65) with $u(t) \in U_B$. Therefore, the necessary conditions are those presented in (4.68) but the condition on u is replaced by

$$u(t) = \arg\min_{\mathrm{u} \in U_B} H_k^V(x_1(t), x_2(t), \lambda_k(t), y_k(t), \mathrm{u}, t) \tag{4.80}$$

which comes from the Pontryagin's minimum principle, Theorem 7.

We can verify that H_k^V is convex (quadratic) with respect to u, which allows to find an analytic solution for the minimization of H_k^V, hence the development of (3.190) is applicable. Therefore the optimal rule for a bounded control is given by

$$u_{opt}(t) = \begin{cases} u_U, & \text{if } u_U \leq \widetilde{u}_{opt}(t), \\ \widetilde{u}_{opt}(t), & \text{if } u_L < \widetilde{u}_{opt}(t) < u_U, \\ u_L, & \text{if } \widetilde{u}_{opt}(t) \leq u_L \end{cases} \tag{4.81}$$

where $u_L = -0.3$, $u_U = 1$, and $\widetilde{u}_{opt}$ is defined in the same manner of the control law for Case 1 (4.69),

$$\widetilde{u}_{opt} = -\frac{\lambda_1}{2} \tag{4.82}$$

For implementing u_{opt} we can rewrite (4.81) into a form more coding-friendly

$$u_{opt}(t) = \min(u_U, \max(\widetilde{u}_{opt}(t), u_L)) \tag{4.83}$$

The optimal control law for y is the same from Case 1,

$$y_{opt} = \frac{\nu_k - \lambda_1}{\mu_k} + ((1 - x_2^2)x_1 - x_2) \tag{4.84}$$

The BVP to be solved at each iteration k has the same form of the BVP presented for Case 1 (4.71), however control laws (4.81) and (4.82) are implemented. This BVP can be solved either with a multiple shooting or a collocation. For the sake of not being to repetitive the development of the solution with each method is omitted, since the process of generating the nonlinear system of equations for solving the BVP with both methods are the same from Case 1.

Using the multiple shooting to solve the BVP, the algorithm obtained a solution after 5 iterations. The objective value obtained was 2.87972. Figure 4.2 shows the profile obtained for the states and control variables.

4. **Case 2 – Indirect Collocation Method:** For obtaining a solution for Case 2 with the collocation method, we can use the same approach to obtain the optimal rule (4.81) and apply it to the BVP (4.71). Then, the BVP is transformed into a nonlinear system of equations with the form (4.73), as done for Case 1.

   Using this approach, the algorithm produced a solution after 5 iterations. The objective value obtained was 2.87967. The problem solution is very similar to the one obtained with indirect multiple shooting, shown in Figure 4.2.

5. **Case 3 – Indirect Multiple Shooting:** For the indirect methods applied to Case 3, the auxiliary problem has the same form of $\mathcal{P}_{\mathcal{L}}^{V}$ (4.65) but with a different objective, since we have to

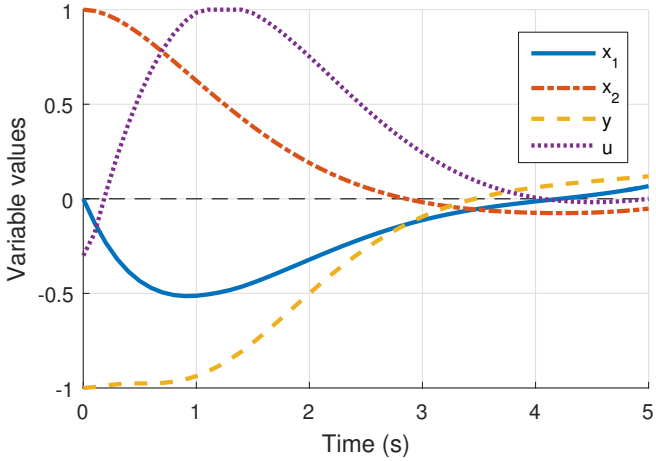

Figure 4.2: Plot of the optimal trajectories for Case 2. Notice that the control signal satisfies the constraint $-0.3 \leq u(t) \leq 1$.

include two algebraic variables y_{x_1} and y_{x_2}, whose equation are given by

$$y_{x_1} = x_1 \tag{4.85a}$$
$$y_{x_2} = x_2 \tag{4.85b}$$

Since there are 3 algebraic equations, the multiplier approximation is given by $\nu = [\nu_1\, \nu_2\, \nu_3]$. Then, the function to be integrated in the objective functional is

$$\begin{aligned}\mathcal{L}_\mu^V = \left(x_1^2 + x_2^2 + u^2\right) &+ \nu_1 \left[(1 - x_2^2)x_1 - x_2 - y\right] \\ &+ \nu_2 \left[x_1 - y_{x_1}\right] + \nu_3 \left[y x_2 - y_{x_2}\right] \\ &+ \frac{\mu}{2} \left\|(1 - x_2^2)x_1 - x_2 - y\right\|^2 \\ &+ \frac{\mu}{2} \left\|x_1 - y_{x_1}\right\|^2 + \frac{\mu}{2} \left\|x_2 - y_{x_2}\right\|^2 \end{aligned} \tag{4.86}$$

which is solved with $x(t) \in X$, $y(t) \in Y$, $y_x(t) \in X_B$, and $u(t) \in U_B$.

Therefore, the Hamiltonian of the problem of Case 3 is

$$H_k^{V_3} = \left(x_1^2 + x_2^2 + u^2\right) + \nu_k \left[(1 - x_2^2)x_1 - x_2 - y\right] + \frac{\mu}{2}\left\|(1 - x_2^2)x_1 - x_2 - y\right\|^2 + \nu_k \left[x_1 - y_{x_1}\right] + \frac{\mu}{2}\left\|x_1 - y_{x_1}\right\|^2 + \nu_k \left[x_2 - y_{x_2}\right] + \frac{\mu}{2}\left\|x_2 - y_{x_2}\right\|^2 + \lambda_1 \left(y + u\right) + \lambda_2 x_1 \quad (4.87)$$

The necessary conditions can be obtained using Theorem 7, in the same form that they were obtained for Case 1 and 2, which led to the necessary conditions (4.68), with the particularity that u and y_x are bounded. Since the proposed algorithm relaxes the algebraic equation making y_x a free variable, we can apply Pontryagin's minimum principle (Theorem 7) to obtain the optimal control rules

$$u_k(t) = \underset{\mathrm{u} \in U_B}{\arg\min}\, H_k^{V_3}(x_{1,k}(t), x_{2,k}(t), \lambda_k(t), y_k(t), y_{x,k}(t), \mathrm{u}, t) \quad (4.88a)$$

$$y_k(t) = \underset{\mathrm{y} \in Y}{\arg\min}\, H_k^{V_3}(x_{1,k}(t), x_{2,k}(t), \lambda_k(t), \mathrm{y}, y_{x,k}(t), u_k(t), t) \quad (4.88b)$$

$$y_{x,k}(t) = \underset{\mathrm{y_x} \in Y_B}{\arg\min}\, H_k^{V_3}(x_{1,k}(t), x_{2,k}(t), y_k, \mathrm{y_x}, u_k(t), t) \quad (4.88c)$$

Since $H_k^{V_3}$ is convex (quadratic) with respect to y and u, we can find an analytic solution for the minimization of $H_k^{V_3}$, hence the development of (3.190) is applicable. Therefore for bounded controls and the additional algebraic variables, the optimal rule is given by

$$u_k(t) = \begin{cases} u_U, & \text{if } u_U \leq \widetilde{u}_{opt}(t), \\ \widetilde{u}_{opt}(t), & \text{if } u_L < \widetilde{u}_{opt}(t) < u_U, \\ u_L, & \text{if } \widetilde{u}_{opt}(t) \leq u_L \end{cases} \quad (4.89a)$$

$$y_k(t) = y_{opt}(t) \quad (4.89b)$$

$$y_{x,k}(t) = \begin{cases} y_U, & \text{if } x_U \leq \widetilde{y}_{x,opt}(t), \\ \widetilde{y}_{x,opt}(t), & \text{if } y_L < \widetilde{y}_{x,opt}(t) < y_U, \\ y_L, & \text{if } \widetilde{y}_{x,opt}(t) \leq x_L \end{cases} \quad (4.89c)$$

where $u_L = -0.3$, $u_U = 1$, $x_L = [-0.4 \ -\infty]^T$, $x_U = [\infty \ \infty]$,

and

$$\widetilde{u}_{opt} = -\frac{\lambda_1}{2} \tag{4.90a}$$

$$y_{opt} = \frac{\nu_k - \lambda_1}{\mu_k} + ((1 - x_2^2)x_1 - x_2) \tag{4.90b}$$

$$\widetilde{y}_{x_1,opt} = \frac{\nu_{2,k}}{\mu_k} + x_1 \tag{4.90c}$$

$$\widetilde{y}_{x_2,opt} = \frac{\nu_{3,k}}{\mu_k} + x_2 \tag{4.90d}$$

The BVP obtained through the application of the necessary conditions for optimality can be solved with multiple shooting and collocation. Again, the development of these are omitted since they are very similar to the procedure of Case 1.

Applying the multiple shooting, the algorithm obtained a solution after 6 iterations. The objective value obtained was 2.9532. Figure 4.3 shows the profile obtained for the states and control variables.

6. **Case 3 – Indirect Collocation method:** To apply the collocation method to the Case 3, we can use the development carried out for the multiple shooting approach to obtain the optimal control rule (4.89). Then the boundary value problem obtained from the optimality conditions with the Hamiltonian (4.87) can be solved by creating a nonlinear system of equations in the form (4.73), in the same manner that was obtained for Case 1.

   Using the indirect collocation method, the solution was obtained after 6 iterations. The objective value obtained was 2.95308. The trajectories obtained with indirect collocation method were very similar to those obtained with the indirect multiple shooting, shown in Figure 4.3.

4.6.4 Solution with Direct Methods

The direct methods will transform the auxiliary OCP into a nonlinear programming problem. The development will be case by case. Two approaches can be used: the multiple shooting and the collocation. In general, the procedure is similar to the applications performed in Section 3.7.

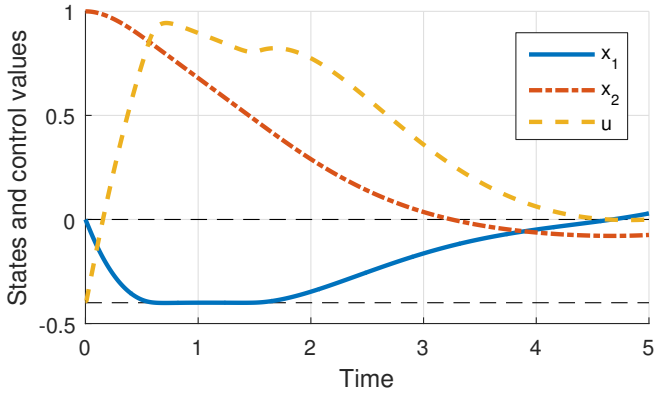

Figure 4.3: Plot of the optimal trajectories for Case 3. Notice that $x1$ satisfy the constraint $-0.4 \leq x_1(t)$.

To use the direct method we need to remove the integral from the objective of the auxiliary OCP $\mathcal{P}_{\mathcal{L}}^{V}$ (4.65), since it is not handled by a NLP solver. With this purpose, let us define a cost state x_c such that

$$\dot{x}_c = \left(x_1^2 + x_2^2 + u^2\right) + \nu\left[(1-x_2^2)x_1 - x_2 - y\right] + \frac{\mu}{2}\left\|(1-x_2^2)x_1 - x_2 - y\right\|^2 \quad (4.91)$$

which has initial condition $x_c(t_0) = 0$. Using Theorem 5, we can include the state x_c into the auxiliary OCP $\mathcal{P}_{\mathcal{L}}^{V}$ (4.65) and adopt the objective $J = x_c(t_f)$, which is can be handled by the direct methods.

- **Case 1 – Direct Multiple Shooting:** For using the multiple shooting method, let us split the integration interval into N subintervals, where each subinterval T_i, with $i = 1, \ldots, N$, begins at t_0^i and ends at t_f^i. Then, for each subinterval T_i, we parametrize the extended control $\widehat{u} = [u, y]$ with the polynomial

$$\widehat{u}(t) = \sum_{j=1}^{K} \widehat{\ell}_j(\tau)\widehat{u}_{ij} \quad (4.92)$$

where $\widehat{u}_{ij} \in \mathbb{R}^{N_u}$, with $i = 1, \ldots, N$ and $j = 1, \ldots, K$ being parameters that define the extended control profile, and $\widehat{\ell}_j$ is defined in (2.121). Let us define the vector of parameters of the subinterval T_i

$$\theta_i = [\widehat{u}_{ij} : j = 1, \ldots, K], \tag{4.93}$$

so that we can refer to the coefficients of the parametrized control in a more compact form.

Let us define the functions

$$F^V_{\mathcal{L},\mu_k}(\widehat{x}^i_0, \theta_i, \nu_k, T_i)$$

$$= \left\{ \widehat{x}_f \in \mathbb{R}^{N_x+1} \;\middle|\; \begin{array}{l} \dot{x}_1 = y + u \\ \dot{x}_2 = x_1 \\ \dot{x}_c = \left(x_1^2 + x_2^2 + u^2\right) \\ \quad +\nu_k \left[(1 - x_2^2)x_1 - x_2 - y\right] \\ \quad +\frac{\mu_k}{2} \left\|(1 - x_2^2)x_1 - x_2 - y\right\|^2 \\ \widehat{u}(t) = \sum_{j=1}^{K} \widehat{\ell}_j(\tau)\widehat{u}_{ij} \\ \left[x_1(t^i_0), x_2(t^i_0), x_c(t^i_0)\right] = \widehat{x}^i_0 \\ \widehat{x}_f = \left[x_1(t^i_f), x_2(t^i_f), x_c(t^i_f)\right] \\ t \in T_i = [t^i_0, t^i_f] \end{array} \right. \tag{4.94a}$$

$$G^V_{DAE}(\widehat{x}_0, \widehat{x}_f) = \begin{bmatrix} \widehat{x}_{0,1} - x_{0,1} \\ \widehat{x}_{0,2} - x_{0,2} \\ \widehat{x}_{0,3} \\ \widehat{x}_{f,3} - J \end{bmatrix} = 0 \tag{4.94b}$$

where $\widehat{x} = [x_1, x_2, x_c]$, and the parameters $x_{0,1}$ and $x_{0,2}$ are the first and second scalars of the vector of initial conditions x_0.

Let $\widehat{x}^i_0$ and $\widehat{x}^i_f$ be the states at the beginning and at the end of the subinterval T_i. Then, we formulate the nonlinear opti-

mization problem,

$$\begin{aligned}
&\mathcal{P}_{\mathcal{L}}^{MS}(\mu_k, \nu_k): \\
&\quad \min_{\theta_i, \widehat{x}_0^i, \widehat{x}_f^i} \quad J && \text{(4.95a)} \\
&\quad \text{s.t.:} \quad \widehat{x}_f^i = F_{\mathcal{L},\mu_k}^V(\widehat{x}_0^i, \theta_i, \nu_k, T_i) \quad i = 1, \ldots, N && \text{(4.95b)} \\
&\qquad\qquad \widehat{x}_f^{i-1} = \widehat{x}_0^i \qquad i = 2, \ldots, N && \text{(4.95c)} \\
&\qquad\qquad G_{DAE}^V(\widehat{x}_0^1, \widehat{x}_f^N) = 0 && \text{(4.95d)}
\end{aligned}$$

At each algorithm iteration, the NLP $\mathcal{P}_{\mathcal{L}}^{MS}$ has to be solved. This task can be performed by IPOPT with the help of Sundials CVODE to solve the underlying IVP.

Using the direct multiple shooting, the algorithm obtained a solution after 4 iterations. The objective value obtained was 2.86724. The optimal trajectories are very similar to those obtained with the indirect multiple shooting, depicted in Figure 4.1.

- **Case 1 – Direct Collocation Method:** If we want to use the direct collocation method, we need to split the integration interval into N subintervals, where each subinterval T_i, with $i = 1, \ldots, N$, begins at t_0^i and ends at t_f^i. For each subinterval T_i, we approximate the extended state $\widehat{x} = [x_1, x_2, x_c]$ and the extended control variable $\widehat{u} = [u, y]$ with the polynomials

$$\widehat{x}(t) = \sum_{j=0}^{K} \ell_j(\tau)\widehat{x}_{ij} \tag{4.96a}$$

$$\widehat{u}(t) = \sum_{j=1}^{K} \widehat{\ell}_j(\tau)\widehat{u}_{ij} \tag{4.96b}$$

for $t \in [t_0^i, t_f^i]$. Where $\widehat{x}_{ij}$ and $\widehat{u}_{ij}$ are the extended state and the extended controls at the collocation j in the subinterval T_i, and the polynomial basis ℓ_j and $\widehat{\ell}_j$ are defined in (2.111) and (2.121).

Let us define an extended dynamic function

$$\widehat{f} = \begin{bmatrix} y + u \\ x_1 \\ \left(x_1^2 + x_2^2 + u^2\right) + \nu_k \left[(1 - x_2^2)x_1 - x_2 - y\right] \\ \qquad + \frac{\mu_k}{2} \left\|(1 - x_2^2)x_1 - x_2 - y\right\|^2 \end{bmatrix} \tag{4.97}$$

which contains the dynamics of x_1, x_2, and x_c.

Following the same procedure that was used in the direct collocation example presented in Section 3.7.2, we obtain the NLP problem

$$\min_{\theta_i, \widehat{x}_0^i, \widehat{x}_f^i} \quad J \tag{4.98a}$$

$$\text{s.t.:} \quad \sum_{j=0}^{K} \widehat{x}_{ij} \frac{d\ell_j}{d\tau}(\tau_k) = h_i \widehat{f}(\widehat{x}_{ik}, \widehat{u}_{ik}, t_{ik}), \quad \forall i, \forall k \tag{4.98b}$$

$$\widehat{x}_{i+1,0} = \sum_{j=0}^{K} \ell_j(1) \widehat{x}_{ij}, \qquad i = 1, \ldots, N-1 \tag{4.98c}$$

$$\widehat{x}_{ij} = \begin{bmatrix} x_{0,1} \\ x_{0,2} \\ 0 \end{bmatrix}, \qquad i = 1, j = 0 \tag{4.98d}$$

$$J = \widehat{x}_{ij,3}, \qquad i = N, j = K \tag{4.98e}$$

Using IPOPT to solve the NLP problem, the algorithm converged within 4 iterations. The objective value obtained was 2.8688. The optimal trajectories agree with the optimal methods, being presented in Figure 4.1.

- **Case 2 – Direct Multiple Shooting** The direct multiple shooting for Case 2 obtains the same NLP of the direct multiple shooting for Case 1 (4.95), with the additional constraint

$$u_L \leq u_{ij} \leq u_U \qquad i = 1, \ldots, N, \; j = 1, \ldots, K. \tag{4.99}$$

where $u_L = -0.3$ and $u_U = 1$.

Using IPOTP to solve the NLP and CVODE for solve the ODE, the algorithm obtains a solution after 5 iterations. The objective value obtained was 2.86732. The optimal trajectories are equal to those in Figure 4.2.

- **Case 2 – Direct Collocation Method** The direct collocation method for Case 2 yields NLP with the same constraints of Case 1, (4.98), with the additional constraints

$$u_L \leq u_{ij} \leq u_U \qquad i = 1, \ldots, N,\ j = 1, \ldots, K. \tag{4.100}$$

  Using IPOPT to obtain the solution of the NLP, the algorithm returned a solution after 5 iterations. The objective value obtained was 2.86725. The trajectories obtained are equal to those depicted in Figure 4.2.

- **Case 3 – Direct Multiple Shooting** As for the Case 2, the direct multiple shooting obtains a NLP equal to Case 1 (4.95), here, however, the additional constraints are

$$x_L \leq \widehat{x}_f^i \leq x_U, \qquad i = 1, \ldots, N, \tag{4.101a}$$
$$u_L \leq u_{ij} \leq u_U, \qquad i = 1, \ldots, N,\ j = 1, \ldots, K. \tag{4.101b}$$

  After 6 iterations a solution was achieved, the objective value obtained was 2.93604. The results were similar to those shown in Figure 4.3.

- **Case 3 – Direct Collocation Method** For the direct collocation, the NLP of Case 3 is similar to (4.98), with the inclusion of the constraints

$$x_L \leq \widehat{x}_{ij} \leq x_U \qquad i = 1, \ldots, N\, j = 1, \ldots, K. \tag{4.102a}$$
$$u_L \leq u_{ij} \leq u_U \qquad i = 1, \ldots, N,\ j = 1, \ldots, K. \tag{4.102b}$$

  After 5 iterations a solution was obtained. The objective value obtained was 2.95373. Figure 4.3 shows the profiles obtained for the states and control variables.

In the following section, the results of each approach are discussed and compared.

4.6.5 Discussion on Numerical Results

This section gives a brief discussion about the numerical results obtained with the various methods considered. For the analysis, the three cases were implemented for the direct and indirect methods, multiple shooting and collocation, using the traditional

approach, as shown in Section 3.7 and 3.6, in addition to the experiments discussed in this chapter.

The numerical results of the algorithm can be evaluated with regards to several aspects. Herein the following points will be considered:

1. According to Theorem 12, the objective value of the solution obtained by the algorithm should converge to the optimal value, $\{J_{\mu_k}\} \to J^*$.
2. Also in Theorem 12, we learned that the violation of the algebraic equation has to go zero as the algorithm converges.
3. Corollary 3 gives an update rule that makes $\{\nu_k\} \to \nu^*$.
4. The computation cost of the algorithm should not be prohibitive.

These points will be addressed individually:

1. For all cases and methods, the reported objective values obtained using the proposed algorithm were equal to those obtained using the traditional approach, presented in Chapter 3. Which allow us to conclude that, at least for the tested cases, $\{J_{\mu_k}\} \to J^*$.
2. To asses whether or not the algorithm is forcing the violation of the algebraic equation to go to zero, a root mean square (RMS) functional is used. The RMS is the same used in signal processing. For a function g, the RMS is given by

$$RMS(g) = \sqrt{\int_{t_0}^{t_f} \|g(x_k, y_k, u_k, t)\|_2^2 \, dt} \tag{4.103}$$

   Using this metric, in all the cases the RMS of the algebraic equation converged to zero. Figure 4.4 shows a plot of the RMS at each iteration of the indirect methods applied for Case 1, where the plot has a logarithmic ordinate axis.
3. To verify the convergence of the multiplier approximation ν_k, the convergence of the sequence $\{\nu_k\}$ was compared to the ν^* obtained using the optimality conditions (3.255). All solutions of Case 1 and 2 showed that ν_k converges to the same trajectory ν^* that was obtained using the traditional approach. With

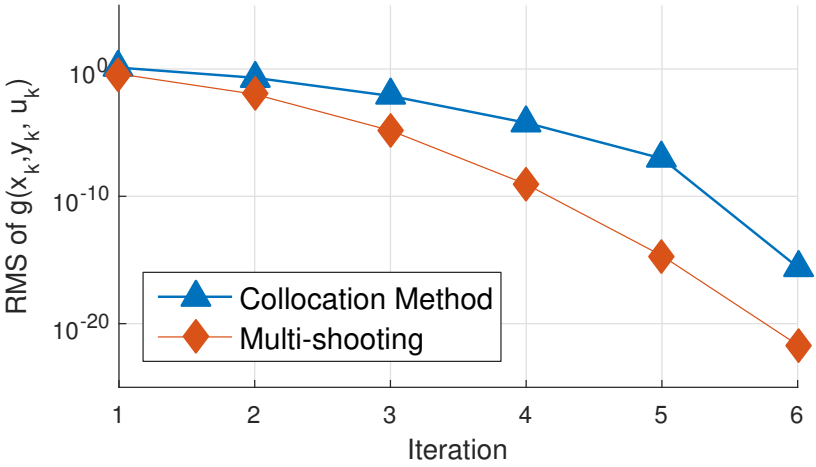

Figure 4.4: RMS of $g(x_k, y_k, u_k, t)$ for Case 1 solved with indirect methods.

the theory provided in this work, it is not possible to formulate the optimality conditions for Case 3, hence it was not possible to verify the convergence of $\{\nu_k\}$ to the optimal ν^*. However, in Case 3, the sequence $\{\nu_k\}$ converged to a smooth and bounded trajectory, which is an indicative that it converged to the optimal ν^*. This matter is a topic for further investigation. Figure 4.5 shows that the profile obtained with the indirect multiple shooting with the traditional approach and with the algorithm after 4 algorithm iterations. Figure 4.6 shows the difference between the two function, we can verify that they close to each other.

4. Regarding the computational cost, it is important to emphasize that being faster than the traditional approaches is not one of the goals of the algorithm, however having a competitive solution time is a good property. The solution time of the indirect methods are presented in Table 4.1, and the solution time of the direct methods are presented in Table 4.2. In these tables, the column "ODE" presents the computational time to obtain a solution of the OCP $\mathcal{P}^V_{ODE}$ (3.246), which is an OCP of the Van der Pol oscillator modeled using ODEs,

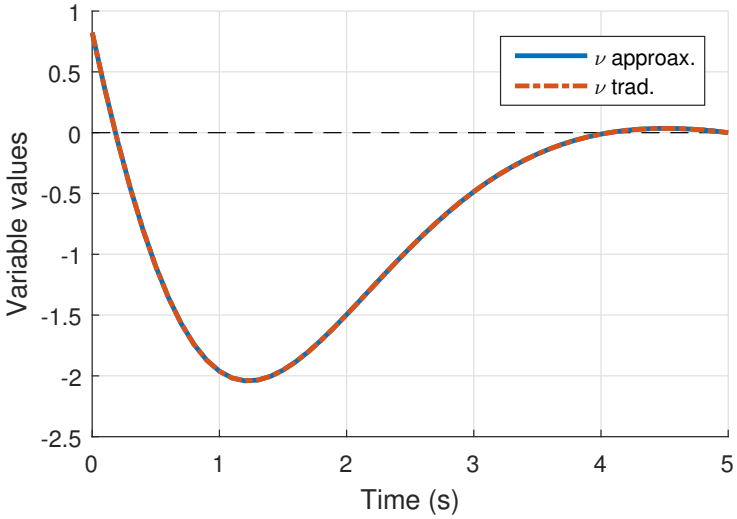

Figure 4.5: Comparison between the multiplier ν_k with k large enough and ν^* obtained using the necessary optimality conditions.

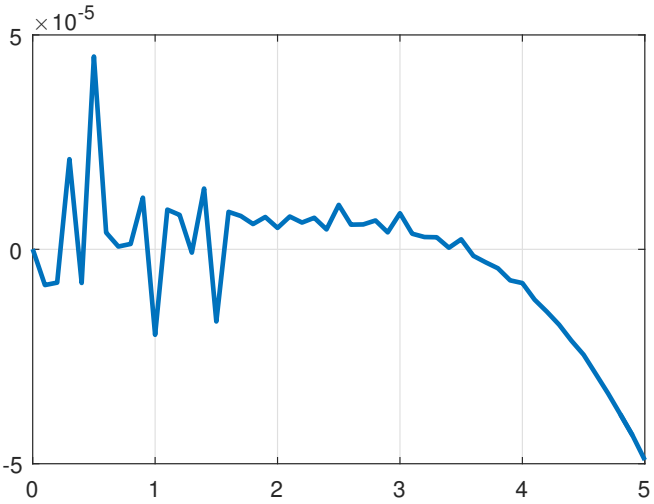

Figure 4.6: Diference between the multiplier ν_k with $k = 4$ and ν^* obtained using the necessary optimality conditions. Notice the that the values are in the range of 10^{-5}.

with the appropriate approaches presented in Chapter 3. The column "DAE" presents the computational time for solving the OCP $\mathcal{P}^V_{DAE}$ (3.247), which is an OCP of the Van der Pol oscillator modeled using DAEs, using the approaches presented in Chapter 3. The column "Aug. Lagrangian" presents the computational for solving the obtaining a solution of the OCP $\mathcal{P}^V_{DAE}$ (4.62), using the proposed algorithm. In the lines, we have the three cases with the times for collocation method ("Coll.") and multiple shooting method ("MSM").

Regarding the results presented in the tables, we can see that when the algorithm is compared to the "DAE" the computational times are close for the indirect methods. On the other hand, when the same comparison is made for direct method, the algorithm presented lower performance, taking about four times more to solve than traditional approach ("DAE" column). This poor result occurs because the solver is taking a considerable time to solve each algorithm iteration, even if a good initial guess is given. It is expected to improve this result by adjusting the parameters of the solver.

Table 4.1: Solution time obtained with indirect methods.

		ODE	DAE	Aug. Lagrangian
Case 1	Coll.	0.69 s	0.89 s	0.95 s
	MSM	0.45 s	0.83 s	1.75 s
Case 2	Coll.	0.64 s	0.86 s	0.97 s
	MSM	0.72 s	1.32 s	1.88 s
Case 3	Coll.			1.22 s
	MSM			3.18 s

Table 4.2: Solution time obtained with direct methods.

		ODE	DAE	Aug. Lagrangian
Case 1	Coll.	0.21 s	0.23 s	0.86 s
	MSM	0.62 s	0.94 s	2.31 s
Case 2	Coll.	0.15 s	0.21 s	2.83 s
	MSM	0.56 s	0.93 s	3.97 s
Case 3	Coll.	0.17 s	0.18 s	0.97 s
	MSM	0.51 s	0.85 s	3.42 s

5 CONCLUSION

5.1 CONTRIBUTIONS

The main contributions of this work are twofold:

1. First, this dissertation gives a compact but tutorial overview of the optimal control theory with application to differential-algebraic equations.

2. Secondly, a new method was introduced for solving OPCs of DAEs, based on the augmented Lagrangian.

By gathering information from different references, this dissertation built up the minimum knowledge to work with optimal control. By no means, it was expected to cover the whole area of optimal control but to give the reader a starting point to understand optimal control. The review of dynamic systems from Chapter 2, gives tools to formalize systems and problems, as well as to solve them. Chapter 3 gave a review on the optimal theory which was primal in the context of this dissertation. Finally, Chapter 4 stated the proposed algorithm, gave mathematical proofs, and showed that the algorithm works with numerical experiments.

The proposed algorithmic framework solves an OCP of DAE systems through a sequence of OCPs of ODE systems. The framework relies on ODE solvers which are computationally efficient and readily available. Another property of the transformation performed by the algorithm is the easy inclusion of bound constraints on the states, which otherwise, would not be possible using the common approaches of indirect methods.

The mathematical properties have shown that if each iteration of the algorithm is the global solution of the auxiliary OCP, then the solution converges to the global minimum of the original OCP, and the relaxed algebraic equation is satisfied. Under not ideal conditions, the solution provided by the proposed algorithm also converges. If each iteration gives a locally optimum of the subproblem, then the solution will converge to a locally optimal solution of the original problem. Also, if the solution of each iteration suboptimal iterations are increasingly closer to the optimal of the auxiliary problem, then the solution will converge to the optimum of the original problem.

Numerical experiments were implemented to verify how the proposed algorithm performs in practice. The experiments followed using direct and indirect approaches and used multiple shooting

and the collocation method. To assess the algorithm performance, different cases were proposed. For all the cases and the approaches the algorithm worked as intended, with the mathematical properties being verified in practice. The experimental results have shown that the proposed method has competitive solution time with respect to the traditional DAE approach, while ensuring a sufficiently small violation in algebraic constraints, using less specialized tools, and solving a larger class of problems.

5.2 FUTURE WORK

In the near future, it is expected to use this algorithm to develop an approach for distributed optimal control for dynamic networks of nonlinear systems.

Consider the classical four tanks problem, presented in Figure 5.2. We can represent this system with a directional graph, where nodes are subsystems and the arcs represents the relation between them, as shown in Figure 5.2.

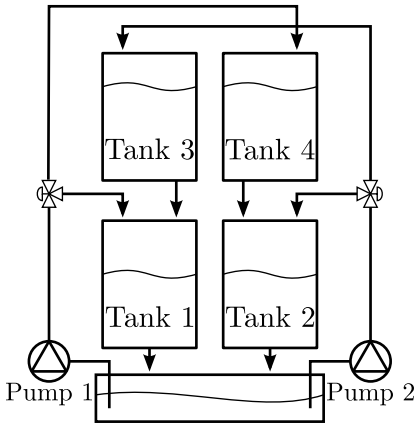

Figure 5.1: Schematic of the Four Tanks system.

To model the dynamics of such a dynamic network, we have for every subsystem (node of the graph) v

$$\dot{x}_v = f_v(x_v, y_v, \widehat{y}_v, u_v) \tag{5.1a}$$
$$g_v(x_v, y_v, \widehat{y}_v, u_v) = 0 \tag{5.1b}$$
$$\widehat{g}_v(\widehat{y}_v, \widehat{y}_{N(v)}) = 0 \tag{5.1c}$$

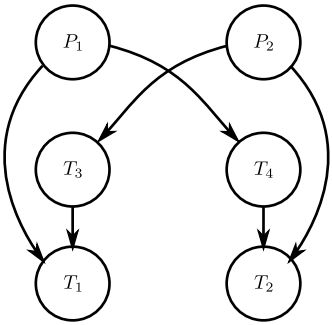

Figure 5.2: Representation of the Dynamic Network of the Four Tanks system..

where $N(v)$ are the neighbors of system v, x_v is the state of the subsystem, y_v is the algebraic variable that is internal to the subsystem, u_v is the control variable of the subsystem, $\widehat{y}_v$ is the algebraic variables of the subsystem v that relate to other subsystems, and $\widehat{y}_{N(v)}$ are the algebraic variables $\widehat{y}_{\widehat{v}}$ of all the neighbors $\widehat{v} \in N(v)$. The function f_v defines the dynamics, the function g_v defines the internal algebraic relations, and the function $\widehat{g}_v$ connects the system v with its neighbors.

Then we can write an OCP in the form

$$\begin{aligned}
\min \quad & \sum_{v \in V} \int_{t_0}^{t_f} \phi_v(x_v, z_v, y_v, u_v)\, dt \\
\text{s.t.:} \quad & \dot{x}_v = f_v(x_v, y_v, \widehat{y}_v, u_v) \quad \forall v \in V \\
& g_v(x_v, y_v, \widehat{y}_v, u_v) = 0 \quad \forall v \in V \\
& \widehat{g}_v(\widehat{y}_v, \widehat{y}_{N(v)}) = 0
\end{aligned}$$

for which we want to use the proposed algorithm to relax the algebraic constraint $\widehat{g}_v(\widehat{y}_v, \widehat{y}_{N(v)}) = 0$. By doing so, we expect to solve it in a distributed and parallelizable fashion. This idea will be pursued in the doctoral program.

BIBLIOGRAPHY

1 BERTSEKAS, D. P. *Constrained Optimization and Lagrange Multiplier Methods*. [S.l.]: Athena Scientific, 1996. (Athena scientific series in optimization and neural computation).

2 BOYD, S. et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, Now Publishers Inc., v. 3, n. 1, 2011.

3 KÖEGEL, M. J.; FINDEISEN, R. Cooperative distributed MPC using the alternating direction multiplier method. In: *8th IFAC Advanced Control of Chemical Processes*. [S.l.: s.n.], 2012.

4 HENTZELT, S.; GRAICHEN, K. An Augmented Lagrangian Method in Distributed Dynamic Optimization Based on Approximate Neighbor Dynamics. *2013 IEEE International Conference on Systems, Man, and Cybernetics*, Ieee, p. 571–576, oct 2013.

5 LASDON, L.; WAREN, A.; RICE, R. An interior penalty method for inequality constrained optimal control problems. *IEEE Transactions on Automatic Control*, v. 12, n. 4, p. 388–395, aug 1967.

6 BELL, M.; SARGENT, R. Optimal control of inequality constrained DAE systems. *Computers & Chemical Engineering*, v. 24, n. 11, p. 2385–2404, nov 2000.

7 ASCHER, U. M.; PETZOLD, L. R. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. 3600 Market Street, 6th Floor Philadelphia, PA 19104-2688: SIAM, 1998. 332 p. ISBN 9780898714128.

8 ASCHER, U. M.; MATTHEIJ, R. M. M.; RUSSELL, R. D. *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*. [S.l.]: Society for Industrial and Applied Mathematics, 1995. 1–5 p. ISBN 978-0-89871-354-1.

9 STOER, J.; BULIRSCH, R. *Introduction to Numerical Analysis*. New York, NY: Springer New York, 2002. xiii–xiv p. ISBN 978-1-4419-3006-4.

10 HORN, R.; JOHNSON, C. *Topics in Matrix Analysis*. [S.l.]: Cambridge University Press, 1994. ISBN 9780521467131.

11 BIEGLER, L. T. *Nonlinear Programming*. [S.l.]: Society for Industrial and Applied Mathematics, 2010. ISBN 978-0-89871-702-0.

12 ALLEN, R. *Mathematical analysis for economists*. [S.l.]: Macmillan, 1962.

13 TROUTMAN, J. L. *Variational calculus and optimal control: optimization with elementary convexity*. [S.l.]: Springer Science & Business Media, 2012.

14 CLARKE, F. *Functional analysis, calculus of variations and optimal control*. [S.l.]: Springer Science & Business Media, 2013.

15 BERGER, M. *Nonlinearity & Functional Analysis: Lectures on Nonlinear Problems in Mathematical Analysis*. [S.l.]: Elsevier Science, 1977. (Pure and Applied Mathematics). ISBN 9780080570440.

16 ERNZERHOF, M. Taylor-series expansion of density functionals. *Physical Review A*, v. 50, n. 6, p. 4593–4607, dec 1994.

17 KIRK, D. *Optimal Control Theory: An Introduction*. 4th. ed. [S.l.]: Dover Publications, 2004. 480 p.

18 PONTRYAGIN, L. S. et al. *Mathematical Theory of Optimal Processes*. English ed. [S.l.]: INTERSCIENCE PUBLISHERS, 1962. 362 p. ISBN 9782881240775, 2881240771.

19 BERTSEKAS, D. P. *Dynamic Programming and Optimal Control Vol I - Third Edition*. [S.l.]: Athena Scientific, 2005. 543 p. ISBN 1-886529-26-4.

20 KHALIL, H. *Nonlinear Systems*. [S.l.]: Prentice Hall, 2002. (Pearson Education). ISBN 9780130673893.

21 SPEYER, J. L.; JACOBSON, D. H. *Primer on Optimal Control Theory*. [S.l.]: Society for Industrial and Applied Mathematics, 2010. ISBN 0898716942, 9780898716948.

22 WÄCHTER, A.; BIEGLER, T. L. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, v. 106, n. 1, p. 25–57, 2005.

23 HINDMARSH, A. C. et al. Sundials: Suite of nonlinear and differential/algebraic equation solvers. *ACM Transactions on Mathematical Software (TOMS)*, ACM, v. 31, n. 3, p. 363–396, 2005.

24 ANDERSSON, J. *A General-Purpose Software Framework for Dynamic Optimization*. Tese (PhD thesis) — Arenberg Doctoral School, 2013.

25 CHENEY, E. W. *Introduction to approximation theory*. 2. ed. [S.l.]: American Mathematical Society, 2000. (AMS Chelsea Publishing). ISBN 9780821813744,0821813749.

26 TREFETHEN, N. Six myths of polynomial interpolation and quadrature. *Mathematics Today*, v. 47, n. 4, p. 184–188, 2011.

27 BERRUT, J.-P.; TREFETHEN, L. N. Barycentric lagrange interpolation. *Siam Review*, SIAM, v. 46, n. 3, p. 501–517, 2004.

28 NOCEDAL, J.; WRIGHT, S. *Numerical Optimization*. [S.l.]: Springer, 2006.

APPENDIX A – DEMONSTRATIONS AND PROOFS

Theorem 1. *Let X be a normed linear space and $I : X \to \mathbb{R}$ be functional differentiable in $x^* \in X$. If I has local extremum at x^*, then $I'(x^*) = 0$*

Proof. By definition if $x^* \in X$ is a minimum of $I : X \to \mathbb{R}$, then there is a $r > 0$ such that $I(x^* + h) \geq I(x^*)$ for all h in a ball $\|h\| < r$.

To prove by contradiction, we suppose that $I'(x^*)h_0 \neq 0$ for some $h_0 \in X$. Assuming

$$h_n = -\frac{1}{n}\frac{|I'(x^*)h_0|}{I'(x^*)h_0}h_0 \tag{A.3}$$

as $n \to \infty$ we have $\|h_n\| \to 0$, and with a sufficiently large N we have $\|h_n\| < r$ for all $n > N$. By definition of derivative

$$\frac{I(x^* + h_n) - I(\bar{x})}{\|h_n\|} = \frac{I'(x^*)h_0}{\|h_n\|} + \epsilon(h_n) \tag{A.4}$$

we know that

$$\frac{I'(x^*)h_n}{\|h_n\|} = \frac{-\frac{1}{n}\frac{|I'(x^*)h_0|}{I'(x^*)h_0}I'(x^*)h_0}{\left|-\frac{1}{n}\frac{|I'(x^*)h_0|}{I'(x^*)h_0}\right|\|h_0\|} = -\frac{|I'(x^*)h_0|}{\|h_0\|} \tag{A.5}$$

For all $n > N$ we have

$$-\frac{|\dot{I}_{x^*}(h_0)|}{\|h_0\|} + \epsilon(h_n) = \frac{I(x^* + h_n) - I(x^*)}{\|h_n\|} \geq 0 \tag{A.6}$$

taking the limit $n \to \infty$, we obtain that $-|\dot{I}_{x^*}(h_0)| \geq 0$ which contradicts the assumptions. Therefore we conclude that $I'(x^*) = 0$. □

Theorem 2. *Let functional $I : X \to \mathbb{R}$ be Fréchet differentiable. Then I is also Gateaux differentiable. Furthermore, the Gateaux and Fréchet derivative agree,*

$$I'(\bar{x})h = \delta I(\bar{x}, h) \tag{A.7}$$

for all $h \in X$.

Proof. Assuming $h = \xi\hat{h}$, the definition of Fréchet derivative gives

$$I(\bar{x} + \xi\hat{h}) = I(\bar{x}) + I'(\bar{x})[\xi\hat{h}] + \epsilon(\xi\hat{h})\|\xi\hat{h}\|. \tag{A.8}$$

Using the linearity of the Fréchet derivative and the homogeneous property of the norm, we have

$$\frac{I(\bar{x}+\xi\hat{h})-I(\bar{x})}{\xi} = I'(\bar{x})\hat{h} + \frac{|\xi|}{\xi}\|\hat{h}\|\epsilon(\xi\hat{h}) \tag{A.9}$$

As $\xi \to 0$, $\xi h \to 0$ and $\epsilon(\xi h) \to 0$, while $\frac{|\xi|}{\xi} = \pm 1$. Therefore,

$$\delta I(\bar{x},\hat{h}) = \lim_{\xi\to 0}\left[\frac{I(\bar{x}+\xi\hat{h})-I(\bar{x})}{\xi} - \frac{|\xi|}{\xi}\|\hat{h}\|\epsilon(\xi\hat{h})\right] = I'(\bar{x})\hat{h} \tag{A.10}$$

□

Theorem 16. *The Fréchet derivative of a Fréchet differentiable functional $I : X \to \mathbb{R}$ at the point $\bar{x} \in X$ is unique.*

Proof. Let $L : X \to \mathbb{R}$ be a linear functional, if

$$\frac{L(h)}{\|h\|} \to 0 \text{ as } \|h\| \to 0 \tag{A.11}$$

then $L(\cdot) = 0$. By contradiction, let $L(h_0) \neq 0$ for some nonzero $h_0 \in X$. Assuming $h_n = \frac{1}{n}h_0$, note that if $n \to \infty$ the norm $\|h\| \to 0$, however using the linearity of L and the homogeneity of the absolute, we have

$$\lim_{n\to\infty}\frac{L(h_n)}{\|h_n\|} = \lim_{n\to\infty}\frac{\frac{1}{n}L(h_0)}{|\frac{1}{n}|\|h_0\|} = \frac{L(h_0)}{\|h_0\|} \neq 0 \tag{A.12}$$

that contradicts the assumption (A.11), therefore the conclusion that $L(\cdot) = 0$ holds.

Let $L_1 : X \to \mathbb{R}$ and $L_2 : X \to \mathbb{R}$ be continuous linear functionals such that

$$I(\bar{x}+h) = I(\bar{x}) + L_1(h) + \epsilon_1(h)\|h\|, \text{ for all } h \in X \tag{A.13a}$$
$$I(\bar{x}+h) = I(\bar{x}) + L_2(h) + \epsilon_2(h)\|h\|, \text{ for all } h \in X \tag{A.13b}$$

with $\epsilon_1 \to 0$ and $\epsilon_2 \to 0$ as $\|h\| \to 0$. Subtracting both equations

$$\frac{(L_1 - L_2)}{\|h\|} = (\epsilon_1 - \epsilon_2)(h) \to 0 \text{ as } \|h\| \to 0 \tag{A.14}$$

therefore $L_1 = L_2$. □

Theorem 17. *Given that a sequence* $\{h_n = g(f_n)\}$ *converges uniformly* h^**, where* $h_n : [0,1] \to \mathbb{R}^{N_g}$*,* $g : \mathbb{R}^{N_f} \to \mathbb{R}^{N_g}$*, and* $f_n : [0,1] \to \mathbb{R}^{N_f}$*. Which means that* $\forall \varepsilon_h > 0$*,* $\exists N_h$*, such that for* $k \geq N_h$

$$\|g(f_n)) - h^*\| < \varepsilon_h \tag{A.15}$$

with the norm

$$\|f\| = \max_{x \in [0,1]} \|f(x)\|_\infty \tag{A.16}$$

Then, under assumption that g *is invertible and that the inverse* g^{-1} *is Lipschitz continuous,* $\{f_n\}$ *converges uniformly to some* $f^* = g^{-1}(h^*)$*. Which means that* $\forall \varepsilon_f$*,* $\exists N_f$*, such that for* $k \geq N_f$

$$\|f_n(x) - f^*(x)\| < \varepsilon_f. \tag{A.17}$$

Proof. The proof follows by contradiction, consider that $\exists \varepsilon_f$, $\forall N_f$, such for $k \geq N_f$

$$\|f_n(x) - f^*(x)\| \geq \varepsilon_f \tag{A.18}$$

Since g is invertible

$$f_n - f^* = g^{-1}(g(f_n)) - g^{-1}(h^*) \tag{A.19}$$

where we define $f^* = g^{-1}(h^*)$

A function $F : X \to Y$ is Lipschitz continuous if

$$\|F(a) - F(b)\|_Y \leq M \|a - b\|_X \tag{A.20}$$

where $a, b \in X$, and $\|\cdot\|_X$ and $\|\cdot\|_Y$ are the norms of the spaces X and Y. Using $F = g^{-1}$, $a = g(f_n(x))$, $b = h^*(x)$, and using the norm (A.16) on both side of the equation,

$$\left\|g^{-1}(g(f_n)) - g^{-1}(h^*)\right\| \leq M \|g(f_n) - h^*\| \tag{A.21}$$

Choose an N_h such that $M\varepsilon_h = \varepsilon_f$. Then, for all $n \geq N_h$

$$\begin{aligned}\left\|g^{-1}(g(f_n(x))) - g^{-1}(h^*(x))\right\| &\leq M \|g(f_n(x)) - h^*(x)\| \\ &< M\varepsilon_h = \varepsilon_f\end{aligned} \tag{A.22}$$

using the identity (A.19)

$$\|f_n - f^*\| < \varepsilon_f \tag{A.23}$$

which contradicts assumption (A.18). Therefore, we must have $\{f_n\} \to g^{-1}(h^*) = f^*$. □

Theorem 18. *Let $g : \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}$ be a continuous function, and the sequence of function f_n to converge uniformly to f, where $f_n : [0,1] \to \mathbb{R}^{d_1}$. Let the function norm $\|\cdot\|$ be given by*

$$\|g\| = \max_{x \in [0,1]} \|g(x)\|_\infty \tag{A.24}$$

Then $\{g(f_n)\}$ converges uniformly to $g(f)$.

Proof. If f_n converges uniformly to f, then for all ε_f there exists an N, such that for all $n > N$

$$\|f_n - f\| < \varepsilon_f \tag{A.25}$$

and there exists an upperbound M such that for all $n \in \mathbb{N}$

$$\|f_n\| \leq M \tag{A.26}$$

Then, consider $g : [-M, M]^{d_1} \to \mathbb{R}^{d_2}$. As g is continuous in a compact set, for all $\varepsilon_g > 0$, there exists a $\delta_g > 0$ such that

$$\|g(z_1) - g(z_2)\| < \varepsilon_g \tag{A.27}$$

for all $\|z_1 - z_2\| < \delta_g$. Using $\varepsilon_f = \delta_g$,

$$\|f_n - f\| < \varepsilon_f = \delta_g \tag{A.28}$$

for all $n > N$. Therefore,

$$\|g(f_n) - g(f)\| < \varepsilon_g \tag{A.29}$$

for all $n > N$. □

APPENDIX B – AUGMENTED LAGRANGIAN FOR CONSTRAINED OPTIMIZATION

In mathematical programming, the augmented Lagrangian method [1] is used to solve an equality constrained optimization problem (COP) through a sequence of unconstrained optimization problem (UOP). Let COP be of the form:

$$\min_{z} \; V(z) \tag{B.1a}$$

$$\text{s.t.:} \; c(z) = 0 \tag{B.1b}$$

The augmented Lagrange method relax the equality constraint (B.1b) and includes a penalization term in the objective function creating an augmented objective function:

$$V_{\mu_k}(z, \lambda_k) = V(z) + \lambda^T c(z) + \frac{\mu}{2} \left\| c(z) \right\|^2 \tag{B.2}$$

where $\mu_k > 0$ is a scalar that belongs to sequence $\{\mu_k\} \to \infty$, and λ_k is approximation of the Lagrange multiplier of the constraint $c(z)$, which belongs to a sequence $\{\lambda_k\} \to \lambda^*$ [1].

The solution of (B.1) is obtained by a sequence of unconstrained minimizations of (B.2), determined by a scalar μ_k and a vector λ_k that are updated at each iteration. The method is outlined in Algorithm 2 [28].

Algorithm 2 Augmented Lagrangian for Constrained Optimization

Require: $\mu_0 > 0$, $\varepsilon_{V,0} > 0$, starting points z_0^s and λ_0:
1: **for** $k = 0, 1, \ldots$ **do**
2: Find a z_k that minimizes $V_{\mu_k}(z, \lambda_k)$, starting at z_k^s, satisfying $\left\| \frac{\partial V_{\mu_k}}{\partial z}(z_k, \lambda_k,) \right\| \leq \varepsilon_{V,k}$,
3: **if** z_k satisfies a convergence condition, **then**
4: **return** the solution z_k,
5: **end if**
6: Obtain λ_{k+1} with the equation $\lambda_{k+1} = \lambda_k + \mu_k c(z_k)$,
7: Choose a new parameter $\mu_{k+1} \geq \mu_k$,
8: Set the starting point for the next iteration $z_{k+1}^s = z_k$,
9: Select tolerance $\varepsilon_{V,k+1}$
10: **end for**

A traditional rule for updating parameter μ_k, in line 7, is

$$\mu_{k+1} = \beta \mu_k \tag{B.3}$$

where β is a scalar greater than 1, usually in the range from 5 to 10. However, if μ_k is large, then the minimization of (B.2) might become ill conditioned [1]. To this end, an alternative update rule is

$$\mu_{k+1} = \begin{cases} \beta\mu_k & \text{if } \beta\mu_k < \mu_{\max} \\ \mu_{\max} & \text{otherwise} \end{cases} \tag{B.4}$$

There exists a theoretical value μ^* that, for any $\mu > \mu^*$, $\{V_\mu(z_k, \lambda_k)\} \rightarrow V(z^*)$ where z^* is a solution for (B.1), if the tolerance $\varepsilon_{V,k+1} \rightarrow 0$ as $k \rightarrow \infty$ and the problem satisfies some conditions [28].