National University of Singapore, MA5232 - Modeling and Numerical Simulations (Part III)

Optimal Control Theory

Qianxiao Li

v0.1

Preface

These lecture notes are compiled for the special topic in MA5232 Modeling and Numerical Simulations in Semester 2 of AY 2020-21.

This document is typeset using LaTeXwith a modified theme based on

https://www.overleaf.com/latex/templates/lecture-note-template/dwyrjrnthdcz

If you find any mistakes or typos in the notes, please send me an email at qianxiao@nus.edu.sg.

Contents

1 Introduction

1.1 Overview

Optimal control theory is an important topic of study in applied mathematics. In some sense, it is the culmination of a series of work on calculus of variations that originates from classical mechanics. In modern times, optimal control finds applications in a variety of fields, including aerospace engineering, systems engineering, financial engineering and machine learning.

These notes gives a brief introduction to the theory of optimal control to mathematics students, with emphasis on both the underlying mathematical theory, and numerical algorithms for control problems. Due to limited time, we will only cover the essential topics in each case, and the interested reader may consult the cited reference books for further study.

1.2 Ordinary Differential Equations

In this section, we introduce some basics of ordinary differential equations that will be useful to us for later chapters. We will not present any proofs since they can be found in any standard introductory text (e.g. [Arn12, Cod12]) the readers are assumed to have some familiarity with the topic, but we will state without proof a few useful properties and illustrate some relevant phenomenon with examples.

1.2.1 Basic Definitions

We will work in $\mathbb{R}^d$. An ordinary differential equation (ODE) is the equation

$$\dot{x}(t) = f(x(t)), \qquad x(0) = x_0 \in \mathbb{R}^d, \tag{1.1}$$

where $\dot{x}$ denotes the time derivative, $f : \mathbb{R}^d \to \mathbb{R}^d$ is a function or vector field and x_0 is the initial condition. This called a *time homogeneous* ODE since the vector field on the right does not depend explicitly on time t. On the other hand, a *time-inhomogeneous* ODE is given by

$$\dot{x}(t) = f(t, x(t)), \qquad x(0) = x_0 \in \mathbb{R}^d. \tag{1.2}$$

We note that minus technical conditions, these two equations are equivalent. First, obviously (1.2) includes (1.1). For the reverse direction, we define an auxiliary variable $x^0 \in \mathbb{R}$ such that $\dot{x}^0(t) = 1$, $x^0(0) = 0$ so that $x^0(t) = t$. Then, we can rewrite (1.2) by defining $\tilde{x} = (x^0, x)$,

$\tilde{f}(\tilde{x}) = (1, f(x^0, x))$ exactly in the form of (1.1). Hence, for convenience we will work with either (1.1) and (1.2), keeping in mind that they are effectively equivalent for most purposes.

By a solution of an ODE on $[0, T]$ we mean a function $\boldsymbol{x} : [0, T] \to \mathbb{R}^d$ with $\boldsymbol{x} := \{x(t) : t \in [0, T]\}$ that satisfies (1.2).

Example 1.1: Linear ODEs

Let $d = 1$ and $f(x) = ax$ with $a \in \mathbb{R}$. Then, check that

$$x(t) = e^{at} x_0, \tag{1.3}$$

is the solution to (1.1). More generally, consider $d \geq 1$ and $f(x) = Ax$ where $A \in \mathbb{R}^{d \times d}$. Then,

$$x(t) = e^{tA} x_0, \tag{1.4}$$

is the solution to (1.1). Here $e^C := \sum_i C^i / i!$ denotes the usual matrix exponential.

The definition of solution requires $\boldsymbol{x}$ to be differentiable on $(0, T)$. But we remark that it is possible to relax this by considering *integral forms*. For example, we can write (1.2) as

$$x(t) = x_0 + \int_0^t f(s, x(s)) ds. \tag{1.5}$$

The advantage here is that we can consider less regular $\boldsymbol{x}$ to be solutions of ODEs, e.g. here it is only required for $\boldsymbol{x}$ to be *absolutely continuous*, meaning that $\boldsymbol{x}$ satisfies (1.2) for almost every t. One of the most basic results concerns when a solution to (1.2) (or (1.5)) exists. The following result gives sufficient conditions, and we will hereafter always assume that a unique solution exists to whichever ODE we deal with.

Theorem 1.2: Picard–Lindelöf Theorem

Let f be continuous in t and uniformly Lipschitz in x, i.e. there exists a constant C such that $\|f(t, x) - f(t, x')\| \leq C\|x - x'\|$ for all $x, x' \in \mathbb{R}^d$ and $t \in [0, T]$. Then, there exists a unique solution to (1.2) on $[0, T]$.

1.2.2 Flow Map and Dependence on Initial Condition

One way to look at ODEs is to look at its solution trajectories given initial condition. Alternatively, we can also look at what the solution does to a set of initial conditions at a fixed terminal time. In other words, we define the *flow* or the *flow map* $\varphi_t : \mathbb{R}^d \to \mathbb{R}^d$

$$\varphi_t(x) := x(t) \qquad \text{where} \qquad \dot{x}(s) = f(s, x(s)), \quad s \in [0, t], \quad x(0) = x \tag{1.6}$$

In fact, the set $\Phi := \{\varphi_t : t \in \mathbb{R}\}$ forms a one-parameter continuous group of transformations on $\mathbb{R}^d$, under the binary operation of function composition. Analyzing the set Φ can be seen as an alternative way to understand ODEs.

The following properties are well-known and easy to check:

- φ_t is continuous for each t
- φ_0 is the identity mapping, $\varphi_0(x) = x$ for all x
- If f does not depend on t, then $\varphi_t \circ \varphi_s = \varphi_{t+s}$, i.e. $t \longrightarrow \varphi_t$ is a homomorphism.

One can also ask how sensitive the terminal state of the ODE is to the initial condition. This can be captured by the jacobian of φ_t, $[\nabla\varphi_t(x)]_{ij} = \partial_j\varphi_{t,i}(x)$. The following result will be useful to us later.

Theorem 1.3: Dependence on Initial Condition

Let f be continuously differentiable in x, and Lipschitz in x uniformly in t. Let $\boldsymbol{x}$ be the solution of the ODE (1.2) with flow map φ_t and $\boldsymbol{v}$ be the solution to the linear time-inhomogeneous ODE

$$\dot{v}(s) = \nabla_x f(s, x(s))v(s), \qquad s \in [0, t], \qquad v(0) = v_0. \tag{1.7}$$

Then, we have

$$\lim_{\varepsilon \to 0^+} \left\| \frac{\varphi_t(x_0 + \varepsilon v_0) - \varphi_t(x_0)}{\varepsilon} - v(t) \right\| \to 0, \tag{1.8}$$

uniformly in $t \in [0, T]$ for $\|v_0\| \leq 1$.

Corollary 1.4

Under the same conditions as in Theorem 1.3, the Jacobian $J(t) := \nabla_x\varphi_t(x_0)$ satisfies the linear equation

$$\dot{J}(t) = \nabla_x f(t, x(t))J(t), \qquad J(0) = I. \tag{1.9}$$

Equation (1.7) is called the variational equation associated with the ODE (1.2). It describes the propagation of variations of the initial condition along the evolution in time, hence its name. We will refer back to these results in our discussion of optimal control theory.

Example 1.5: Flow Map and Variational Equations for Linear Systems

Recall the linear system in Example 1.1. In this case, the flow map is a linear function

$$\varphi_t(x) = e^{tA}x, \tag{1.10}$$

with Jacobian $J(t) = e^{tA}$ (in this case, the Jacobian does not depend on x_0). Check that $J(t)$ satisfies the variational equation

$$\dot{J}(t) = AJ(t), \qquad J(0) = I, \tag{1.11}$$

as shown in the above corollary.

1.2.3 Numerical Solution of ODEs

Often, ODEs do not admit explicit solutions and we have to compute a solution numerically. There are many methods for doing so and it is not the purpose here to give a thorough introduction. Pertaining to the topic discussed in these notes, it is sufficient to first introduce the simplest possible method, the *forward Euler method*.

In this method, we construct a solution to (1.2) by discretizing time and setting

$$\widehat{x}(k+1) = \widehat{x}(k) + \Delta t f(k\Delta t, \widehat{x}(k)), \qquad \widehat{x}(0) = x_0, \tag{1.12}$$

which can be seen as a first-order Taylor expansion of the integral form of the ODE (1.5) for small Δt. The latter is called the *step size*. We expect that this approximation to get better as the step size Δt becomes small. This is made precise in the following result.

Theorem 1.6: Global Truncation Error of Forward Euler Method

Let f be Lipschitz in x uniformly in t and continuous in t. Let $\boldsymbol{x}$ be a solution of the ODE (1.2) with initial condition x_0 and $\widehat{\boldsymbol{x}}$ be the iterates defined in (1.12), then for each $K > 0$ there exists a constant $C > 0$ such that

$$\max_{k \le K} \|\widehat{x}(k) - x(k\Delta t)\| \le C\Delta t. \tag{1.13}$$

2 Optimal Control Theory

The study of optimal control theory originates from the classical theory of the calculus of variations, beginning with the seminal work of Euler and Lagrange in the 1700s. These culminated in the so-called Lagrangian mechanics that reformulate Newtonian mechanics in terms of extremal principles. In a nut shell, the calculus of variations studies optimization over "curves", which one can picture as an infinite dimensional extension of traditional optimization problems.

Optimal control theory is a nontrivial extension of the classical theory of calculus of variations in two main directions: to dynamical and non-smooth settings. This builds on important contributions of Weierstrass and others and led in two inter-related directions: the Pontryagin's maximum principle and the Hamilton-Jacobi-Bellman theory. An interesting historical account of the developments can be found in [Lib12].

In this section, we give a minimal introduction of the problem formulation of optimal control problems, paying particular attention to the so-called *Bolza problems*. The reader is referred to comprehensive texts on optimal control theory for a more complete account [AF13, Lib12, BP07].

2.1 From Calculus of Variations to Optimal Control

2.1.1 A Motivating Example

Finite-dimensional optimization problems are of the form

$$\inf_{x \in X} \Phi(x), \qquad \Phi : X \to \mathbb{R}, \tag{2.1}$$

where X is usually a subset of a Euclidean space. On the other hand, a calculus of variations problem minimizes some functional J over some infinite dimensional space $\mathcal{X}$, i.e.

$$\inf_{\boldsymbol{x} \in \mathcal{X}} J[\boldsymbol{x}] \qquad J : \mathcal{X} \to \mathbb{R}. \tag{2.2}$$

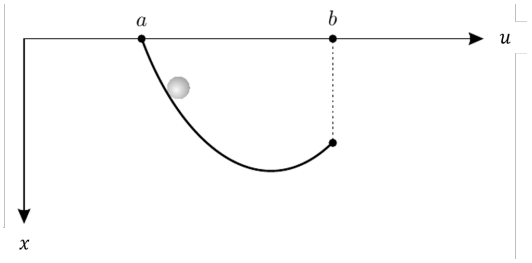
There are many possible forms of the functional J and the space $\mathcal{X}$. For example, one may encounter functionals in the form of an integral, where the argument $\boldsymbol{x} = \{x(u) : u \in [a, b]\}$ is a function of a scalar variable u, i.e.

$$J[\boldsymbol{x}] = \int_a^b L(u, x(u), x'(u)) du \tag{2.3}$$

Let us consider motivating example problem of this nature that is also of substantial historical importance.

Example 2.1: Rolling a Ball Down a Ramp

Let $a < b$ be two points on a horizontal plane, and our goal is to build a ramp such that when we release the ball from point a, it can arrive at a point directly under point b in the *shortest* time possible. See figure below. We will assume that there is no friction.



What shape of the ramp will achieve this task? It turns out that we can phrase this as a calculus of variations problem. Let $s(u)$ be the instantaneous speed of the ball when its horizontal coordinate is at u, and let $\{x(u)\}$ denote the shape of the ramp and that $x(a) = 0$. By conservation of energy we find that

$$\frac{1}{2}ms(u)^2 = mgx(u) \qquad \Rightarrow \qquad s(u) = \sqrt{2gx(u)} \tag{2.4}$$

Hence, the total time taken from a to b is the integral of the arc-length divided speed, i.e.

$$\text{Total time} = J[\boldsymbol{x}] = \int_a^b \frac{\sqrt{1 + x'(u)^2}}{\sqrt{2gx(u)}} du, \tag{2.5}$$

which is of the form (2.3) where $L(u, x, v) = \sqrt{1 + v^2}/\sqrt{2gx}$.

The problem in Example 2.1 is known as the *Brachistochrone*[1] problem, and is first posed by Johann Bernoulli in 1696. One can see from the example above that to solve this problem, it is needed to solve optimization problems over curves. A classical result due to Euler and Lagrange gives a necessary condition for optimality that allows us to solve this problem.

Theorem 2.2: Euler-Lagrange Equations

Let $\boldsymbol{x} \in C^1([a, b], \mathbb{R})$ be an extremum of J as defined in (2.3). Then, $\boldsymbol{x}$ satisfies the *Euler-Lagrange Equations*

$$\partial_x L\left(u, x(u), x'(u)\right) = \frac{d}{du}\partial_{x'} L\left(u, x(u), x'(u)\right), \qquad u \in [a, b]. \tag{2.6}$$

[1] In Greek, "Brachistochrone" is literally "shortest time".

We have deliberately left several notions rather undefined, such as the meaning of an extremum. We will revisit this slightly subtle issue in the next part. Here, we will not present a proof of the Euler-Lagrange equations, since it is not required for the rest of our discussions. A proof can be found in any standard texts on the subject of calculus of variations, say [GS00, Lib12].

Exercise 2.3: Brachistochrone Solution

Consider the Brachistochrone problem in Example 2.1. By choosing appropriate units one can set $g = 1/2$. Show that the optimal ramp shapes are *cycloids* whose parametric forms are

$$\begin{aligned} u(\theta) &= a + c(\theta - \sin\theta) \\ x(\theta) &= c(1 - \cos\theta) \end{aligned} \qquad \theta \in [0, 2\pi], \quad c > 0. \tag{2.7}$$

2.1.2 The Problem of Optimal Control

In passing to optimal control, we consider additionally two aspects of the problem, namely the type of extrema studied, as well as the setting in which such calculus of variations problems are phrased.

Throughout these notes, the word "extrema" refers to either a minimum or a maximum in the function/functional under consideration. Since maximization is just equivalent to minimization by replacing the objective function(al) with its negation, we will hereafter only discuss minima, unless otherwise stated.

We start with distinguishing different types of minima.

2.1.3 Weak vs Strong Minima

In finite-dimensional optimization, it is easy to define the notion of local and global minima. Let $\Phi : \mathbb{R}^d \to \mathbb{R}$ be a function.

- We say that x^* is a *local minimum* of Φ if there exists a $\delta > 0$ such that $\Phi(x^*) \leq \Phi(x)$ for all $\|x - x^*\| \leq \delta$.
- We say that x^* is a *global minimum* of Φ if $\Phi(x^*) \leq \Phi(x)$ for all $x \in \mathbb{R}^d$.

Hence, all global minima are automatically local minima. If Φ is differentiable, then a necessary condition for a local minima is that $\nabla\Phi(x^*) = 0$.

In extending these ideas to infinite dimensions, one needs to be slightly more careful. Notice that the definition of minima (local or global) depends on the norm $\|\cdot\|$ which gives us a sense

of closeness. We did not specify what norm we used in the finite dimensional case above, since all of them are equivalent[2].

In the infinite dimensional case of minimization of functionals, the norm we choose will affect our results, and some curve $\boldsymbol{x}$ may be a local minimum of J under one norm but not under another.

We now distinguish between two notions of minima – *weak* and *strong* minima – commonly encountered in calculus of variations and optimal control.

Let us consider for the moment that our curve $\boldsymbol{x}$ is C^1. Moreover, let us simplify things and consider one spatial dimension, so that $x(u) \in \mathbb{R}$ for $u \in [a, b]$. Now there are two natural choices of norm that we can use

- 0-Norm: $\|\boldsymbol{x}\|_0 = \sup_{u\in[a,b]} |x(u)|$.
- 1-Norm: $\|\boldsymbol{x}\|_1 = \|\boldsymbol{x}\|_0 + \sup_{u\in(a,b)} |x'(u)|$.

Each of these norms then allows us define the notion of minimum.

Definition 2.4: Strong and Weak Minima

Let $J : C^1([a, b], \mathbb{R}) \to \mathbb{R}$ be a functional and $\boldsymbol{x}^* \in C^1([a, b], \mathbb{R})$. We say that $\boldsymbol{x}^*$ is a strong local minimum if there exists a $\delta > 0$ such that $J[\boldsymbol{x}^*] \leq J[\boldsymbol{x}]$ for all $\|\boldsymbol{x} - \boldsymbol{x}^*\|_0 \leq \delta$. We say that $\boldsymbol{x}^*$ is a weak local minimum if we place the norm $\|\cdot\|_0$ by $\|\cdot\|_1$. The global versions are defined similarly.

Now, it is easy to see that any C^1 curve which is a strong minima must also be a weak minima, but the converse is not true. The Euler-Lagrange equations (Thm. 2.2) apply to weak minima, whereas we need more advanced tools to handle strong minima. We now consider a simple example below where a weak minima simply do not exist, but we will see later that this does not prevent the existence of a strong minima that is not C^1. All of these reasons motivate us to go past the setting of Euler and Lagrange and into the realm of optimal control.

Example 2.5: Piece-wise C^1 Minimizer

Consider the problem of minimizing the functional

$$J[\boldsymbol{x}] = \int_{-1}^{1} [x(u)]^2 [x'(u) - 1]^2 du, \tag{2.8}$$

subject to the boundary conditions $x(-1) = 0$ and $x(1) = 1$. Clearly, for all $\boldsymbol{x} \in C^1$ we have

[2] Let $\|\cdot\|_A$ and $\|\cdot\|_B$ be two norms on $\mathbb{R}^d$, then there exists $c \in (0, 1]$ such that $c\|x\|_A \leq \|x\|_B \leq \frac{1}{c}\|x\|$ for all $x \in \mathbb{R}^d$.

$J[\boldsymbol{x}] > 0$. But the curve

$$x(u) = \begin{cases} 0 & -1 \le u < 0 \\ x & 0 \le u \le 1 \end{cases} \tag{2.9}$$

achieves $J[\boldsymbol{x}] = 0$, but is only piece-wise C^1. In fact, C^1 curves can get closer and closer to $x(u)$ with lower and lower cost, thus a C^1 global minimizer does not exist.

2.1.4 A Dynamical View on the Calculus of Variations

Optimal control is another way to look at calculus of variations problems, in that we view things in a dynamical nature. Concretely, we may re-parameterize the curves $x(u)$ considered via infinitesimal changes in it, in the form of a control. Let us motivate this approach in the context of the Brachistochrone problem.

Example 2.6: Control Formulation of Brachistochrone

Consider the Brachistochrone problem 2.1, but this time we parameterize the ramp by a parametric form from the outset, i.e. $(u(t), x(t))$ where t is time. Then, the speed at time t is $s(u(t)) = s(t) = \sqrt{\dot{u}(t)^2 + \dot{x}(t)^2}$. Then, conservation of energy leads to

$$2gx(t) = \dot{x}(t)^2 + \dot{u}(t)^2. \tag{2.10}$$

Now, we imagine the reverse scenario treating the velocities $\dot{x}, \dot{u}$ as *controls*, by setting

$$\theta_1(t) = \dot{u}(t)/\sqrt{2gx(t)} \qquad \theta_2(t) = \dot{x}(t)/\sqrt{2gx(t)}. \tag{2.11}$$

Then, we end up with a control system that defines the equation of the ramp

$$\begin{aligned} \dot{u}(t) &= \theta_1(t)\sqrt{2gx(t)} \\ \dot{x}(t) &= \theta_2(t)\sqrt{2gx(t)} \\ \theta_1(t)^2 + \theta_2(t)^2 &= 1 \\ (u(t_0), x(t_0)) &= (a, 0), \qquad u(t_1) = b \end{aligned} \tag{2.12}$$

The cost function in this case is the time taken, so $J = \int_{t_0}^{t_1} 1dt = t_1 - t_0$.

It is worth noting that by formulating the original calculus of variations problem as a control problem, we actually gained some generality:

- It is no longer assumed that x can be written as a function of u
- It is not necessary for x to be differentiable with respect to u

2.1.5 The Optimal Control Formulation

Now, let us formulate precisely the optimal control problem in the general setting.

The Dynamics. Consider the ordinary differential equation

$$\dot{x}(t) = f(t, x(t), \theta(t)), \qquad t \in [t_0, t_1], \qquad x(t_0) = x_0. \tag{2.13}$$

Here $x(t) \in \mathbb{R}^d$ is the *state*, $\theta(t) \in \Theta \subset \mathbb{R}^m$ is the *control*, with Θ the *control set*. We will assume that the control set is closed (but it need not be bounded).

We will assume that the following conditions on f holds, unless otherwise stated:

- $f(t, x, \theta)$ is continuous in t and θ for all x
- $f(t, x, \theta)$ is continuously differentiable in x for all t, θ

These conditions are sufficient to ensure that (2.13) is well-posed by a similar result as in Theorem 1.2. See [BP07].

Remark. *The conditions outlined above are certainly not the weakest possible to imply local well-posedness of solutions, and they can be weakened in various ways (See e.g. [BP07] Ch.2).*

*We also emphasize two crucial points **not** assumed*

- *We did not assume that f is differentiable with respect to θ*
- *We did not assume that $t \mapsto \theta(t)$ is regular. In fact, in the general case we can consider θ to be a essentially bounded function of t*

The Cost Functional Let us now define the objective functionals. We will consider functionals of the form

$$J[\boldsymbol{\theta}] = \int_{t_0}^{t_1} L(t, x(t), \theta(t))dt + \Phi(t_1, x(t_1)) \tag{2.14}$$

- $L : \mathbb{R} \times \mathbb{R}^d \times \Theta \to \mathbb{R}$ is called the *running cost*
- $\Phi : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}$ is called the *terminal cost*

The Bolza Problem of Optimal Control Now, we state the *Bolza* problem of optimal control, which will be the primary object of analysis in these notes.

$$\begin{aligned} &\inf_{\boldsymbol{\theta}} J[\boldsymbol{\theta}] = \int_{t_0}^{t_1} L(t, x(t), \theta(t))dt + \Phi(t_1, x(t_1)) \\ &\text{subject to} \\ &\dot{x}(t) = f(t, x(t), \theta(t)), \qquad t \in [t_0, t_1], \qquad x(t_0) = x_0. \end{aligned} \tag{2.15}$$

For historical reasons, the case where $\Phi = 0$ (no terminal cost) is called a *Lagrange* problem, where as the case with $L = 0$ (no running cost) is called a *Mayer* problem. In optimal control theory, we often consider x_0 (initial condition) and t_0 (initial time) to be fixed. However, the terminal time t_1 can either be fixed or it can vary. Moreover, there can be a constraint set placed on the terminal state $x(t_1)$. We will mostly consider the case where the final time t_1 is fixed (so that we can neglect the t_1 dependence of Φ), and there is no constraint on the terminal state, and we will discuss how the various results may change if we consider the general case.

As with classical optimization problems, the primary object of study is optimality conditions. One differentiates between *necessary* and *sufficient* conditions for optimality. The former asks what conditions must any local/global optimum satisfy, and the latter concerns a condition that is enough to guarantee optimality. In the following sections, we will investigate each of these aspects in turn.

2.2 Pontryagin's Maximum Principle

In this section, we discuss a necessary condition for optimality – the Pontryagin's Maximum Principle (PMP) – that is a hallmark result in optimal control theory and the calculus of variations. It greatly generalizes the Euler Lagrange equations in highly nontrivial ways.

We will present the proof of the PMP in the case of fixed end time, without constraints on the terminal state. In this case, the problem is

$$\begin{aligned} &\inf_{\boldsymbol{\theta}} J[\boldsymbol{\theta}] = \int_{t_0}^{t_1} L(t, x(t), \theta(t))dt + \Phi(x(t_1)) \\ &\text{subject to} \\ &\dot{x}(t) = f(t, x(t), \theta(t)), \qquad t \in [t_0, t_1], \qquad x(t_0) = x_0. \end{aligned} \tag{2.16}$$

The proof of the PMP for this case is quite accessible, and hence we will present it in full. We will discuss the PMP for other variants of the basic formulation, but we will omit the proofs as they can be significantly more involved.

2.2.1 The Maximum Principle

To state the Pontryagin's maximum principle, we need some definitions. Let us define the *Hamiltonian*

$$\begin{aligned} &H : \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^d \times \Theta \to \mathbb{R}, \\ &H(t, x, p, \theta) = p^\top f(t, x, \theta) - L(t, x, \theta). \end{aligned} \tag{2.17}$$

For a control $\boldsymbol{\theta} = \{\theta(t) : t \in [t_0, t_1]\}$, we say it is *admissable* if $\theta(t) \in \Theta$ for all $t \in [t_0, t_1]$.

Theorem 2.7: Pontryagin's Maximum Principle

Let $\boldsymbol{\theta}^*$ be a bounded, measurable and admissable control that optimizes (2.16), and $\boldsymbol{x}^*$ be its corresponding state trajectory. Then, there exists an absolutely continuous process $\boldsymbol{p} = \{p(t) : t \in [t_0, t_1]\}$ such that

$$\dot{x}^*(t) = \nabla_p H(t, x^*(t), p^*(t), \theta^*(t)), \qquad x^*(t_0) = x_0 \tag{2.18}$$

$$\dot{p}^*(t) = -\nabla_x H(t, x^*(t), p^*(t), \theta^*(t)), \qquad p^*(t_1) = -\nabla_x \Phi(x^*(t_1)) \tag{2.19}$$

$$H(t, x^*(t), p^*(t), \theta^*(t)) \geq H(t, x^*(t), p^*(t), \theta) \quad \forall \theta \in \Theta \text{ and } a.e.\ t \in [t_0, t_1] \tag{2.20}$$

Proof 2.7: Proof of the PMP (Theorem 2.7)

The proof proceeds in several steps. To make the proof instructive, we will first assume that the function $t \mapsto \theta^*(t)$ is continuous, and we will relax this assumption at the end.

Step 1: Convert to Mayer Problem. Define an auxiliary scalar variable $x^0(t)$, with

$$\dot{x}^0(t) = L(t, x(t), \theta(t)), \qquad x^0(t_0) = 0. \tag{2.21}$$

Then, by going one dimension higher and setting $\tilde{x} = (x^0, x)$, $\tilde{f} = (L, f)$, and $\tilde{\Phi}(\tilde{x}) = \Phi(x) + x^0$ we can rewrite (2.16) as one without running cost in the new augmented coordinates. Hence, we will hereafter drop the tildes and assume without loss of generality that $L \equiv 0$.

Step 2: Needle Perturbation. Fix $\tau > 0$ and an admissible $s \in \Theta$. Define the *needle perturbation* to the optimal control

$$\theta_\varepsilon(t) = \begin{cases} s & \text{if } t \in [\tau - \varepsilon, \tau] \\ \theta^*(t) & \text{otherwise} \end{cases} \tag{2.22}$$

Let $\boldsymbol{x}_\varepsilon$ be the corresponding controlled trajectory, i.e. the solution of

$$\dot{x}_\varepsilon(t) = f(t, x_\varepsilon(t), \theta_\varepsilon(t)), \qquad x_\varepsilon(t_0) = x_0. \tag{2.23}$$

Our goal is to derive necessary conditions for which any such needle perturbation will be sub-optimal, thus resulting in a necessary condition for a strong minima in the cost functional.

Step 3: Variational Equation It is clear that $x_\varepsilon(t) = x^*(t)$ for $t \le \tau - \varepsilon$. Let us define for $t \ge \tau$

$$v(t) := \lim_{\varepsilon \to 0+} \frac{x_\varepsilon(t) - x^*(t)}{\epsilon}. \tag{2.24}$$

This measures the propagation of the effect of the needle perturbation as time increases. In particular, at $t = \tau$, $v(\tau)$ is the tangent vector of the curve $\epsilon \mapsto x_\varepsilon(\tau)$, given by

$$\begin{aligned} v(\tau) &= \lim_{\varepsilon \to 0+} \left\{ \frac{1}{\varepsilon} \int_{\tau-\varepsilon}^{\tau} f\left(t, x_\varepsilon(t), s\right) dt - \frac{1}{\varepsilon} \int_{\tau-\varepsilon}^{\tau} f\left(t, x^*(t), \theta^*(t)\right) dt \right\} \\ &= f\left(\tau, x^*(\tau), s\right) - f\left(\tau, x^*(\tau), \theta^*(\tau)\right). \end{aligned} \tag{2.25}$$

For the remaining time $t \in [\tau, T]$, x_ε follows the same ODE (2.23). Thus, by Theorem 1.3 $v(t)$ is well-defined and solves the linear variational equation

$$\dot{v}(t) = \nabla_x f\left(t, x^*(t), \theta^*(t)\right) v(t), \qquad t \in [\tau, t_1], \tag{2.26}$$

with initial condition given by (2.25). In particular, the vector $v(t_1)$ describes the variation in the end point $x_\varepsilon(t_1)$ due to the needle perturbation.

Step 4: Optimality Condition at End Point. By our assumption, the control $\boldsymbol{\theta}^*$ is optimal, hence we must have

$$\Phi(x^*(t_1)) \le \Phi(x_\varepsilon(t_1)). \tag{2.27}$$

Thus, we have

$$0 \le \lim_{\varepsilon \to 0+} \frac{\Phi\left(x_\varepsilon(t_1)\right) - \Phi\left(x^*(t_1)\right)}{\varepsilon} = \frac{d}{d\varepsilon} \Phi\left(x_\varepsilon(t_1)\right)\Big|_{\varepsilon=0+} = \nabla\Phi\left(x^*(t_1)\right) \cdot v(t_1) \tag{2.28}$$

In fact, the inequality (2.28) holds for any τ and s that characterizes the needle perturbation.

Step 5: The Adjoint Equation and the Maximum Principle. The idea is now to derive consequence that the end-point optimality condition have on each τ. To this end, we define $\boldsymbol{p}^*(t)$ as the solution of the backward Cauchy problem

$$\dot{p}^*(t) = -\nabla_x f(t, x^*(t), \theta^*(t))^\top p^*(t), \qquad p^*(t_1) = -\nabla\Phi(x^*(t_1)). \tag{2.29}$$

Then, observe that we indeed have

$$\frac{d}{dt}[p^*(t)^\top v(t)] = 0 \quad \forall t \in [\tau, t_1] \qquad \Rightarrow \qquad p^*(\tau)^\top v(\tau) = p^*(t_1)^\top v(t_1) \le 0, \tag{2.30}$$

which implies that for any $\tau \in (t_0, t_1]$ we have

$$[p^*(\tau)]^\top f(\tau, x^*(\tau), \theta^*(\tau)) \geq [p^*(\tau)]^\top f(\tau, x^*(\tau), s) \tag{2.31}$$

for any $s \in \Theta$. By continuity this also holds for $t = t_0$.

By undoing the conversion in in Step 1, we can back to a general Bolza problem by sending $\boldsymbol{p}^* \to (\boldsymbol{p}^0, \boldsymbol{p}^*)$. In particular, observe that $\dot{p}^0(t) = 0$ and $p^0(t_1) = -1$. Hence, $p^0(t) \equiv -1$. Hence, we get from the optimality condition (2.31) that

$$\underbrace{p^*(\tau)^\top f(\tau, x^*(\tau), \theta^*(\tau)) - L(\tau, x^*(\tau), \theta^*(\tau))}_{H(\tau, x^*(\tau), p^*(\tau), \theta^*(\tau))} \geq \underbrace{p^*(\tau)^\top f(\tau, x^*(\tau), s) - L(\tau, x^*(\tau), s)}_{H(\tau, x^*(\tau), p^*(\tau), s)}, \tag{2.32}$$

where $\boldsymbol{p}^*$ satisfies the adjoint equation

$$\dot{p}^*(t) = -\nabla_x H(t, x^*(t), p^*(t), \theta^*(t)), \qquad p^*(t_1) = -\nabla\Phi(x^*(t_1)). \tag{2.33}$$

Step 6: Extending to Measurable Controls. The last step is purely of technical interest, where we relax the assumption that $t \mapsto \theta^*(t)$ is continuous. By the Lebesgue differentiation theorem, we have for almost every $\tau \in (t_0, t_1)$,

$$\lim_{\varepsilon \to 0+} \frac{1}{\varepsilon} \int_{\tau-\varepsilon}^{\tau+\varepsilon} |f(t, x^*(t), \theta^*(t)) - f(\tau, x^*(\tau), \theta^*(\tau))| \, dt = 0, \tag{2.34}$$

that is, the measurable function $t \mapsto f(t, x^*(t), \theta^*(t))$ is *quasi-continuous*. Hence, the proof steps 1-5 proceeds exactly as before, only that τ is required to be a Lebesgue point, and hence the solutions of the state and adjoint equations are now only absolutely continuous, and the maximization condition (2.32) now only holds at Lebesgue points, which is almost every $t \in [t_0, t_1]$. This concludes the proof of the maximum principle. □

Let us make some remarks on the maximum principle.

- The equation (2.18) is called the *state equation*, and it is simply

  $$\dot{x}^*(t) = f(t, x^*(t), \theta^*(t)), \tag{2.35}$$

  and it describes the evolution of the state under the optimal control.

- The equation (2.19) is called the *co-state equation*, with $\boldsymbol{p}^*$ being the *co-state*. As evidenced in the proof of the PMP, the role of the co-state equation is to propagate back the optimality condition and is the adjoint of the variational equation. In fact, one can also connect $\boldsymbol{p}^*$ formally to a Lagrange multiplier enforcing the constraint of the ODE. However, this approach can only derive the weaker optimality condition that H is stationary at the optimum.

- The maximization condition (2.20) is the heart of the maximum principle. It says that an optimal control must *globally* maximize the Hamiltonian. One can regard this as

a nontrivial generalization of the Euler-Lagrange equations to handle strong extrema (See [BP07], Theorem 6.5.2), as well as a generalization of the KKT conditions to non-smooth settings.

2.2.2 Other Forms of the Maximum Principle

The reason why we called the result (2.7) a maximum *principle* is to emphasize that it is not just one result, but a class of results of similar nature. Indeed, there are many variants of the maximum principle, and we state one of them below, which is for a *fixed-end-point* variant of the Bolza problem (variation highlighted)

$$\begin{aligned} &\inf_{\boldsymbol{\theta}} J[\boldsymbol{\theta}] = \int_{t_0}^{t_1} L(t, x(t), \theta(t))dt \cancel{+\Phi(x(t_1))} \\ &\text{subject to} \\ &\dot{x}(t) = f(t, x(t), \theta(t)), \qquad t \in [t_0, t_1], \qquad x(t_0) = x_0, \qquad x(t_1) = x_1. \end{aligned} \tag{2.36}$$

In this case, the maximum principle now reads

$$\dot{x}^*(t) = \nabla_p H(t, x^*(t), p^*(t), \theta^*(t)), \qquad x^*(t_0) = x_0 \quad x^*(t_1) = x_1 \tag{2.37}$$

$$\dot{p}^*(t) = -\nabla_x H(t, x^*(t), p^*(t), \theta^*(t)), \qquad \cancel{p^*(t_1) = -\nabla_x \Phi(x^*(t_1))} \tag{2.38}$$

$$\begin{aligned} &H(t, x^*(t), p^*(t), \theta^*(t)) \geq H(t, x^*(t), p^*(t), \theta) \\ &\qquad \forall \theta \in \Theta \text{ and } a.e.\ t \in [t_0, t_1] \end{aligned} \tag{2.39}$$

Example 2.8: Piece-wise C^1 Minimizer Revisted

Let us consider the problem in Example 2.5 and we now show that the piece-wise C^1 minimizer satisfies the PMP (2.37). Notice that we can convert the problem into a fixed-end-point problem

$$\begin{aligned} &\min_{\boldsymbol{\theta}} \int_{-1}^{1} x(t)^2(\theta(t) - 1)^2 dt \\ &\text{subject to} \\ &\dot{x}(t) = \theta(t), \qquad t \in [-1, 1], \qquad x(-1) = 0, \qquad x(1) = 1. \end{aligned} \tag{2.40}$$

That is, $f(t, x, \theta) = \theta$ and running cost is $L(t, x, \theta) = x^2(\theta - 1)^2$. Writing out the PMP equations for an optimal $\boldsymbol{\theta}^*$, we get $H(t, x, p, \theta) = p\theta - x^2(1 - \theta)^2$

$$\dot{x}^*(t) = \theta^*(t), \qquad x^*(-1) = 0, \quad x^*(1) = 1, \tag{2.41}$$

$$\dot{p}^*(t) = 2x^*(t)(1 - \theta^*(t))^2, \tag{2.42}$$

$$\theta^*(t) \in \arg\max_{\theta \in \mathbb{R}} \{p^*(t)\theta - [x^*(t)]^2(1 - \theta^2)\}. \tag{2.43}$$

One can then check that the control

$$\theta^*(t) = \begin{cases} 0 & -1 \le t < 0 \\ 1 & 0 \le t \le 1 \end{cases} \tag{2.44}$$

satisfies the PMP above with $x^*(t)$ given by (2.9) and $p^*(t) = 0$.

Example 2.9: Driving a Car

Suppose we are driving a car on a straight road for $t \in [0, T]$. Let $x(t)$ denote the position of the car at time t. We suppose that we are initially at rest at the origin, and we want to drive forwards on the road. We have control over an accelerator, which we can use to accelerate or brake, but acceleration costs fuel. The problem statement is, suppose we want to drive far yet save fuel, how should we drive?

This problem can be formulated as a Bolza problem with fixed end time and free end point (2.16) as follows:

$$\begin{aligned} &\inf_{\boldsymbol{\theta}} J[\boldsymbol{\theta}] = \int_0^T \frac{1}{2} \max(0, \theta(t))^2 dt - x(T) \\ &\text{subject to} \\ &\dot{x}(t) = v(t), \qquad x(0) = 0, \\ &\dot{v}(t) = \theta(t), \qquad v(0) = 0, \\ &\theta(t) \in [-1, 1] \text{ for all } t. \end{aligned} \tag{2.45}$$

Here, the fuel cost is related to the acceleration by $\frac{1}{2}\max(0, \theta)^2$ (braking spends no fuel).

Let us now apply the PMP (2.7) to derive a solution. In this case, the Hamiltonian is

$$H(t, x, v, p_x, p_v, \theta) = p_x v + p_v \theta - \frac{1}{2}\max(0, \theta)^2. \tag{2.46}$$

Thus, the PMP equations are

$$\dot{x}^*(t) = v^*(t), \qquad x^*(0) = 0, \tag{2.47}$$
$$\dot{v}^*(t) = \theta^*(t), \qquad v^*(0) = 0, \tag{2.48}$$
$$\dot{p}_x^*(t) = 0, \qquad p_x^*(T) = 1, \tag{2.49}$$
$$\dot{p}_v^*(t) = -p_x^*(t), \qquad p_v^*(T) = 0, \tag{2.50}$$

and hence $p_x^*(t) = 1$, $p_v^*(t) = T - t$. Therefore, the optimal control is found by maximizing

the Hamiltonian:

$$\begin{aligned}\theta^*(t) &\in \arg\max_{\theta\in[-1,1]} H(t, x^*(t), v^*(t), p_x^*(t), p_v^*(t), \theta) \\ &\in \arg\max_{\theta\in[-1,1]} v^*(t) + (T-t)\theta - \frac{1}{2}\max(0,\theta)^2 \\ &= \min(T-t, 1).\end{aligned} \tag{2.51}$$

Thus, we should drive at maximum acceleration, and then ease off on the accelerator linearly.

Exercise 2.10: Driving a Better Car

As an extension of Example 2.9, we can consider the following scenario: the car has been upgraded so that the fuel cost now scales linearly with acceleration, i.e. the running cost is now $\max(0,\theta)$ instead of $\max(0,\theta)^2$. What is the optimal way to drive in this case?

2.2.3 Further Reading

Besides the basic fixed end time setting considered in the previous part, other variants of the PMP can be derived for different scenarios, including: variable end time, general set constraints on initial and final states. The proofs of these results are more involved than what is proposed above, requiring some machinery from functional analysis. For the purpose of the application cases in these notes, the previous formulation is enough. However, the interested reader is encouraged to consult optimal control references for various generalizations, or proofs under weaker assumptions e.g. [Lib12, BP07].

2.3 Hamilton-Jacobi-Bellman Equations

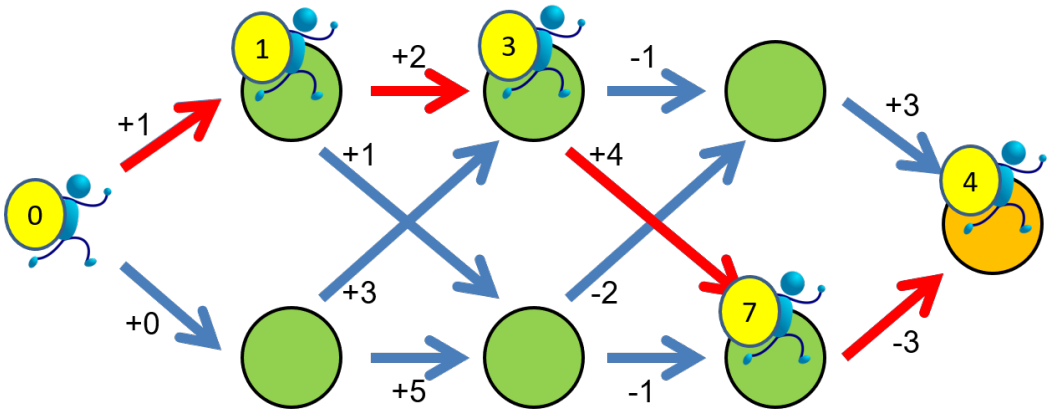
As a key alternative to the maximum principle, we now discuss another line of work that establishes necessary and sufficient conditions for optimality for optimal control problems. This presents another approach to optimal control theory that is important in its own right, as it depends on the very general idea of *dynamic programming* [Bel66].

2.3.1 Motivating Example of Dynamic Programming

Example 2.11: A Toy Maze

Consider the following maze where we want to get to the orange circle while maximizing the reward obtained along the way. When we cross each arrow, we gain a reward equal

to the number attached to that arrow. The red path shows an example path with a final reward of 4.



Suppose that there are N circles to choose from per step and T steps in total. Then, the total number of paths is N^T and grows exponentially with T. This is known as the *curse of dimensionality*.

Instead of a brute force search over all paths, we can use the principle of dynamic programming to find a solution much more efficiently. To do this, let us introduce some notation. We will index each time step in the maze by $t = 0, 1, \ldots, T$. Also, we denote by S_t the circle we step on at the t^{th} step, and R_t the reward we obtain at the t^{th} step.

Define the function

$$V(t, x) = \max \left\{ \sum_{s=t+1}^{T} R_s : S_t = x \right\}. \tag{2.52}$$

In other words, $V(t, x)$ is the best possible reward we can get starting from state x at time t. Then, we can work backwards easily!

Let us just consider the case in Example 2.11, where $S_t = 1$ or 2 for $t = 1, 2, 3$. Here, $S_t = 1$ denotes the top circle and $S_t = 2$ is the bottom circle. The initial state is $S_0 = 0$. Then, clearly we have

$$V(3, 1) = +3, \qquad V(3, 2) = -3, \tag{2.53}$$

since both cases we only have one choice – and this is the best we can do. Now, let us consider $t = 2$. Given we are at $S_2 = 1$, then there are two choices, either we go to $S_3 = 1$ or $S_3 = -1$. If we go to $S_3 = 1$ we get a reward of -1 and then, the best we can do from there would be $V(3, 1) = +3$. Similarly, if we take $S_3 = 2$ then we get $+4$ reward and the best we can do from $S_3 = 2$ is $V(3, 2) = -3$. Hence,

$$V(2, 1) = \max\{-1 + V(3, 1), +4 + V(3, 2)\} = +2. \tag{2.54}$$

A similar calculation shows that $V(2,2) = +1$. Once we know these values we can then compute $V(1,\cdot)$ and so on. This allows us to calculate backwards to obtain $V(0,0) = +6$. This is the best possible reward we can get, and we have obtained it without resorting to brute force search over all the paths! Moreover, once we have solved for $V(t,x)$ for all t, x, we can also easily find the optimal policy to navigate this maze. We simply proceed *greedily* with respect to the value function: at time t we always go the circle in the next step with the highest $V(t,x)$ plus the immediate reward.

In fact, the above methodology is known as *dynamic programming* [Bel66]. Let us look at the computational complexity of dynamic programming versus a brute force search, which takes N^T steps. In dynamic programming, we simply have to traverse the time steps once, starting the from the end. For each time step, we have to compute N values of $V(t,x)$, each depends on a linear combination of $V(t+1,s)$. Hence, for each time step we incur a computation overhead of N^2. Therefore, the entire dynamical programming procedure solves the problem in N^2T steps. This is *much* less than N^T!.

The key idea behind dynamic programming is defining the so called cost-to-go $V(t,x)$ (2.52), which allows us to derive a recursion in $V(t,x)$ that gives a solution to our original problem. The function $V(t,x)$ is also known as the *value function*, emphasizing the fact that it represents the "value" of a given state. This understanding will motivate the alternative approach we present next on optimal control.

2.3.2 The Dynamic Programming Principle

Now, let us state and prove the dynamic programming principle as applied to optimal control problems. We recall the Bolza problem with fixed end time:

$$\begin{aligned} &\inf_{\boldsymbol{\theta}} J[\boldsymbol{\theta}] = \int_{t_0}^{t_1} L(t, x(t), \theta(t))dt + \Phi(x(t_1)) \\ &\text{subject to} \\ &\dot{x}(t) = f(t, x(t), \theta(t)), \qquad t \in [t_0, t_1], \qquad x(t_0) = x_0. \end{aligned} \tag{2.55}$$

Following the idea of dynamic programming, we embed this problem in a *bigger* class of problems:

$$\begin{aligned} &V(s,z) := \inf_{\boldsymbol{\theta}} \int_{s}^{t_1} L(t, x(t), \theta(t))dt + \Phi(x(t_1)) \\ &\text{subject to} \\ &\dot{x}(t) = f(t, x(t), \theta(t)), \qquad t \in [s, t_1], \qquad x(s) = z. \end{aligned} \tag{2.56}$$

The function $V : [t_0, t_1] \times \mathbb{R}^d \to \mathbb{R}$ is called the *value function*. In words, it is the *minimum cost attainable starting from the initial condition z at time t*. Observe that $V(t_0, x_0)$ is the optimal cost of (2.55).

It may appear that we have made the problem more difficult, since we are not considering a much larger class of optimal control problems. However, it turns out that we can derive a recursion on V in terms of a partial differential equation, thereby deriving an elegant characterization of optimal controls.

Now, let us state and prove the dynamic programming principle concerning the value function for the optimal control problem.

Theorem 2.12: Dynamic Programming Principle

For every $\tau, s \in [t_0, t_1]$, $s \leq \tau$, and $z \in \mathbb{R}^d$, we have

$$V(s,z) = \inf_{\boldsymbol{\theta}} \left\{ \int_s^\tau L(t, x(t), \theta(t))dt + V(\tau, x(\tau)) \right\}, \tag{2.57}$$

where on the right hand side, $\boldsymbol{x}$ solves $\dot{x}(t) = f(t, x(t), \theta(t))$ on $t \in [s, \tau]$ with $x(s) = z$.

The meaning of the dynamic programming principle is that the optimization problem defining $V(s,z)$ can be split into two parts:

- First, solve the optimization problem on $[\tau, t_1]$ with the usual running cost L and terminal cost Φ, but for all initial conditions $z' \in \mathbb{R}^d$. This gives us the value function $V(\tau, \cdot)$
- Next, we solve the optimization problem on $[s, \tau]$ with running cost L and terminal cost $V(\tau, \cdot)$ given by the step before.

Proof 2.12: Dynamic Programming Principle

Let us denote the right hand side of (2.57) as J^τ. We first show that $J^\tau \leq V(s,z)$. We fix $\varepsilon > 0$ and choose a control $\boldsymbol{\theta} : [s, t_1] \to \Theta$ such that

$$J[\boldsymbol{\theta}] \leq V(s,z) + \varepsilon. \tag{2.58}$$

This $\boldsymbol{\theta}$ always exists since $V(s,z)$ is defined as the infimum of such $J[\boldsymbol{\theta}]$. Under this control, we have again by the definition of the value function

$$V(\tau, x(\tau)) \leq \int_\tau^{t_1} L(t, x(t), \theta(t))dt + \Phi(x(t_1)). \tag{2.59}$$

Then, we have

$$J^\tau \leq \int_s^\tau L(t, x(t), \theta(t))dt + V(\tau, x(\tau)) \tag{2.60}$$

$$\leq \int_s^{t_1} L(t, x(t), \theta(t))dt + \Phi(x(t_1)) \tag{2.61}$$

$$= J[\boldsymbol{\theta}] \leq V(s, z) + \varepsilon. \tag{2.62}$$

Since $\varepsilon > 0$ is arbitrary, we have $J^\tau \leq V(s, z)$.

Next, we show the reverse inequality. Fix $\varepsilon > 0$. Then, there exists a control $\boldsymbol{\theta}_1 : [s, \tau] \to \Theta$ such that

$$\int_s^\tau L(t, x(t), \theta_1(t))dt + V(\tau, x(\tau)) \leq J^\tau + \varepsilon. \tag{2.63}$$

Now, similarly there exists a control $\boldsymbol{\theta}_2 : [\tau, t_1] \to \Theta$ such that

$$\int_\tau^{t_1} L(t, x(t), \theta_2(t))dt + \Phi(x(t_1)) \leq V(\tau, x(\tau)) + \varepsilon. \tag{2.64}$$

This allows us to concatenate the two controls together to define

$$\theta(t) = \begin{cases} \theta_1(t) & t \in [s, \tau], \\ \theta_2(t) & t \in (\tau, t_1]. \end{cases} \tag{2.65}$$

Then, combining (2.63) and (2.65) we have

$$V(s, z) \leq J[\boldsymbol{\theta}] \leq J^\tau + 2\varepsilon, \tag{2.66}$$

and since $\varepsilon > 0$ is arbitrary, we obtain the desired result. □

2.3.3 Hamilton-Jacobi-Bellman Equations

In this section, we will derive the key result from the dynamic programming approach to optimal control problems, which establishes connections with partial differential equations, in particular the Hamilton-Jacobi equations. As defining the right sort of solutions for these equations turns out to be a slightly involved problem, we will proceed mostly formally in this section, but we will discuss at the end the key ideas in making these steps rigorous.

The basic motivation here is to derive an *infinitesimal* version of the dynamic programming principle (Theorem 2.12). To this end, we will make extensive use of Taylor expansions by assuming that $\tau = s + \Delta s$ with $\Delta s \ll 1$ in Eq. (2.57), giving the infinitesimal dynamic programming

principle

$$V(s,z) = \inf_{\boldsymbol{\theta}} \left\{ \int_s^{s+\Delta s} L(t, x(t), \theta(t))dt + V(s+\Delta s, x(s+\Delta s)) \right\}, \tag{2.67}$$

where again on the right hand side $\boldsymbol{x}$ follows the ODE

$$\dot{x}(t) = f(t, x(t), \theta(t)), \qquad t \in [s, s+\Delta s], \qquad x(s) = z. \tag{2.68}$$

Applying Taylor's expansion on the ODE, we have

$$x(s+\Delta s) = z + \int_s^{s+\Delta s} f(t, x(t), \theta(t))dt = z + f(s, z, \theta(s))\Delta s + o(\Delta s), \tag{2.69}$$

Furthermore, assuming that V is sufficiently regular, we have

$$V(s+\Delta s, x(s+\Delta s)) = V(s,z) + \partial_s V(s,z)\Delta s + [\nabla_z V(s,z)]^\top f(s,z,\theta(s))\Delta s + o(\Delta s). \tag{2.70}$$

Similarly, we can also expand the running cost

$$\int_s^{s+\Delta s} L(t, x(t), \theta(t))dt = L(s, z, \theta(s))\Delta s + o(\Delta s). \tag{2.71}$$

Combining (2.67), (2.70) and (2.71), we have

$$V(s,z) = \inf_{\boldsymbol{\theta}} \left\{ L(s,z,\theta(s))\Delta s + V(s,z) + \partial_s V(s,z)\Delta s + [\nabla_z V(s,z)]^\top f(s,z,\theta(s))\Delta s + o(\Delta s) \right\}. \tag{2.72}$$

Cancelling the term $V(s,z)$ on both sides and taking the limit $\Delta s \to 0$, the infimum over paths $\boldsymbol{\theta}$ on $t \in [s, s+\Delta s]$ becomes an infimum over a scalar $\theta = \theta(s) \in \Theta$, thus we obtain:

$$\partial_s V(s,z) + \inf_{\theta \in \Theta} \{ L(s,z,\theta) + [\nabla_z V(s,z)]^\top f(s,z,\theta) \} = 0. \tag{2.73}$$

This is known as the *Hamilton-Jacobi-Bellman* (HJB) equation for the value function. It remains to specify the boundary conditions. One can quickly observe that at time $s = t_1$, we in fact have by definition, $V(t_1, z) = \Phi(z)$.

Now, we note that the derivations above are purely formal for at least two reasons:

- We do not know if $V(s,z)$ is sufficiently regular to admit Taylor expansions.
- We do not know if the partial differential equation (2.73) is well-posed, i.e. whether it admits a unique solution, and in what sense should a solution be defined.

This is a common difficulty faced by many nonlinear partial differential equations. In this case, the Hamilton-Jacobi structure allows one to use the concept of *viscosity solutions* [CL83] as an appropriate notion of solution. Loosely speaking, viscosity solutions are a class of weak solutions to nonlinear PDEs defined by being some sense of an extremum of a sequence of smooth functions that satisfy an inequality corresponding to the PDE. One can also see them

as limits of solutions of the original PDE regularized with a diffusive term (hence the term "viscosity"). For more information on viscosity solutions, the reader is referred to [FS06]. With the notion of viscosity solutions, we in fact can put the HJB equations on a rigorous footing. Let us now state the main theorem in this section, whose proof we omit (but see [BP07], Theorem 8.7.1). For convenience we will replace (s, z) by (t, x) in the following.

Theorem 2.13: Hamilton-Jacobi-Bellman Equation

Let $V : [t_0, t_1] \times \mathbb{R}^d \to \mathbb{R}$ be the value function defined by (2.56). Then, V is the unique viscosity solution of the Hamilton-Jacobi-Bellman equation

$$\begin{aligned} &\partial_t V(t,x) + \inf_{\theta \in \Theta} \left\{ L(t,x,\theta) + [\nabla_x V(t,x)]^\top f(t,x,\theta) \right\} \qquad (t,x) \in (t_0,t_1) \times \mathbb{R}^d \\ &V(t_1,x) = \Phi(x) \end{aligned} \tag{2.74}$$

2.3.4 Implications for Optimal Control

Recall that we have the correspondence

$$V(t_0, x_0) = \inf_{\boldsymbol{\theta}} J[\boldsymbol{\theta}], \tag{2.75}$$

hence the solution of the HJB equations will give us the optimal cost that we can obtain for the Bolza problem. In fact, we will see that this gives us much more.

A Necessary Condition. It should be clear from our discussions so far that what we have formally derived is that the HJB constitutes a necessary condition for global optimality. Indeed, suppose we have a family of optimal controls $\{\boldsymbol{\theta}^*_{s,z} : s \in [t_0, t_1], z \in \mathbb{R}^d\}$ and define

$$\begin{aligned} &\widehat{V}(s,z) = \Phi(x^*_{s,z}(t_1)) + \int_s^{t_1} L(t, x^*_{s,z}(t), \theta^*_{s,z}(t))dt, \\ &\text{where } \dot{x}^*_{s,z}(t) = f(t, x^*_{s,z}(t), \theta^*_{s,z}(t)), \qquad t \in [s, t_1], \qquad x^*_{s,z}(s) = z. \end{aligned} \tag{2.76}$$

Then, by Theorem 2.13 $\widehat{V} \equiv V$ satisfies the HJB equation.

In fact, let us fix $s, \tau \in [t_0, t_1)$ and $z \in \mathbb{R}^d$. By the assumption of global optimality we can rewrite the dynamic programming principle (2.57) as

$$\begin{aligned} V(s,z) &= \inf_{\boldsymbol{\theta}} \left\{ \int_s^\tau L(t, x(t), \theta(t))dt + V(\tau, x(\tau)) \right\} \\ &= \int_s^\tau L(t, x^*_{s,z}(t), \theta^*_{s,z}(t))dt + V(\tau, x^*_{s,z}(\tau)). \end{aligned} \tag{2.77}$$

We may now proceed as before using Taylor expansions to derive an infinitesimal version of the above. Let us call $\boldsymbol{\theta}^* = \boldsymbol{\theta}^*_{t_0,x_0}$ the optimal control for our original problem, and $\boldsymbol{x}^*$ is

corresponding controlled state trajectory. Then, Taylor expanding and comparing with the usual dynamic programming principle we obtain the equality

$$\begin{aligned} -\partial_s V(s, x^*(t)) &= \min_{\theta \in \Theta} \left\{ L(t, x^*(t), \theta(t)) + [\nabla_x V(t, x^*(t))]^\top f(t, x^*(t), \theta) \right\}, \\ &= L(t, x^*(t), \theta^*(t)) + [\nabla_x V(t, x^*(t))]^\top f(t, x^*(t), \theta^*(t)), \end{aligned} \tag{2.78}$$

which we can rewrite as

$$H(t, x^*(t), -\nabla_x V(t, x^*(t)), \theta^*(t)) = \max_{\theta \in \Theta} H(t, x^*(t), -\nabla_x V(t, x^*(t)), \theta) \tag{2.79}$$

where the Hamiltonian is defined exactly as in the case of the PMP (2.17)

$$H(t, x, p, \theta) = p^\top f(t, x, \theta) - L(t, x, \theta). \tag{2.80}$$

Thus, this is similar to the statement of the PMP, except that the co-state $p^*(t)$ is now replaced by $-\nabla_x V(t, x^*(t))$. However, there is a nontrivial difference in that now, this is also a sufficient condition for global optimality, as we now show.

A Sufficient Condition. Let us now assume that a continuously differentiable function V satisfies the HJB (2.74) and moreover that a control $\widehat{\boldsymbol{\theta}} : [t_0, t_1] \to \Theta$ satisfies

$$H(t, \widehat{x}(t), -\nabla_x V(t, \widehat{x}(t)), \widehat{\theta}(t)) = \max_{\theta \in \Theta} H(t, \widehat{x}(t), -\nabla_x V(t, \widehat{x}(t)), \theta), \tag{2.81}$$

for all $t \in [t_0, t_1]$, where $\widehat{\boldsymbol{x}}$ is the state process corresponding to the control $\widehat{\boldsymbol{\theta}}$, then $\widehat{\boldsymbol{\theta}}$ is a globally optimal control that solves (2.55) with optimal cost $V(t_0, x_0)$.

To show this, observe that if we set $x = \widehat{x}(t)$ in the HJB equation for V, noting the condition (2.81), we have

$$\partial_t V(t, \widehat{x}(t)) + [\nabla_x V(t, \widehat{x}(t))]^\top f(t, \widehat{x}(t), \widehat{\theta}(t)) + L(t, \widehat{x}(t), \widehat{\theta}(t)) = 0, \tag{2.82}$$

which means

$$\frac{d}{dt} V(t, \widehat{x}(t)) + L(t, \widehat{x}(t), \widehat{\theta}(t)) = 0. \tag{2.83}$$

Integrating from t_0 to t_1 and using the boundary condition $V(t_1, x) = \Phi(x)$, we have

$$V(t_0, x_0) = \int_{t_0}^{t_1} L(t, \widehat{x}(t), \widehat{\theta}(t)) dt + \Phi(\widehat{x}(t_1)) = J[\widehat{\boldsymbol{\theta}}]. \tag{2.84}$$

On the other hand, if $\boldsymbol{\theta}$ be any other control whose trajectory is $\boldsymbol{x}$, we would have

$$\partial_t V(t, x(t)) + [\nabla_x V(t, x(t))]^\top f(t, x(t), \theta(t)) + L(t, x(t), \theta(t)) \geq 0, \tag{2.85}$$

which yields

$$0 \leq \underbrace{\int_{t_0}^{t_1} L(t, x(t), \theta(t))dt + V(t_1, x(t_1))}_{J[\boldsymbol{\theta}],\ \text{since } V(t_1,x(t_1))=\Phi(x(t_1))} - V(t_0, x_0), \tag{2.86}$$

or

$$J[\widehat{\boldsymbol{\theta}}] = V(t_0, x_0) \leq J[\boldsymbol{\theta}]. \tag{2.87}$$

This shows that $\widehat{\boldsymbol{\theta}}$ is globally optimal, with cost $V(t_0, x_0)$.

Example 2.14: Nondifferentiable Value Function ([Lib12], Example 5.2.1)

Consider the scalar control system

$$\dot{x}(t) = x(t)\theta(t), \qquad t \in [0, T], \qquad x(0) = x_0 \in \mathbb{R}, \qquad \theta(t) \in \Theta \equiv [-1, 1]. \tag{2.88}$$

We set running cost $L \equiv 0$ and terminal cost $\Phi(x) = x$. The optimal control is just $-\text{Sign}(x_0)$ if $x_0 \neq 0$, and if $x_0 = 0$ the cost is always 0. Hence, the value function is simply

$$V(t, x) = \begin{cases} e^{-(T-t)}x & \text{if } x > 0 \\ e^{T-t}x & \text{if } x < 0 \\ 0 & \text{if } x = 0 \end{cases} \tag{2.89}$$

Observe that it is not differentiable at $x = 0$.

Let us now check that the value function satisfies the HJB, which is now

$$\partial_t V(t, x) - |x\partial_x V(t, x)| = 0, \qquad V(T, x) = x. \tag{2.90}$$

Clearly, this is the case. In fact, we can derive the value function from the HJB by applying the method of characteristics (See [Eva98], Ch. 3).

Remark. *We end this section with a remark on the HJB solution. Recall that we can write the optimal control as*

$$\theta^*(t) = u(t, x^*(t)) := \min_{\theta}\{L(t, x^*(t), \theta) + [\nabla_x V(t, x^*(t))]^\top f(t, x^*(t), \theta)\}. \tag{2.91}$$

In other words, provided we can solve the HJB, the optimal control solution is of feed-back *or* closed-loop *form, meaning that it tells how to steer the system by just observing the state trajectory $\boldsymbol{x}^*$. We can contrast with the PMP, where we obtain* open-loop *controls that are pre-computed (since it also depends on the co-state) and cannot be applied on-the-fly. This is an important distinction.*

2.3.5 Further Reading

The principle of optimality has been referenced in different manners throughout the development of calculus of variations, dating back to the solution of the Brachistochrone problem of Jacob Bernoulli in 1697. The building of the Hamilton-Jacobi-Bellman theory for optimal control rests on important works of Carathéodory, Bellman and Kalman in the early 1900s. The theory is first put on rigorous footing via the introduction of viscosity solutions by Crandall and Lions [CL83]. See also [FS06] for a general exposition of viscosity solutions. Here we also omitted the interesting topic of how the HJB and the PMP are related. In fact, they can be related via the method of characteristics ([Eva98] Ch. 3): the PMP equations can be interpreted, at least formally, as characteristic equations associated with the HJB. See [Lib12], Ch. 5.2.

2.4 Stochastic Control

So far, optimal control problems are analyzed in the deterministic (ODE) setting. This follows as a natural development from classical calculus of variations. Recently, many applications requires the control of a noisy process, examples of which includes control of robots in uncertain environments, optimal execution of trading strategies, etc.

In this section, we will briefly introduce the stochastic variant of the theory of optimal control. For mathematical simplicity, we will only introduce the Hamilton-Jacobi-Bellman approach. The Pontryagin's approach turns out to be rather involved for stochastic processes, and requires some theory on backward stochastic differential equations which are beyond the scope of these notes. The interested reader may consult standard references, e.g. [YZ99] on this topic.

We assume a basic familiarity with stochastic differential equations. Unfamiliar readers may refer to textbooks e.g. [Øks03]. Note that we will not use any advanced techniques beyond Itô's formula.

2.4.1 Control of Stochastic Differential Equations

We consider the following Itô stochastic differential equation, also known as a diffusion process:

$$dX(t) = f(t, X(t), \theta(t))dt + \sigma(t, X(t), \theta(t))dW(t), \qquad X(0) = x_0, \quad t \in [0, T]. \tag{2.92}$$

Here, $X(t) \in \mathbb{R}^d$ is the stochastic process, and we use capital letter to highlight its stochastic nature, as is conventional. The process $W(t)$ is the standard Wiener process, or Brownian motion in $\mathbb{R}^p$. The matrix-valued function $\sigma : [0, T] \times \mathbb{R}^d \times \mathbb{R}^m \to \mathbb{R}^{d\times p}$ is called the *diffusion matrix*. In some applications, f is called the *drift* and σ is called the *volatility*. We hereafter assume they are uniformly Lipschitz in the state argument to guarantee existence of strong solutions to (2.92) (see [Øks03]). The initial condition $x_0 \in \mathbb{R}^d$ can be deterministic or random.

Next, we specify the cost functionals. Similar to the deterministic counterpart, we consider a terminal cost Φ and a running cost L. The only difference now is that the cost function should be

defined in an averaged sense, so we simply add an expectation. Thus, we obtain the stochastic version of the Bolza problem (2.15)

$$\begin{aligned} &\inf_{\boldsymbol{\theta}\in\mathcal{A}_{0,T}} J[\boldsymbol{\theta}] = \mathbb{E}\left[\int_0^T L(t,X(t),\theta(t))dt + \Phi(X(T))\right] \\ &\text{subject to} \\ &dX(t) = f(t,X(t),\theta(t))dt + \sigma(t,X(t),\theta(t))dW(t) \qquad t\in[0,T], \qquad X(0)=x_0. \end{aligned} \tag{2.93}$$

Here, the expectation is taken over the Wiener process, and possibly over the initial condition. The control set $\mathcal{A}_{0,T}$ is a subset of W-adapted processes, meaning that they cannot look into the future of the Wiener process. Sometimes, $\mathcal{A}_{0,T}$ is called the *admissable set* of the control problem. Note that this control problem generalizes the classical control problem of Bolza, and reduces to it if we take $\sigma = 0$.

2.4.2 The Stochastic Dynamic Programming Principle

The procedure is almost identical to the deterministic case. We define the value function

$$\begin{aligned} &V(s,z) := \inf_{\boldsymbol{\theta}\in\mathcal{A}_{s,T}} \mathbb{E}_{s,z}\left[\int_s^T L(t,x(t),\theta(t))dt + \Phi(x(T))\right] \\ &\text{subject to} \\ &dX(t) = f(t,X(t),\theta(t))dt + \sigma(t,X(t),\theta(t))dW(t) \qquad t\in[s,T], \qquad X(s)=z. \end{aligned} \tag{2.94}$$

The expectation $\mathbb{E}_{s,z}$ represents a conditional expectation on $X_s = z$.

The first main result is the stochastic version of the dynamic programming principle

Theorem 2.15: Stochastic Dynamic Programming Principle

For every $\tau, s \in [0,T]$, $s \le \tau$, with τ a stopping time[a], and $z \in \mathbb{R}^d$, we have

$$V(s,z) = \inf_{\boldsymbol{\theta}} \mathbb{E}_{s,z}\left\{\int_s^\tau L(t,X(t),\theta(t))dt + V(\tau,X(\tau))\right\}, \tag{2.95}$$

[a] A random variable τ is a stopping time if $\{\tau \le t\}$ is measurable with respect to the Brownian filtration up to time t.

The proof is identical to that of Thm. 2.12, since one can observe that whether the dynamics is an ODE or an SDE is not used in the derivation, which only requires some arguments based on optimality. The stopping time criterion is required as a technical condition so that the combined control (analogue of Eq. (2.65)) is adapted to the Wiener process. The reader should prove Thm. 2.15 as an exercise.

2.4.3 Stochastic Hamilton-Jacobi-Bellman Equation

As in the deterministic case, the form of the dynamic programming principle can be turned into a more instructive form by infinitesimal perturbations, i.e. when $\tau \approx s + \Delta s$. In the deterministic case, we obtained the HJB PDE. We now show that in the stochastic case, we obtain a very similar PDE. To avoid mathematical technicalities, we will proceed formally by simply assuming that we have the required smoothness to perform Taylor expansions using Itô's formula.

Let us recall the the Itô's formula.

Theorem 2.16: Itô's Formula

Consider the stochastic process in $\mathbb{R}^d$

$$dX(t) = f(t, X(t))dt + \sigma(t, X(t))dW(t). \tag{2.96}$$

Let $F : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}$ be twice differentiable, then

$$dF = \left[\frac{\partial F}{\partial t} + (\nabla_x F)^\top f + \frac{1}{2}\mathrm{Tr}\left[\sigma^\top(\nabla_x^2 F)\sigma\right]\right] dt + (\nabla_x F)^\top \sigma dW, \tag{2.97}$$

where all functions are evaluated at $(t, X(t))$.
One can write the above in integral form

$$\begin{aligned} F(\tau, X(\tau)) - F(s, X(s)) = &\int_s^\tau \frac{\partial F}{\partial t} + (\nabla_x F)^\top f + \frac{1}{2}\mathrm{Tr}\left[\sigma^\top(\nabla_x^2 F)\sigma\right] dt \\ &+ \int_s^\tau (\nabla_x F)^\top \sigma dW. \end{aligned} \tag{2.98}$$

We start with the stochastic dynamic programming principle with $\tau = s + \Delta s$

$$V(s, z) = \inf_{\boldsymbol{\theta}} \mathbb{E}_{s,z}\left\{\int_s^{s+\Delta s} L(t, X(t), \theta(t))dt + V(s + \Delta s, X(s + \Delta s))\right\}. \tag{2.99}$$

Applying Itô's formula (2.16), we get

$$\begin{aligned} V(s + \Delta s, X(s + \Delta s)) =& V(s, X(s)) + \int_s^{s+\Delta s} \left(\frac{\partial V}{\partial s} + f^\top \nabla_x V + \frac{1}{2}\mathrm{Tr}\left[\sigma^\top(\nabla_x^2 V)\sigma\right]\right) dt \\ &+ \int_s^{s+\Delta s} (\nabla_x V)^\top \sigma dW(t). \end{aligned} \tag{2.100}$$

Note that the last term is a Martingale and thus has zero conditional expectation. Substitute this into (2.99), we get

$$\inf_{\boldsymbol{\theta} \in \mathcal{A}_{s,s+\Delta s}} \mathbb{E}_{s,z}\left[\int_s^{s+\Delta s} \left(\frac{\partial V}{\partial s} + L + f^\top \nabla_x V + \frac{1}{2}\mathrm{Tr}\left[\sigma^\top(\nabla_x^2 V)\sigma\right]\right) dt\right] = 0. \tag{2.101}$$

Now, we can replace all the time varying terms $(t, X(t))$ in the integral By $(s, X(s)) = (s, z)$ and incur only errors of $o(\Delta s)$. Taking the limit $\Delta s \to 0$, we get

$$\partial_s V(s,z) + \inf_\theta \left\{ L(s,z,\theta) + f(s,z,\theta)^\top \nabla_x V(s,z) + \frac{1}{2}\mathrm{Tr}\left[\sigma(s,z,\theta)^\top (\nabla_x^2 V(s,z))\sigma(s,z,\theta)\right]\right\} = 0. \tag{2.102}$$

As in the deterministic case, the terminal condition is $V(T, z) = \Phi(z)$. This is the stochastic Hamilton-Jacobi-Bellman equation.

As before, our derivation is not rigorous, since the Taylor expansion based on Itô's formula requires regularity of the value function, which we typically cannot guarantee. The more mathematically precise method to handle this issue again appeals to viscosity solutions and comparison principles [CL83].

Theorem 2.17: Stochastic HJB Equation

The value function is the unique viscosity solution of the following Hamilton Jacobi Bellman equation

$$\begin{aligned} &\partial_s V(t,x) \\ &+ \inf_\theta \left\{ L(t,x,\theta) + f(t,x,\theta)^\top \nabla_x V(t,x) + \frac{1}{2}\mathrm{Tr}\left[\sigma(t,x,\theta)^\top (\nabla_x^2 V(t,x))\sigma(t,x,\theta)\right]\right\} = 0, \\ &V(t,x) = \Phi(x). \end{aligned} \tag{2.103}$$

Following the same line of argument as the deterministic case, we can show that the infimum in the HJB actually obtains a globally optimal control.

2.4.4 Further Reading

Here we only introduced the bare-basics of stochastic control. For a more complete treatment of the theory, the reader may consult many standard references, such as [FR12]. Backward SDEs, which began with the attempt to generalize Pontryagin's theory, was introduced in [Pen90, PP90]. This development led to topics beyond stochastic control, including the theory of nonlinear expectation. See [Pen10].

3 Numerical Methods for Optimal Control

3.1 Overview

So far, our discussion focused on formulating necessary and sufficient conditions for optimality for (stochastic) optimal control problems. There were two main lines of approach, namely the Pontryagin's maximum principle, and the Hamilton-Jacobi-Bellman equation. In practice, these conditions rarely lead to explicitly solvable equations. Hence, numerical solution is an important tool to study control problems.

This section gives a brief introduction to a few types of numerical algorithms that can be used to solve optimal control problems.

3.2 Numerical methods based on the PMP

We begin with methods based on the Pontryagin's maximum principle. This are also known as *indirect* methods, in that we solve a necessary condition for optimality, which typically involve integration of ODEs and small optimization problems, instead of the complete solution of a non-linear programming problem. The latter are known as *direct* methods.

3.2.1 The Method of Successive Approximations

The first such method is called the *method of successive approximations* (MSA) or the *sweeping method*. The derivation of this method is very simple. Let us consider the Bolza problem on the time interval $[0, T]$. Recall that the PMP equations take the form

$$\dot{x}^*(t) = f(t, x^*(t), \theta^*(t)), \qquad x^*(0) = x_0 \tag{3.1}$$

$$\dot{p}^*(t) = -\nabla_x H(t, x^*(t), p^*(t), \theta^*(t)), \qquad p^*(T) = -\nabla_x \Phi(x^*(T)) \tag{3.2}$$

$$\begin{aligned} &H(t, x^*(t), p^*(t), \theta^*(t)) \geq H(t, x^*(t), p^*(t), \theta) \\ &\qquad \forall \theta \in \Theta \text{ and } a.e.\ t \in [0, T] \end{aligned} \tag{3.3}$$

where the Hamiltonian has the form

$$H(t, x, p, \theta) = p^\top f(t, x, \theta) - L(t, x, \theta). \tag{3.4}$$

Notice the following:

- Three equations for three unknowns
- If we know $\boldsymbol{\theta}^*$, we can compute $\boldsymbol{x}^*$ via (3.1)
- If we know $\boldsymbol{\theta}^*, \boldsymbol{x}^*$, we can compute $\boldsymbol{p}^*$ via (3.2)
- If we know $\boldsymbol{x}^*$ and $\boldsymbol{p}^*$, we can compute $\boldsymbol{\theta}^*$ via (3.3)

Observe that this then forms a loop that we can iterate. If the iteration stops, then we have found a solution of the PMP equations, from which the $\boldsymbol{\theta}^*$ thus obtained is now a candidate optimal control. This is just the method of successive approximations. We summarize this algorithm in Alg. 1.

Algorithm 1: Method of Successive Approximations

Initialize: $\boldsymbol{\theta} \in L^\infty([0,T],\Theta)$
while *stopping criterion not reached* **do**
  $\boldsymbol{x} \leftarrow$ Solution of $\dot{x}(t) = f(t, x(t), \theta(t))$;
  $\boldsymbol{p} \leftarrow$ Solution of $\dot{p}(t) = -\nabla_x H(t, x(t), p(t), \theta(t)), \quad p(T) = -\nabla\Phi(x(T)).$;
  for $t \in [0,T]$ **do**
    $\theta(t) \leftarrow \arg\max_{\theta\in\Theta} H(t, x(t), p(t), \theta)$
  end
end
return $\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{\theta}$

The solutions of $\boldsymbol{x}, \boldsymbol{p}$ relies on the solution of differential equations. Thus, they can be solved by any ODE solution methods, such as Euler methods, Runge-Kutta methods, or symplectic methods, whichever is better suited to the dynamical problem at hand. The third step requires some discussion. While this is still an optimization problem, this is a finite-dimensional one, since we solve a *separate* optimization problem for each time t. This is often quite tractable for a variety of reasons, e.g.

1. There may be an exact solution
2. A simple sub-routine can calculate an approximate solution very quickly
3. In fact, we do not really need an exact solution to this problem

The last point stems from an error estimate that one can derive [KC62], which says that for any controls $\boldsymbol{\theta}, \boldsymbol{\theta}'$, suppose we define by $\boldsymbol{x}^{\boldsymbol{\theta}}$ the solution of the ODE (3.1) with control $\boldsymbol{\theta}$, and by $\boldsymbol{p}^{\boldsymbol{\theta}}$ the solution of the ODE (3.2) with control $\boldsymbol{\theta}$ and state $\boldsymbol{x}^{\boldsymbol{\theta}}$, then we have the following estimate (subject to some technical conditions)

$$\begin{aligned} J[\boldsymbol{\theta}'] - J[\boldsymbol{\theta}] \leq &- \int_0^T H(t, x^{\boldsymbol{\theta}}(t), p^{\boldsymbol{\theta}}(t), \theta'(t)) - H(t, x^{\boldsymbol{\theta}}(t), p^{\boldsymbol{\theta}}(t), \theta(t))dt \\ &+ C\int_0^T \|\theta'(t) - \theta(t)\|^2 dt \end{aligned} \tag{3.5}$$

One can imagine $\boldsymbol{\theta}$ as the current MSA iterate, and $\boldsymbol{\theta}'$ is the next iterate obtained by not necessarily solving the maximum of the Hamiltonian. The bound (3.5) then tell us that as long as 1) The Hamiltonian is sufficiently increased; and 2) The parameters do not change too much, then we should see a decrement in the objective functional. In fact, this turns out to be necessary, as the MSA is known to diverge if θ moves too much. These can be achieved by two methods:

1. Replacing the maximization by steepest ascent:

$$\theta(t) \leftarrow \theta(t) + \eta \nabla_\theta H(t, x(t), p(t), \theta(t)) \tag{3.6}$$

2. Replacing the maximization by a regularized problem

$$\theta(t) \leftarrow \arg\max_\theta H(t, x(t), p(t), \theta) - \lambda \|\theta - \theta(t)\|^2 \tag{3.7}$$

3.2.2 Solution of Two-point Boundary Value Problem

Observe that the MSA is an iterative algorithm, and we need to perform a potentially large number of iterations to find a solution. In the special case where the Hamiltonian maximization step has an exact solution, we can simply this procedure.

Concretely, suppose that there is a function $\xi : \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^d \to \Theta$ such that

$$H(t, x, p, \xi(t, x, p)) = \max_{\theta \in \Theta} H(t, x, p, \theta). \tag{3.8}$$

Then, the PMP equations reduce to the following set of ODEs

$$\begin{aligned} \dot{x}^*(t) &= f(t, x^*(t), \xi(t, x^*(t), p^*(t)), & x^*(0) &= x_0, \\ \dot{p}^*(t) &= -\nabla_x H(t, x^*(t), p^*(t), \xi(t, x^*(t), p^*(t))), & p^*(T) &= -\nabla\Phi(x^*(T)). \end{aligned} \tag{3.9}$$

This is simply a pair of ODEs, but this is not a usual initial value problem. The equation for the state has an initial condition, but the equation for the co-state has a terminal condition. This equation is known as a *two-point boundary value problem* (2PBVP). There are a variety of numerical methods developed for 2PBVPs. Here we outline a simple method called *shooting* method, which guess an initial condition for $\boldsymbol{p}^*$, integrates the ODEs in (3.9) (now an initial value problem) forward in time. We would obtain a solution to the 2PBVP (3.9) if the terminal co-state agrees with $-\nabla\Phi$. This can be solved by a root-finding algorithm, such as Newton's method, quasi-Newton methods (e.g. L-BFGS), or Krylov sub-space methods (e.g GMRES, Conjugate Gradient). The resulting algorithm is summarized in Alg. 2.

Remark. *It is well known in the solution of boundary value problems that shooting methods become unstable when the time horizon $[0, T]$ increases. This is because the ODEs may be ill-conditioned and a small change in the initial conditions may produce a large change in the final state, or vice versa. In either cases, the root-finding step becomes very difficult. An effective method to deal with this is to break the time interval into small sub-intervals $\{[T_n, T_{n+1}] : n = 0, \dots, N-1\}$ with $T_0 = 0$ and $T_N = T$. We can then apply shooting individually to each time interval. This is known as multiple shooting [SB13], and can be used to improve the usual shooting method for large T.*

Algorithm 2: Shooting Method for 2PBVP Formulaton of PMP Equations

Hyperparameters: RootFind (root finding algorithm), ξ (explicit solution of Hamiltonian maximization)

For $a \in \mathbb{R}^d$, Define $(\boldsymbol{x}^a, \boldsymbol{p}^a)$ as the solution of the IVP

$$\begin{aligned} x^a(t) &= f(t, x^a(t), \xi(t, x^a(t), p^a(t))), & x^a(0) &= x_0 \\ p^a(t) &= -\nabla_x H(t, x^a(t), p^a(t), \xi(t, x^a(t), p^a(t))), & p^a(0) &= a \end{aligned} \tag{3.10}$$

$a^* \leftarrow \text{RootFind}(p^a(T) + \nabla\Phi(x^a(T)))$

return $\theta^*(\cdot) = \xi(\cdot, x^{a^*}(\cdot), p^{a^*}(\cdot))$

3.3 Nonlinear Programming

Now we briefly discuss another approach which falls under the category of direct methods. Here, instead of solving for optimality conditions, we directly solve a discretized version of the optimal control problem.

First, we introduce a time-discretization size $\Delta t \ll q$. Then, we can approximate the control space $L^\infty([0, T], \Theta)$ by Θ^N with $N = T/\Delta t$ the number of discretization points.

Similarly, the ODE

$$\dot{x}(t) = f(t, x(t), \theta(t)) \tag{3.11}$$

is now discretized as

$$\frac{x_{n+1} - x_n}{\Delta t} = f(n\Delta t, x_n, \theta_n) \tag{3.12}$$

so that $x_n \approx x(n\Delta t)$.

Performing a similar discretization to the cost functional, we obtain

$$\begin{aligned} &\min_{\boldsymbol{\theta} \in \Theta^N} \Phi(x_N) + \sum_{n=0}^{N} \Delta t L(n\Delta t, x_n, \theta_n) \\ &\text{Subject to} \\ &x_{n+1} = x_n + \Delta t f(n\Delta t, x_n, \theta_n), \qquad n = 0, \dots, N-1. \end{aligned} \tag{3.13}$$

This is a constrained optimization problem of the form

$$\min_z F(z) \text{ subject to } G(z) = 0, \tag{3.14}$$

where $z = (\boldsymbol{x}, \boldsymbol{\theta})$. This is a standard equality-constrained optimization problem, and can be solved by any of the standard nonlinear programming methods. [Ber97].

3.4 Numerical Methods based on the HJB

The PMP based methods solves optimal control problems by solving coupled ODEs and a pointwise in time optimization problem. The advantage is that these methods are quite cheap to implement, especially when the ambient dimension d is large. However, one disadvantage is that the optimal control calculated is specific to the initial condition x_0. If we are given another initial condition, in general we will have to repeat the calculation again.

Recall that the dynamic programming approach and the HJB precisely avoids this issue. The control we compute from the HJB equations are of *feed-back* form

$$\begin{aligned}\theta^*(t) =& \xi(t, x^*(t)) \\ =& \arg\min_{\theta\in\Theta}\{L(t, x^*(t), \theta) + f(t, x^*(t), \theta)^\top \nabla_x V(t, x^*(t))\}\end{aligned} \tag{3.15}$$

where V is the value function computed from the HJB equations. This control can be applied to any state trajectory, whether or not it begins with x_0 is irrelevant. Thus, the HJB equations give a stronger solution to optimal control problems. This is also known as a *closed-loop* control. Let us now discuss some methods for solving the HJB equations.

Recall the HJB equations

$$\begin{aligned}&\partial_t V(t,x) + \inf_{\theta\in\Theta}\left\{L(t,x,\theta) + [\nabla_x V(t,x)]^\top f(t,x,\theta)\right\} \qquad (t,x)\in(0,T)\times\mathbb{R}^d \\ &V(T,x) = \Phi(x)\end{aligned} \tag{3.16}$$

It is customary to define the function

$$\mathcal{H}(t,x,p) = +\inf_{\theta\in\Theta}\left\{L(t,x,\theta) + p^\top f(t,x,\theta)\right\} \tag{3.17}$$

Then, the above reduces to the standard Hamilton-Jacobi equation

$$\begin{aligned}&\partial_t V(t,x) + \mathcal{H}(t,x,\nabla_x V(t,x)) = 0 \qquad (t,x)\in(0,T)\times\mathbb{R}^d \\ &V(T,x) = \Phi(x)\end{aligned} \tag{3.18}$$

In fact, we can also consider the stochastic control problem where we have

$$\begin{aligned}&\partial_t V(t,x) + \mathcal{H}(t,x,\nabla_x V(t,x), \nabla_x^2 V(t,x)) = 0 \qquad (t,x)\in(0,T)\times\mathbb{R}^d \\ &V(T,x) = \Phi(x)\end{aligned} \tag{3.19}$$

with

$$\mathcal{H}(t,x,p,Q) = \inf_{\theta\in\Theta}\left\{L(t,x,\theta) + p^\top f(t,x,\theta) + \mathrm{Tr}(\sigma(t,x,\theta)^\top Q\sigma(t,x,\theta))\right\} \tag{3.20}$$

We will now consider this case since it is more general.

Note that the PDE (3.19) is a fully nonlinear PDE, in the sense that the highest order derivative $\nabla_x^2 V$ enters non-linearly. Such equations are notoriously hard to solve due to numerical instabilities. However, in the case of control problems we can exploit the special structures that are present in Hamilton-Jacobi equations. For an introduction to the theory of HJ equations, the reader is referred to [Eva98]. The theory and methodologies presented here are from the works of [BS91].

Ellipticity. Let $\mathcal{S}_d$ denote the set of $d \times d$ real symmetric matrices. We say that a function

$$\mathcal{H} : \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^d \times \mathcal{S}_d \to \mathbb{R} \tag{3.21}$$

is *elliptic* if for any $A \geq B$ (meaning $A - B$ is positive semi-definite), we have

$$\mathcal{H}(t, x, p, A) \leq \mathcal{H}(t, x, p, B) \qquad \text{for all } (t, x, p) \in \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^d. \tag{3.22}$$

Clearly, for the case of deterministic control this is automatically satisfied since $\mathcal{H}$ is independent of the last argument. In general, it can be shown that under standard technical conditions, HJBs arising from stochastic control satisfies ellipticity.

The main results of [BS91] shows that if the ellipticity condition is inherited by the numerical discretization scheme, then the numerical method converges to the right solution. The discretized version of this result is called *monotonicity*, which we now introduce.

Monotonicity and Consistency. Let us consider a discretized version of Eq. (3.19). By setting $t \mapsto T - t$, we can regard it as a standard Cauchy (initial value) problem. Using a spatial grid size of Δx and a temporal grid size of Δt, we get

$$\begin{aligned} &S(h, t, x, V_h(t, x), [V_h]_{t,x}) = 0 && \text{in } \mathcal{G}_h \setminus \{t = 0\} \\ &V_h(0, x) = \Phi(x) && \text{on } \mathcal{G}_h \cap \{t = 0\} \end{aligned} \tag{3.23}$$

where $h = (\Delta x, \Delta t)$ and $\mathcal{G}_h = \Delta t\{0, 1, \ldots, N_T\} \times \Delta x Z^d$ with $Z \subset \mathbb{Z}$ ($|Z| = N_x$) a subset of grid points in space. Thus $\mathcal{G}_h$ denotes a regular grid in $[0, T] \times \mathbb{R}^d$. The function S encodes the discretization schemes for the differential operators, whose forms may vary. Here $V_h(t, x)$ is the approximation of the solution V at (t, x) with grid size h. For $(t, x) \in \mathcal{G}_h$, the symbol $[V_h]_{t,x}$ denotes the values of V_h at all points in $\mathcal{G}_h$ except (t, x).

Let us assume the following:

- Monotonicity: If $u \leq v$ (element-wise), then

$$S(h, t, x, r, u) \geq S(h, t, x, r, v) \tag{3.24}$$

- Consistency: For any smooth function $V(t, x)$, we have

$$\lim_{h \to 0} S(h, t, x, V(t, x), [V]_{t,x}) = \partial_t V(t, x) - \mathcal{H}(t, x, \nabla_x V(t, x), \nabla_x^2 V(t, x)) \tag{3.25}$$

  for all (t, x).

- Stability: For every $h > 0$, Eq. (3.23) admits a solution V_h and there exists a constant $C > 0$ such that $\sup_h \|V_h\| \leq C$, i.e. the solutions are bounded uniformly in h.

The following result shows that the above assumptions are enough to guarantee the convergence of solutions of the numerical scheme.

Theorem 3.1: Barles-Souganidis

If the numerical scheme (3.23) satisfies monotonicity, consistency and stability, then its solution u_h converges locally uniformly, as $h \to 0$, to the unique viscosity solution of (3.19).

Exercise 3.2: Heat Equation

Consider the 1D heat equation

$$\partial_t V(t,x) = \partial_{xx}^2 V(t,x). \tag{3.26}$$

We perform the standard forward-time, central space discretization to obtain the form (3.23) with

$$S\left(\Delta t, \Delta x, (n+1)\Delta T, i\Delta x, V_i^{n+1}, \left[V_{i-1}^n, V_i^n, V_{i+1}^n\right]\right) = \frac{V_i^{n+1} - V_i^n}{\Delta t} - \frac{V_{i+1}^n + V_{i-1}^n - 2V_i^n}{\Delta x^2}. \tag{3.27}$$

Show that the scheme is consistent, and that it is monotone and stable if the following CFL condition is satisfied:

$$\Delta t \leq \frac{1}{2}\Delta x^2. \tag{3.28}$$

Thus, Theorem 3.1 is consistent with the usual Lax-equivalence theorem for numerical analysis of PDEs.

Let us now give an example of a nonlinear HJ equation and an associated monotone scheme. Consider the 1D Hamilton-Jacobi equation

$$\partial_t V(t,x) + |\partial_x V(t,x)|^2 = 0, \qquad V(0,x) = |x|. \tag{3.29}$$

Observe that we must have $V(t,x) \geq 0$ for all t, x. Using FTCS, we can discretize this to

$$\frac{V_i^{n+1} - V_i^n}{\Delta t} + \frac{(V_{i+1}^n - V_{i-1}^n)^2}{4\Delta x^2} = 0. \tag{3.30}$$

Unlike the heat equation example, no matter what the value of the step sizes, this scheme is not monotone, and hence we cannot guarantee convergence.

An alternative is to consider the following discretization of $|\partial_x V|^2$:

$$\frac{(V_{i+1}^n - V_{i-1}^n)^2}{4\Delta x^2} - \alpha \frac{V_{i+1}^n - 2V_i^n + V_{i-1}^n}{\Delta x} \tag{3.31}$$

Note that this is still consistent, since the last term added vanishes in the limit $\Delta x \to 0$. Suppose that there exists a constant C such that $u(t,x) = |u(t,x)| \leq C$ for all t, x, then, taking $\alpha > C$ we obtain a monotone scheme. In general, constructing monotone schemes can be quite challenging, especially for higher order accuracy. The reader is referred to [Shu07].

3.5 Further Reading

We have only scratched the surface of the large topic of numerical methods for optimal control. For a good survey on direct and indirect methods based on nonlinear programming or the Pontryagin's maximum principle, the reader is referred to [Rao10]. For more information on monotone methods for HJBs, we refer to [Tou] and references therein. Lastly, the solution of HJB is very much related to the field of reinforcement learning and approximate dynamic programming [SB18]. Although most of the problems there are solved in discrete time, the techniques (e.g. value iteration, Monte-Carlo sampling) provide an alternative way to solve optimal control problems.

Bibliography

[AF13] Michael Athans and Peter L Falb. *Optimal Control: An Introduction to the Theory and Its Applications*. Courier Corporation, 2013.

[Arn12] Vladimir Igorevich Arnold. *Geometrical Methods in the Theory of Ordinary Differential Equations*, volume 250. Springer Science & Business Media, 2012.

[Bel66] Richard Bellman. Dynamic programming. *Science*, 153(3731):34–37, 1966.

[Ber97] Dimitri P Bertsekas. Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334, 1997.

[BP07] Alberto Bressan and Benedetto Piccoli. *Introduction to the Mathematical Theory of Control*, volume 2. American institute of mathematical sciences Springfield, 2007.

[BS91] Guy Barles and Panagiotis E Souganidis. Convergence of approximation schemes for fully nonlinear second order equations. *Asymptotic analysis*, 4(3):271–283, 1991.

[CL83] Michael G. Crandall and Pierre-Louis Lions. Viscosity solutions of Hamilton-Jacobi equations. *Trans. Amer. Math. Soc.*, 277(1):1–42, 1983.

[Cod12] Earl A. Coddington. *An Introduction to Ordinary Differential Equations*. Courier Corporation, 2012.

[Eva98] L C Evans. *Partial Differential Equations*. American Mathematical Society, 1998.

[FR12] Wendell H Fleming and Raymond W Rishel. *Deterministic and stochastic optimal control*, volume 1. Springer Science & Business Media, 2012.

[FS06] Wendell H. Fleming and Halil Mete Soner. *Controlled Markov Processes and Viscosity Solutions*, volume 25. Springer Science & Business Media, 2006.

[GS00] Izrail Moiseevitch Gelfand and Richard A. Silverman. *Calculus of Variations*. Courier Corporation, 2000.

[KC62] Ivan A Krylov and Felix L Chernousko. On the method of successive approximations for solution of optimal control problems. *J. Comp. Mathem. and Mathematical Physics*, 2(6), 1962.

[Lib12] Daniel Liberzon. *Calculus of Variations and Optimal Control Theory: A Concise Introduction*. Princeton University Press, Princeton ; Oxford, 2012.

[Øks03] Bernt Øksendal. Stochastic differential equations. In *Stochastic differential equations*, pages 65–84. Springer, 2003.

[Pen90] Shige Peng. A general stochastic maximum principle for optimal control problems. *SIAM Journal on control and optimization*, 28(4):966–979, 1990.

[Pen10] Shige Peng. Nonlinear expectations and stochastic calculus under uncertainty. *arXiv preprint arXiv:1002.4546*, 24, 2010.

[PP90] Etienne Pardoux and Shige Peng. Adapted solution of a backward stochastic differential equation. *Systems & Control Letters*, 14(1):55–61, 1990.

[Rao10] Anil Rao. A Survey of Numerical Methods for Optimal Control. *Advances in the Astronautical Sciences*, 135, January 2010.

[SB13] Josef Stoer and Roland Bulirsch. *Introduction to numerical analysis*, volume 12. Springer Science & Business Media, 2013.

[SB18] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[Shu07] Chi-Wang Shu. *HIGH ORDER NUMERICAL METHODS FOR TIME DEPENDENT HAMILTON-JACOBI EQUATIONS*, volume 11, pages 47–91. WORLD SCIENTIFIC, October 2007.

[Tou] Agnes Tourin. An introduction to Finite Difference methods for PDEs in Finance. page 12.

[YZ99] Jiongmin Yong and Xun Yu Zhou. *Stochastic controls: Hamiltonian systems and HJB equations*, volume 43. Springer Science & Business Media, 1999.