Communications and Control Engineering

Rushikesh Kamalapurkar
Patrick Walters · Joel Rosenfeld
Warren Dixon

Reinforcement Learning for Optimal Feedback Control

A Lyapunov-Based Approach

Springer

Communications and Control Engineering

Communications and Control Engineering is a high-level academic monograph series publishing research in control and systems theory, control engineering and communications. It has worldwide distribution to engineers, researchers, educators (several of the titles in this series find use as advanced textbooks although that is not their primary purpose), and libraries.

The series reflects the major technological and mathematical advances that have a great impact in the fields of communication and control. The range of areas to which control and systems theory is applied is broadening rapidly with particular growth being noticeable in the fields of finance and biologically-inspired control. Books in this series generally pull together many related research threads in more mature areas of the subject than the highly-specialised volumes of *Lecture Notes in Control and Information Sciences.* This series's mathematical and control-theoretic emphasis is complemented by *Advances in Industrial Control* which provides a much more applied, engineering-oriented outlook.

Publishing Ethics: Researchers should conduct their research from research proposal to publication in line with best practices and codes of conduct of relevant professional bodies and/or national and international regulatory bodies. For more details on individual ethics matters please see:
https://www.springer.com/gp/authors-editors/journal-author/journal-author-help-desk/publishing-ethics/14214.

More information about this series at http://www.springer.com/series/61

Rushikesh Kamalapurkar · Patrick Walters
Joel Rosenfeld · Warren Dixon

Reinforcement Learning for Optimal Feedback Control

A Lyapunov-Based Approach

Springer

Rushikesh Kamalapurkar
Mechanical and Aerospace Engineering
Oklahoma State University
Stillwater, OK
USA

Joel Rosenfeld
Electrical Engineering
Vanderbilt University
Nashville, TN
USA

Patrick Walters
Naval Surface Warfare Center
Panama City, FL
USA

Warren Dixon
Department of Mechanical
and Aerospace Engineering
University of Florida
Gainesville, FL
USA

ISSN 0178-5354 ISSN 2197-7119 (electronic)
Communications and Control Engineering
ISBN 978-3-319-78383-3 ISBN 978-3-319-78384-0 (eBook)
https://doi.org/10.1007/978-3-319-78384-0

Library of Congress Control Number: 2018936639

Mathematics Subject Classification (2010): 49-XX, 34-XX, 46-XX, 65-XX, 68-XX, 90-XX, 91-XX, 93-XX

Printed on acid-free paper

This Springer imprint is published by the registered company Springer International Publishing AG
part of Springer Nature
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

To my nurturing grandmother, Mangala Vasant Kamalapurkar.

—Rushikesh Kamalapurkar

To my strong and caring grandparents.

—Patrick Walters

To my wife, Laura Forest Gruss Rosenfeld, with whom I have set out on the greatest journey of my life.

—Joel Rosenfeld

To my beautiful son, Isaac Nathaniel Dixon.

—Warren Dixon

Preface

Making the best possible decision according to some desired set of criteria is always difficult. Such decisions are even more difficult when there are time constraints and can be impossible when there is uncertainty in the system model. Yet, the ability to make such decisions can enable higher levels of autonomy in robotic systems and, as a result, have dramatic impacts on society. Given this motivation, various mathematical theories have been developed related to concepts such as optimality, feedback control, and adaptation/learning. This book describes how such theories can be used to develop optimal (i.e., the best possible) controllers/policies (i.e., the decision) for a particular class of problems. Specifically, this book is focused on the development of concurrent, real-time learning and execution of approximate optimal policies for infinite-horizon optimal control problems for continuous-time deterministic uncertain nonlinear systems.

The developed approximate optimal controllers are based on reinforcement learning-based solutions, where learning occurs through an actor–critic-based reward system. Detailed attention to control-theoretic concerns such as convergence and stability differentiates this book from the large body of existing literature on reinforcement learning. Moreover, both model-free and model-based methods are developed. The model-based methods are motivated by the idea that a system can be controlled better as more knowledge is available about the system. To account for the uncertainty in the model, typical actor–critic reinforcement learning is augmented with unique model identification methods. The optimal policies in this book are derived from dynamic programming methods; hence, they suffer from the curse of dimensionality. To address the computational demands of such an approach, a unique function approximation strategy is provided to significantly reduce the number of required kernels along with parallel learning through novel state extrapolation strategies.

The material is intended for readers that have a basic understanding of nonlinear analysis tools such as Lyapunov-based methods. The development and results may help to support educators, practitioners, and researchers with nonlinear systems/control, optimal control, and intelligent/adaptive control interests working in aerospace engineering, computer science, electrical engineering, industrial

engineering, mechanical engineering, mathematics, and process engineering disciplines/industries.

Chapter 1 provides a brief introduction to optimal control. Dynamic programming-based solutions to optimal control problems are derived, and the connections between the methods based on dynamic programming and the methods based on the calculus of variations are discussed, along with necessary and sufficient conditions for establishing an optimal value function. The chapter ends with a brief survey of techniques to solve optimal control problems. Chapter 2 includes a brief review of dynamic programming in continuous time and space. In particular, traditional dynamic programming algorithms such as policy iteration, value iteration, and actor–critic methods are presented in the context of continuous-time optimal control. The role of the optimal value function as a Lyapunov function is explained to facilitate online closed-loop optimal control. This chapter also highlights the problems and limitations of existing techniques, thereby motivating the development in this book. The chapter concludes with some historic remarks and a brief classification of the available dynamic programming techniques.

In Chap. 3, online adaptive reinforcement learning-based solutions are developed for infinite-horizon optimal control problems for continuous-time uncertain nonlinear systems. A novel actor–critic–identifier structure is developed to approximate the solution to the Hamilton–Jacobi–Bellman equation using three neural network structures. The actor and the critic neural networks approximate the optimal controller and the optimal value function, respectively, and a robust dynamic neural network identifier asymptotically approximates the uncertain system dynamics. An advantage of using the actor–critic–identifier architecture is that learning by the actor, critic, and identifier is continuous and concurrent, without requiring knowledge of system drift dynamics. Convergence is analyzed using Lyapunov-based adaptive control methods. The developed actor–critic method is extended to solve trajectory tracking problems under the assumption that the system dynamics are completely known. The actor–critic–identifier architecture is also extended to generate approximate feedback-Nash equilibrium solutions to N-player nonzero-sum differential games. Simulation results are provided to demonstrate the performance of the developed actor–critic–identifier method.

Chapter 4 introduces the use of an additional adaptation strategy called concurrent learning. Specifically, a concurrent learning-based implementation of model-based reinforcement learning is used to solve approximate optimal control problems online under a finite excitation condition. The development is based on the observation that, given a model of the system, reinforcement learning can be implemented by evaluating the Bellman error at any number of desired points in the state space. By exploiting this observation, a concurrent learning-based parameter identifier is developed to compensate for uncertainty in the parameters. Convergence of the developed policy to a neighborhood of the optimal policy is established using a Lyapunov-based analysis. Simulation results indicate that the developed controller can be implemented to achieve fast online learning without the addition of ad hoc probing signals as in Chap. 3. The developed model-based reinforcement learning method is extended to solve trajectory tracking problems for

uncertain nonlinear systems and to generate approximate feedback-Nash equilibrium solutions to N-player nonzero-sum differential games.

Chapter 5 discusses the formulation and online approximate feedback-Nash equilibrium solution for an optimal formation tracking problem. A relative control error minimization technique is introduced to facilitate the formulation of a feasible infinite-horizon total-cost differential graphical game. A dynamic programming-based feedback-Nash equilibrium solution to the differential graphical game is obtained via the development of a set of coupled Hamilton–Jacobi equations. The developed approximate feedback-Nash equilibrium solution is analyzed using a Lyapunov-based stability analysis to yield formation tracking in the presence of uncertainties. In addition to control, this chapter also explores applications of differential graphical games to monitoring the behavior of neighboring agents in a network.

Chapter 6 focuses on applications of model-based reinforcement learning to closed-loop control of autonomous vehicles. The first part of the chapter is devoted to online approximation of the optimal station keeping strategy for a fully actuated marine craft. The developed strategy is experimentally validated using an autonomous underwater vehicle, where the three degrees of freedom in the horizontal plane are regulated. The second part of the chapter is devoted to online approximation of an infinite-horizon optimal path-following strategy for a unicycle-type mobile robot. An approximate optimal guidance law is obtained through the application of model-based reinforcement learning and concurrent learning-based parameter estimation. Simulation results demonstrate that the developed method learns an optimal controller which is approximately the same as an optimal controller determined by an off-line numerical solver, and experimental results demonstrate the ability of the controller to learn the approximate solution in real time.

Motivated by computational issues arising in approximate dynamic programming, a function approximation method is developed in Chap. 7 that aims to approximate a function in a small neighborhood of a state that travels within a compact set. The development is based on the theory of universal reproducing kernel Hilbert spaces over the n-dimensional Euclidean space. Several theorems are introduced that support the development of this State Following (StaF) method. In particular, it is shown that there is a bound on the number of kernel functions required for the maintenance of an accurate function approximation as a state moves through a compact set. Additionally, a weight update law, based on gradient descent, is introduced where good accuracy can be achieved provided the weight update law is iterated at a high enough frequency. Simulation results are presented that demonstrate the utility of the StaF methodology for the maintenance of accurate function approximation as well as solving the infinite-horizon optimal regulation problem. The results of the simulation indicate that fewer basis functions are required to guarantee stability and approximate optimality than are required when a global approximation approach is used.

The authors would like to express their sincere appreciation to a number of individuals whose support made the book possible. Numerous intellectual discussions and research support were provided by all of our friends and colleagues in the Nonlinear Controls and Robotics Laboratory at the University of Florida, with particular thanks to Shubhendu Bhasin, Patryk Deptula, Huyen Dinh, Keith Dupree, Nic Fischer, Marcus Johnson, Justin Klotz, and Anup Parikh. Inspiration and insights for our work were provided, in part, through discussions with and/or reading foundational literature by Bill Hager, Michael Jury, Paul Robinson, Frank Lewis (the academic grandfather or great grandfather to several of the authors), Derong Liu, Anil Rao, Kyriakos Vamvoudakis, Richard Vinter, Daniel Liberzon, and Draguna Vrabie. The research strategies and breakthroughs described in this book would also not have been possible without funding support provided from research sponsors including: NSF award numbers 0901491 and 1509516, Office of Naval Research Grants N00014-13-1-0151 and N00014-16-1-2091, Prioria Robotics, and the Air Force Research Laboratory, Eglin AFB. Most importantly, we are eternally thankful for our families who are unwavering in their love, support, and understanding.

Stillwater, OK, USA — Rushikesh Kamalapurkar
Panama City, FL, USA — Patrick Walters
Nashville, TN, USA — Joel Rosenfeld
Gainesville, FL, USA — Warren Dixon
January 2018

Contents

Symbols

Lists of abbreviations and symbols used in definitions, lemmas, theorems, and the development in the subsequent chapters.

$\mathbb{R}$	Set of real numbers
$\mathbb{R}_{\geq (\leq) a}$	Set of real numbers greater (less) than or equal to a
$\mathbb{R}_{> (<) a}$	Set of real numbers strictly greater (less) than a
$\mathbb{R}^n$	n-dimensional real Euclidean space
$\mathbb{R}^{n \times m}$	The space of $n \times m$ matrices of real numbers
$\mathbb{C}^n$	n-dimensional complex Euclidean space
$\mathcal{C}^n(\mathcal{D}_1, \mathcal{D}_2)$	The space of n-times continuously differentiable functions with domain $\mathcal{D}_1$ and codomain $\mathcal{D}_2$, and the domain and the codomain are suppressed when clear from the context
I_n	$n \times n$ Identity matrix
$\mathbf{0}_{n \times n}$	$n \times n$ Matrix of zeros
$\mathbf{1}_{n \times n}$	$n \times n$ Matrix of ones
$\mathrm{diag}\{x_1, \ldots, x_n\}$	Diagonal matrix with $x_1, \ldots, x_n$ on the diagonal
$\in$	Belongs to
$\forall$	For all
$\subset$	Subset of
$\triangleq$	Equals by definition
$f : \mathcal{D}_1 \to \mathcal{D}_2$	A function f with domain $\mathcal{D}_1$ and codomain $\mathcal{D}_2$
$\longrightarrow$	Approaches
$\mapsto$	Maps to
$\Rightarrow$	Implies that
$*$	Convolution operator
$\lvert\cdot\rvert$	Absolute value
$\lVert\cdot\rVert$	Euclidean norm
$\lVert\cdot\rVert_F$	Frobenius norm, $\lVert\theta\rVert_F = \sqrt{\mathrm{tr}(\theta^T\theta)}$
$\lVert\cdot\rVert_\infty$	Induced infinity norm

$\lambda_{\min}$	Minimum eigenvalue
$\lambda_{\max}$	Maximum eigenvalue
$\dot{x}, \ddot{x}, \ldots, x^{(i)}$	First, second, …, ith time derivative of x
$\frac{\partial f(x,y,\ldots)}{\partial y}$	Partial derivative of f with respect to y
$\nabla_y f(x, y, \ldots)$	Gradient of f with respect to y
$\nabla f(x, y, \ldots)$	Gradient of f with respect to the first argument
B_r	The ball $x \in \mathbb{R}^n \,\vert\, \|x\| < r$
$\mathrm{B}_r(y)$	The ball $x \in \mathbb{R}^n \,\vert\, \|x - y\| < r$
$\bar{A}$	Closure of a set A
$\mathrm{int}(A)$	Interior of a set A
$\partial(A)$	Boundary of a set A
$\mathbf{1}_A$	Indicator function of a set A
$\mathcal{L}_\infty(\mathcal{D}_1, \mathcal{D}_2)$	Space of uniformly essentially bounded functions with domain $\mathcal{D}_1$ and codomain $\mathcal{D}_2$, and the domain and the codomain are suppressed when clear from the context
$\mathrm{sgn}(\cdot)$	Vector and scalar signum function
$\mathrm{tr}(\cdot)$	Trace of a matrix
$\mathrm{vec}(\cdot)$	Stacks the columns of a matrix to form a vector
$\mathrm{proj}(\cdot)$	A smooth projection operator
$[\cdot]^\times$	Skew-symmetric cross product matrix

Chapter 1
Optimal Control

1.1 Introduction

The ability to learn behaviors from interactions with the environment is a desirable characteristic of a cognitive agent. Typical interactions between an agent and its environment can be described in terms of actions, states, and rewards (or penalties). Actions executed by the agent affect the state of the system (i.e., the agent and the environment), and the agent is presented with a reward (or a penalty). Assuming that the agent chooses an action based on the state of the system, the behavior (or the policy) of the agent can be described as a map from the state-space to the action-space.

Desired behaviors can be learned by adjusting the agent-environment interaction through the rewards/penalties. Typically, the rewards/penalties are qualified by a cost. For example, in many applications, the correctness of a policy is often quantified in terms of the Lagrange cost and the Mayer cost. The Lagrange cost is the cumulative penalty accumulated along a path traversed by the agent and the Mayer cost is the penalty at the boundary. Policies with lower total cost are considered better and policies that minimize the total cost are considered optimal. The problem of finding the optimal policy that minimizes the total Lagrange and Meyer cost is known as the Bolza optimal control problem.

1.2 Notation

Throughout the book, unless otherwise specified, the domain of all the functions is assumed to be $\mathbb{R}_{\geq 0}$. Function names corresponding to state and control trajectories are reused to denote elements in the range of the function. For example, the notation $u(\cdot)$ is used to denote the function $u : \mathbb{R}_{\geq t_0} \to \mathbb{R}^m$, the notation u is used to denote an arbitrary element of $\mathbb{R}^m$, and the notation $u(t)$ is used to denote the value of the function $u(\cdot)$ evaluated at time t. Unless otherwise specified, all the mathematical quantities are assumed to be time-varying, an equation of the form $g(x) = f + h(y, t)$ is interpreted as $g(x(t)) = f(t) + h(y(t), t)$ for all $t \in \mathbb{R}_{\geq 0}$, and a definition of the form $g(x, y) \triangleq f(y) + h(x)$ for functions $g : A \times B \to C$, $f : B \to C$ and

R. Kamalapurkar et al., *Reinforcement Learning for Optimal Feedback Control*, Communications and Control Engineering,
https://doi.org/10.1007/978-3-319-78384-0_1

$h : A \to C$ is interpreted as $g(x, y) \triangleq f(y) + h(x)$, $\forall (x, y) \in A \times B$. The notation $\overline{\|h\|}^{\chi}$ denotes $\sup_{\xi \in \chi} \|h(\xi)\|$, for a continuous function $h : \mathbb{R}^n \to \mathbb{R}^k$ and a compact set χ. When the compact set is clear from the context, the notation $\overline{\|h\|}$ is utilized.

1.3 The Bolza Problem

Consider a controlled dynamical system described by the initial value problem

$$\dot{x}(t) = f(x(t), u(t), t), \; x(t_0) = x_0, \tag{1.1}$$

where t_0 is the initial time, $x : \mathbb{R}_{\geq t_0} \to \mathbb{R}^n$ denotes the system state and $u : \mathbb{R}_{\geq t_0} \to U \subset \mathbb{R}^m$ denotes the control input, and U denotes the action-space.

To ensure local existence and uniqueness of Carathéodory solutions to (1.1), it is assumed that the function $f : \mathbb{R}^n \times U \times \mathbb{R}_{\geq t_0} \to \mathbb{R}^n$ is continuous with respect to t and u, and continuously differentiable with respect to x. Furthermore, the control signal, $u(\cdot)$, is restricted to be piecewise continuous. The assumptions stated here are sufficient but not necessary to ensure local existence and uniqueness of Carathéodory solutions to (1.1). For further discussion on existence and uniqueness of Carathéodory solutions, see [1, 2]. Further restrictions on the dynamical system are stated, when necessary, in subsequent chapters.

Consider a fixed final time optimal control problem where the optimality of a control policy is quantified in terms of a cost functional

$$J(t_0, x_0, u(\cdot)) = \int_{t_0}^{t_f} L(x(t; t_0, x_0, u(\cdot)), u(t), t)\, \mathrm{d}t + \Phi\left(x_f\right), \tag{1.2}$$

where $L : \mathbb{R}^n \times U \times \mathbb{R}_{\geq 0} \to \mathbb{R}$ is the Lagrange cost, $\Phi : \mathbb{R}^n \to \mathbb{R}$ is the Mayer cost, and t_f and $x_f \triangleq x\left(t_f\right)$ denote the final time and state, respectively. In (1.2), the notation $x(t; t_0, x_0, u(\cdot))$ is used to denote a trajectory of the system in (1.1), evaluated at time t, under the controller $u(\cdot)$, starting at the initial time t_0, and with the initial state x_0. Similarly, for a given policy $\phi : \mathbb{R}^n \to \mathbb{R}^n$, the short notation $x(t; t_0, x_0, \phi(x(\cdot)))$ is used to denote a trajectory under the feedback controller $u(t) = \phi(x(t; t_0, x_0, u(\cdot)))$. Throughout the book, the symbol x is also used to denote generic initial conditions in $\mathbb{R}^n$. Furthermore, when the controller, the initial time, and the initial state are understood from the context, the shorthand $x(\cdot)$ is used when referring to the entire trajectory, and the shorthand $x(t)$ is used when referring to the state of the system at time t.

The two most popular approaches to solve Bolza problems are Pontryagin's maximum principle and dynamic programming. The two approaches are independent, both conceptually and in terms of their historic development. Both the approaches are developed on the foundation of calculus of variations, which has its origins in

Newton's Minimal Resistance Problem dating back to 1685 and Johann Bernoulli's Brachistochrone problem dating back to 1696. The maximum principle was developed by the Pontryagin school at the Steklov Institute in the 1950s [3]. The development of dynamic programming methods was simultaneously but independently initiated by Bellman at the RAND Corporation [4]. While Pontryagin's maximum principle results in optimal control methods that generate optimal state and control trajectories starting from a specific state, dynamic programming results in methods that generate optimal policies (i.e., they determine the optimal decision to be made at any state of the system).

Barring some comparative remarks, the rest of this monograph will focus on the dynamic programming approach to solve Bolza problems. The interested reader is directed to the books by Kirk [5], Bryson and Ho [6], Liberzon [7], and Vinter [8] for an in-depth discussion of Pontryagin's maximum principle.

1.4 Dynamic Programming

Dynamic programming methods generalize the Bolza problem. Instead of solving the fixed final time Bolza problem for particular values of t_0, t_f, and x, a family of Bolza problems characterized by the cost functionals

$$J(t, x, u(\cdot)) = \int_t^{t_f} L(x(\tau; t, x, u(\cdot)), u(\tau), \tau)\, \mathrm{d}\tau + \Phi\left(x_f\right) \tag{1.3}$$

is solved, where $t \in \left[t_0, t_f\right]$, $t_f \in \mathbb{R}_{\geq 0}$, and $x \in \mathbb{R}^n$. A solution to the family of Bolza problems in (1.3) can be characterized using the optimal cost-to-go function (i.e., the optimal value function) $V^* : \mathbb{R}^n \times \mathbb{R}_{\geq 0} \to \mathbb{R}$, defined as

$$V^*(x, t) \triangleq \inf_{u_{[t,t_f]}} J(t, x, u(\cdot)), \tag{1.4}$$

where the notation $u_{[t,\tau]}$ for $\tau \geq t \geq t_0$ denotes the controller $u(\cdot)$ restricted to the time interval $[t, \tau]$.

1.4.1 Necessary Conditions for Optimality

In the subsequent development, a set of necessary conditions for the optimality of the value function are developed based on Bellman's principle of optimality.

Theorem 1.1 [7, p. 160] *The value function, V^*, satisfies the principle of optimality. That is, for all $(x, t) \in \mathbb{R}^n \times \left[t_0, t_f\right)$, and for all $\Delta t \in \left(0, t_f - t\right]$,*

$$V^*(x,t) = \inf_{u_{[t,t+\Delta t]}} \left\{ \int_t^{t+\Delta t} L(x(\tau), u(\tau), \tau)\, \mathrm{d}\tau + V^*(x(t+\Delta t), t+\Delta t) \right\}. \tag{1.5}$$

Proof Consider the function $V : \mathbb{R}^n \times [t_0, t_f) \to \mathbb{R}$ defined as

$$V(x,t) \triangleq \inf_{u_{[t,t+\Delta t]}} \left\{ \int_t^{t+\Delta t} L(x(\tau), u(\tau), \tau)\, \mathrm{d}\tau + V^*(x(t+\Delta t), t+\Delta t) \right\}.$$

Based on the definition in (1.4)

$$V(x,t) = \inf_{u_{[t,t+\Delta t]}} \left\{ \int_t^{t+\Delta t} L(x(\tau), u(\tau), \tau)\, \mathrm{d}\tau + \inf_{u_{[t+\Delta t, t_f]}} J(t+\Delta t, x(t+\Delta t), u(\cdot)) \right\}.$$

Using (1.3) and combining the integrals,

$$V(x,t) = \inf_{u_{[t,t+\Delta t]}} \left\{ \inf_{u_{[t+\Delta t, t_f]}} J(t,x,u(\cdot)) \right\} \geq \inf_{u_{[t,t_f]}} J(t,x,u(\cdot)) = V^*(x,t). \tag{1.6}$$

Thus, $V(x,t) \geq V^*(x,t)$. On the other hand, by the definition of the infimum, for all $\epsilon > 0$, there exists a controller $u_\epsilon(\cdot)$ such that

$$V^*(x,t) + \epsilon \geq J(t,x,u_\epsilon(\cdot)).$$

Let x_ϵ denote the trajectory corresponding to u_ϵ. Then,

$$\begin{aligned} J(t,x,u_\epsilon) &= \int_t^{t+\Delta t} L(x_\epsilon(\tau), u_\epsilon(\tau), \tau)\, \mathrm{d}\tau + J(t+\Delta t, x_\epsilon(t+\Delta t), u_\epsilon), \\ &\geq \int_t^{t+\Delta t} L(x_\epsilon(\tau), u_\epsilon(\tau), \tau)\, \mathrm{d}\tau + V(x_\epsilon(t+\Delta t), t+\Delta t) \geq V(x,t). \end{aligned}$$

Thus, $V(x,t) \leq V^*(x,t)$, which, along with (1.6), implies $V(x,t) = V^*(x,t)$. □

Under the assumption that $V^* \in \mathcal{C}^1\left(\mathbb{R}^n \times [t_0, t_f), \mathbb{R}\right)$, the optimal value function can be shown to satisfy

$$0 = -\nabla_t V^*(x,t) - \inf_{u \in U} \left\{ L(x,u,t) + \nabla_x V^{*T}(x,t)\, f(x,u,t) \right\},$$

for all $t \in \left[t_0, t_f\right)$ and all $x \in \mathbb{R}^n$, with the boundary condition $V^*\left(x, t_f\right) = \Phi\left(x\right)$, for all $x \in \mathbb{R}^n$. In fact, the Hamilton–Jacobi–Bellman equation along with a Hamiltonian maximization condition completely characterize the solution to the family of Bolza problems.

1.4.2 Sufficient Conditions for Optimality

Theorem 1.2 presents necessary and sufficient conditions for a function to be the optimal value function.

Theorem 1.2 *Let* $V^* \in \mathcal{C}^1\left(\mathbb{R}^n \times \left[t_0, t_f\right], \mathbb{R}\right)$ *denote the optimal value function. Then, a function* $V : \mathbb{R}^n \times \left[t_0, t_f\right] \to \mathbb{R}$ *is the optimal value function (i.e.,* $V\left(x, t\right) = V^*\left(x, t\right)$ *for all* $(x, t) \in \mathbb{R}^n \times \left[t_0, t_f\right]$*) if and only if:*

1. $V \in \mathcal{C}^1\left(\mathbb{R}^n \times \left[t_0, t_f\right], \mathbb{R}\right)$ and V satisfies the Hamilton–Jacobi–Bellman equation

$$0 = -\nabla_t V\left(x, t\right) - \inf_{u \in U}\left\{L\left(x, u, t\right) + \nabla_x V^T\left(x, t\right) f\left(x, u, t\right)\right\}, \tag{1.7}$$

   for all $t \in \left[t_0, t_f\right)$ and all $x \in \mathbb{R}^n$, with the boundary condition $V\left(x, t_f\right) = \Phi\left(x\right)$, for all $x \in \mathbb{R}^n$.
2. For all $x \in \mathbb{R}^n$, there exists a controller $u\left(\cdot\right)$, such that the function V, the controller $u\left(\cdot\right)$, and the trajectory $x\left(\cdot\right)$ of (1.1) under $u\left(\cdot\right)$ with the initial condition $x\left(t_0\right) = x$, satisfy the equation

$$\begin{aligned} L\left(x\left(t\right), u\left(t\right), t\right) + \nabla_x V^T\left(x\left(t\right), t\right) f\left(x\left(t\right), u\left(t\right), t\right) \\ = \min_{\hat{u} \in U}\left\{L\left(x\left(t\right), \hat{u}, t\right) + \nabla_x V^T\left(x\left(t\right), t\right) f\left(x\left(t\right), \hat{u}, t\right)\right\}, \end{aligned} \tag{1.8}$$

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

Proof See [7, Sect. 5.1.4]. □

1.5 The Unconstrained Affine-Quadratic Regulator

The focus of this monograph is on unconstrained infinite-horizon total cost Bolza problems for nonlinear systems that are affine in the controller and cost functions that are quadratic in the controller. That is, optimal control problems where the system dynamics are of the form

$$\dot{x}(t) = f(x(t)) + g(x(t))\,u(t), \tag{1.9}$$

where $f : \mathbb{R}^n \to \mathbb{R}^m$ and $g : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ are locally Lipschitz functions, and the cost functional is of the form

$$J(t_0, x_0, u(\cdot)) = \int_{t_0}^{\infty} r(x(\tau; t_0, x_0, u(\cdot)), u(\tau))\,\mathrm{d}\tau, \tag{1.10}$$

where the local cost $r : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is defined as

$$r(x, u) \triangleq Q(x) + u^T R u, \tag{1.11}$$

where $Q : \mathbb{R}^n \to \mathbb{R}$ is a positive definite function and $R \in \mathbb{R}^{m\times m}$ is a symmetric positive definite matrix.

To ensure that the optimal control problem is well-posed, the minimization problem is constrained to the set of admissible controllers (see [9, Definition 1]), and the existence of at least one admissible controller is assumed. It is further assumed that the optimal control problem has a continuously differentiable value function. This assumption is valid for a large class of problems. For example, most unconstrained infinite horizon optimal control problems with smooth data have smooth value functions. However, there is a large class of relevant optimal control problems for which the assumption fails. For example, problems with bounded controls and terminal costs typically have nondifferentiable value functions. Dynamic programming-based solutions to such problems are characterized by viscosity solutions to the corresponding Hamilton–Jacobi–Bellman equation. For further details on viscosity solutions to Hamilton–Jacobi–Bellman equations, the reader is directed to [10] and [11].

Provided the aforementioned assumptions hold, the optimal value function is time-independent, That is,

$$V^*(x) \triangleq \inf_{u_{[t,\infty]}} J(t, x, u(\cdot)), \tag{1.12}$$

for all $t \in \mathbb{R}_{\geq t_0}$. Furthermore, the Hamiltonian minimization condition in (1.8) is satisfied by the controller $u(t) = u^*(x(t))$, where the policy $u^* : \mathbb{R}^n \to \mathbb{R}^m$ is defined as

$$u^*(x) = -\frac{1}{2} R^{-1} g^T(x) \left(\nabla_x V^*(x)\right)^T. \tag{1.13}$$

Hence, assuming that an optimal controller exists, a complete characterization of the solution to the optimal control problem can be obtained using the Hamilton–Jacobi–Bellman equation.

Remark 1.3 While infinite horizon optimal control problems naturally arise in feedback control application where stability is of paramount importance, path planning applications often involve finite-horizon optimal control problems. The method of

dynamic programming has extensively been studied for finite horizon problems [12–20], although such problems are out of the scope of this monograph.

Remark 1.4 The control-affine model in (1.9) is applicable to a wide variety of electro-mechanical systems. In particular, any linear system and any Euler-Lagrange nonlinear system that has a known and invertible inertia matrix can be modeled using a control-affine model. Examples include industrial manipulators, fully actuated autonomous underwater and air vehicles (where the range of operation does not include singular configurations), kinematic wheels, etc. Computation of the policy in (1.13) exploits the control-affine nature of the dynamics, and knowledge of the control effectiveness function, g, is required to implement the policy. The methods detailed in this monograph can be extended to systems with uncertain control effectiveness functions and to nonaffine systems (cf. [21–28]).

The following theorem fully characterizes solutions to optimal control problems for affine systems.

Theorem 1.5 *For a nonlinear system described by (1.9), $V^* \in \mathcal{C}^1(\mathbb{R}^n, \mathbb{R})$ is the optimal value function corresponding to the cost functional (1.10) if and only if it satisfies the Hamilton–Jacobi–Bellman equation*

$$r\left(x, u^*(x)\right) + \nabla_x V^*(x)\left(f(x) + g(x)\, u^*(x)\right) = 0, \qquad \forall x \in \mathbb{R}^n, \tag{1.14}$$

with the boundary condition $V(0) = 0$. Furthermore, the optimal controller can be expressed as the state-feedback law $u(t) = u^*(x(t))$.

Proof For each $x \in \mathbb{R}^n$ we have

$$\frac{\partial\left(r(x,u) + \nabla_x V^*(x)\left(f(x) + g(x)\, u\right)\right)}{\partial u} = 2u^T R + \nabla_x V^*(x)\, g(x).$$

hence, $u = -\frac{1}{2} R^{-1} g^T(x)\left(\nabla_x V^*(x)\right)^T = u^*(x)$ extremizes $r(x,u) + \nabla_x V^*(x)\left(f(x) + g(x)\, u\right)$. Furthermore, the Hessian

$$\frac{\partial^2\left(r(x,u) + \nabla_x V^*(x)\left(f(x) + g(x)\, u\right)\right)}{\partial^2 u} = 2R$$

is positive definite. Hence, $u = u^*(x)$ minimizes $r(x,u) + \nabla_x V^*(x)\left(f(x) + g(x)\, u\right)$. Hence, Eq. (1.14) is equivalent to the conditions in (1.7) and (1.8). □

1.6 Input Constraints

The Bolza problem detailed in the previous section is an unconstrained optimal control problem. In practice, actuators are limited in the amount of control effort they can produce. Let u_i denote the i^{th} component of the control vector u. The

affine-quadratic formulation can be extended to systems with actuator constraints of the form $|u_i(t)| \leq \overline{u},\ \forall t \in \mathbb{R}_{\geq t_0},\ \forall i = 1, \ldots, m$ using a non-quadratic penalty function first introduced in [29].

Let $\psi : \mathbb{R} \to \mathbb{R}$ be a strictly monotonically increasing continuously differentiable function such that the $\operatorname{sgn}(\psi(a)) = \operatorname{sgn}(a),\ \forall a \in \mathbb{R}$, and $|\psi(a)| \leq \overline{u}$ (e.g., $\psi(a) = \tanh(a)$). Consider a cost function of the form $r(x, u) = Q(x) + U(u)$, where

$$U(u) \triangleq 2\sum_{i=1}^{m} r_i \left(\int_0^{u_i} \psi^{-1}(\xi)\, \mathrm{d}\xi \right), \tag{1.15}$$

and r_i denotes the i^{th} diagonal element of the matrix R.

The following theorem characterizes the solutions to optimal control problems for affine systems with actuation constraints.

Theorem 1.6 *For a nonlinear system described by (1.9), $V^* \in \mathcal{C}^1(\mathbb{R}^n, \mathbb{R})$ is the optimal value function corresponding to the cost functional in (1.10), with the control penalty in (1.15), if and only if it satisfies the Hamilton–Jacobi–Bellman equation*

$$r(x, \phi(x)) + \nabla_x V^*(x)(f(x) + g(x)\phi(x)) = 0,\ \forall x \in \mathbb{R}^n, \tag{1.16}$$

with the boundary condition $V^*(0) = 0$, where $\phi(x) \triangleq -\psi\left(\frac{1}{2}R^{-1}g^T(x)(\nabla_x V^*(x))^T\right)$. Furthermore, the optimal controller can be expressed as the state-feedback law $u(t) = \overline{u}^*(x(t))$, where

$$\overline{u}^*(x) \triangleq -\psi\left(\frac{1}{2}R^{-1}g^T(x)\left(\nabla_x V^*(x)\right)^T\right).$$

Proof For each $x \in \mathbb{R}^n$,

$$\frac{\partial\left(r(x,u) + \nabla_x V^*(x)(f(x) + g(x)u)\right)}{\partial u} = 2\psi^{-1}\left(u^T\right)R + \nabla_x V^*(x)g(x).$$

hence, $u = -\psi\left(\frac{1}{2}R^{-1}g^T(x)(\nabla_x V^*(x))^T\right)$ extremizes $r(x,u) + \nabla_x V^*(x)(f(x) + g(x)u)$. Furthermore, the Hessian is

$$\frac{\partial^2\left(r(x,u) + \nabla_x V^*(x)(f(x) + g(x)u)\right)}{\partial^2 u} = 2R\begin{bmatrix} \nabla_{u_1}\psi^{-1}(u_1) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \nabla_{u_m}\psi^{-1}(u_m) \end{bmatrix}.$$

Provided the function ψ is strictly monotonically increasing, the Hessian is positive definite. Hence, $u = -\psi\left(\frac{1}{2}R^{-1}g^T(x)(\nabla_x V^*(x))^T\right)$ minimizes $r(x,u) + \nabla_x V^*(x)(f(x) + g(x)u)$. □

1.7 Connections with Pontryagin's Maximum Principle

To apply Pontryagin's maximum principle to the unconstrained affine-quadratic regulator, define the Hamiltonian $H : \mathbb{R}^n \times U \times \mathbb{R}^n \to \mathbb{R}$ as

$$H(x, u, p) = p^T (f(x) + g(x) u) - r(x, u).$$

Pontryagin's maximum principle provides the following necessary condition for optimality.

Theorem 1.7. [3, 5, 7] *Let $x^* : \mathbb{R}_{\geq t_0} \to \mathbb{R}^n$ and $u^* : \mathbb{R}_{\geq t_0} \to U$ denote the optimal state and control trajectories corresponding to the optimal control problem in Sect. 1.5. Then there exists a trajectory $p^* : \mathbb{R}_{\geq t_0} \to \mathbb{R}^n$ such that $p^*(t) \neq 0$ for some $t \in \mathbb{R}_{\geq t_0}$ and x^* and p^* satisfy the equations*

$$\dot{x}^*(t) = \left(\nabla_p H\left(x^*(t), u^*(t), p^*(t)\right)\right)^T,$$
$$\dot{p}^*(t) = -\left(\nabla_x H\left(x^*(t), u^*(t), p^*(t)\right)\right)^T,$$

with the boundary condition $x^*(t_0) = x_0$. Furthermore, the Hamiltonian satisfies

$$H\left(x^*(t), u^*(t), p^*(t)\right) \geq H\left(x^*(t), u, p^*(t)\right), \tag{1.17}$$

for all $t \in \mathbb{R}_{\geq t_0}$ and $u \in U$, and

$$H\left(x^*(t), u^*(t), p^*(t)\right) = 0, \tag{1.18}$$

for all $t \in \mathbb{R}_{\geq t_0}$.

Proof See, e.g., [7, Sect. 4.2]. □

Under further assumptions on the state and the control trajectories, and on the functions f, g, and r, the so-called *natural* transversality condition $\lim_{t\to\infty} p(t) = 0$ can be obtained (cf. [30–32]). The natural transversality condition does not hold in general for infinite horizon optimal control problems. For some illustrative counterexamples and further discussion, see [30–35].

A quick comparison of Eq. (1.14) and (1.18) suggests that the optimal costate should satisfy

$$p^*(t) = -\left(\nabla_x V\left(x^*(t)\right)\right)^T. \tag{1.19}$$

Differentiation of (1.19) with respect to time yields

$$\dot{p}^*(t) = -\nabla_x \left(\nabla_x V\left(x^*(t)\right)\right)^T \left(f\left(x^*(t)\right) + g\left(x^*(t)\right) u(t)\right).$$

Differentiation of (1.14) with respect to the state yields

$$\left(f\left(x^*\right)+g\left(x^*\right)u\right)^T\nabla_x\left(\nabla_x V\left(x^*\right)\right)^T=-\nabla_x V\left(x^*\right)\left(\nabla_x f\left(x^*\right)+\nabla_x g\left(x^*\right)u\right)-\nabla_x r\left(x^*,u^*\right).$$

Provided the second derivatives are continuous, then $\nabla_x\left(\nabla_x V\left(x^*\right)\right)^T=\left(\nabla_x\left(\nabla_x V\left(x^*\right)\right)^T\right)^T$. Hence, the time derivative of the costate can be computed as

$$\begin{aligned}\dot{p}^*(t)&=-\left(\nabla_x\left(f\left(x^*(t)\right)+g\left(x^*(t)\right)u(t)\right)\right)^T\left(\nabla_x V\left(x^*(t)\right)\right)^T-\left(\nabla_x r\left(x^*(t),u^*(t)\right)\right)^T,\\&=-\left(\nabla_x H\left(x^*(t),u^*(t),p^*(t)\right)\right)^T.\end{aligned}$$

Therefore, the expression of the costate in (1.19) satisfies Theorem 1.7. The relationship in (1.19) implies that the costate is the sensitivity of the optimal value function to changes in the system state trajectory. Furthermore, the Hamiltonian maximization conditions in (1.8) and (1.17) are equivalent. Dynamic programming and Pontryagin's maximum principle methods are therefore closely related. However, there are a few key differences between the two methods.

The solution in (1.13) obtained using dynamics programming is a feedback law. That is, dynamic programming can be used to generate a policy that can be used to close the control loop. Furthermore, once the Hamilton–Jacobi–Bellman equation is solved, the resulting feedback law is guaranteed to be optimal for any initial condition of the dynamical system. On the other hand, Pontryagin's maximum principle generates the optimal state, costate, and control trajectories for a given initial condition. The controller must be implemented in an open-loop manner. Furthermore, if the initial condition changes, the optimal solution is no longer valid and the optimal control problem needs to be solved again.

Since dynamic programming generates a feedback law, it provides much more information than the maximum principle. However, the added benefit comes at a heavy computational cost. To generate the optimal policy, the Hamilton–Jacobi–Bellman partial differential equation must be solved. In general, numerical methods to solve the Hamilton–Jacobi–Bellman equation grow exponentially in numerical complexity with increasing dimensionality. That is, dynamic programming suffers from the so-called Bellman's curse of dimensionality.

1.8 Further Reading

1.8.1 Numerical Methods

One way to develop optimal controllers for general nonlinear systems is to use numerical methods [5]. A common approach is to formulate the optimal control problem in terms of a Hamiltonian and then to numerically solve a two point boundary value problem for the state and co-state equations [36, 37]. Another approach is to cast the optimal control problem as a nonlinear programming problem via direct transcription and then solve the resulting nonlinear program [30, 38–42]. Numerical methods are offline, do not generally guarantee stability, or optimality, and are often

open-loop. These issues motivate the desire to find an analytical solution. Developing analytical solutions to optimal control problems for linear systems is complicated by the need to solve an algebraic Riccati equation or a differential Riccati equation. Developing analytical solutions for nonlinear systems is even further complicated by the sufficient condition of solving a Hamilton–Jacobi–Bellman partial differential equation, where an analytical solution may not exist in general. If the nonlinear dynamics are exactly known, then the problem can be simplified at the expense of optimality by solving an algebraic Riccati equations through feedback-linearization methods (cf. [43–47]).

Alternatively, some investigators temporarily assume that the uncertain system could be feedback-linearized, solve the resulting optimal control problem, and then use adaptive/learning methods to asymptotically learn the uncertainty [48–51] (i.e., asymptotically converge to the optimal controller). The nonlinear optimal control problem can also be solved using inverse optimal control [52–61] by circumventing the need to solve the Hamilton–Jacobi–Bellman equation. By finding a control Lyapunov function, which can be shown to also be a value function, an optimal controller can be developed that optimizes a derived cost. However, since the cost is derived rather than specified by mission/task objectives, this approach is not explored in this monograph. Optimal control-based algorithms such as state dependent Riccati equations [62–65] and model-predictive control [66–72] have been widely utilized for control of nonlinear systems. However, both state dependent Riccati equations and model-predictive control are inherently model-based. Furthermore, due to nonuniqueness of state dependent linear factorization in state dependent Riccati equations-based techniques, and since the optimal control problem is solved over a small prediction horizon in model-predictive control, they generally result in suboptimal policies. Furthermore, model-predictive control approaches are computationally intensive, and closed-loop stability of state dependent Riccati equations-based methods is generally impossible to establish a priori and has to be established through extensive simulation.

1.8.2 Differential Games and Equilibrium Solutions

A multitude of relevant control problems can be modeled as multi-input systems, where each input is computed by a player, and each player attempts to influence the system state to minimize its own cost function. In this case, the optimization problem for each player is coupled with the optimization problem for other players. Hence, in general, an optimal solution in the usual sense does not exist for such problems, motivating the formulation of alternative optimality criteria.

Differential game theory provides solution concepts for many multi-player, multi-objective optimization problems [73–75]. For example, a set of policies is called a Nash equilibrium solution to a multi-objective optimization problem if none of the players can improve their outcome by changing their policy while all the other players abide by the Nash equilibrium policies [76]. Thus, Nash equilibrium solutions

provide a secure set of strategies, in the sense that none of the players have an incentive to diverge from their equilibrium policy. Hence, Nash equilibrium has been a widely used solution concept in differential game-based control techniques. For an in-depth discussion on Nash equilibrium solutions to differential game problems, see Chaps. 3 and 4.

Differential game theory is also employed in multi-agent optimal control, where each agent has its own decentralized objective and may not have access to the entire system state. In this case, graph theoretic models of the information structure are utilized in a differential game framework to formulate coupled Hamilton–Jacobi equations (c.f. [77]). Since the coupled Hamilton–Jacobi equations are difficult to solve, reinforcement learning is often employed to get an approximate solution. Results such as [77, 78] indicate that adaptive dynamic programming can be used to generate approximate optimal policies online for multi-agent systems. For an in-depth discussion on the use of graph theoretic models of information structure in a differential game framework, see Chap. 5

1.8.3 Viscosity Solutions and State Constraints

A significant portion of optimal control problems of practical importance require the solution to satisfy state constraints. For example, autonomous vehicles operating in complex contested environments are required to observe strict static (e.g., due to policy or mission objectives or known obstacles/structures in the environment) and dynamic (e.g., unknown and then sensed obstacles, moving obstacles) no-entry zones. The value functions corresponding to optimal control problems with state constraints are generally not continuously differentiable, and may not even be differentiable everywhere. Hence, for these problems, the Hamilton–Jacobi–Bellman equation fails to admit classical solutions, and alternative solution concepts are required. A naive generalization would be to require a function to satisfy the Hamilton–Jacobi–Bellman equation almost everywhere. However, the naive generalization is not useful for optimal control since such generalized solutions are often unrelated to the value function of the corresponding optimal control problem.

An appropriate notion of generalized solutions to the Hamilton–Jacobi–Bellman equation, called viscosity solutions, was developed in [10]. It has been established that under the condition that the value function is continuous, it is a solution to the Hamilton–Jacobi–Bellman equation. Some uniqueness results are also available under further assumptions on the value function. For a detailed treatment of viscosity solutions to Hamilton–Jacobi–Bellman equations, see [79].

Various methods have been developed to approximate viscosity solutions to Hamilton–Jacobi–Bellman equations [79–81]; however, these methods are offline, require knowledge of the system dynamics, and are computationally expensive. Online computation of approximate classical solutions to the Hamilton–Jacobi–Bellman equation is achieved through dynamic programming methods. Dynamic programming methods in continuous state and time rely on a differential [82] or an

integral [83] formulation of the temporal difference error (called the Bellman error). The corresponding reinforcement learning algorithms are generally designed to minimize the Bellman error. Since such minimization yields estimates of generalized solutions, but not necessarily viscosity solutions, to the Hamilton–Jacobi–Bellman equation, reinforcement learning in continuous time and space for optimal control problems with state constraints has largely remained an open area of research.

References

1. Carathéodory C (1918) Vorlesungen über reelle Funktionen. Teubner
2. Coddington EA, Levinson N (1955) Theory of ordinary differential equations. McGraw-Hill
3. Pontryagin LS, Boltyanskii VG, Gamkrelidze RV, Mishchenko EF (1962) The mathematical theory of optimal processes. Interscience, New York
4. Bellman R (1954) The theory of dynamic programming. Technical report, DTIC Document
5. Kirk D (2004) Optimal control theory: an introduction. Dover, Mineola
6. Bryson AE, Ho Y (1975) Applied optimal control: optimization, estimation, and control. Hemisphere Publishing Corporation
7. Liberzon D (2012) Calculus of variations and optimal control theory: a concise introduction. Princeton University Press
8. Vinter R (2010) Optimal control. Springer Science & Business Media
9. Beard R, Saridis G, Wen J (1997) Galerkin approximations of the generalized Hamilton–Jacobi–Bellman equation. Automatica 33:2159–2178
10. Crandall M, Lions P (1983) Viscosity solutions of Hamilton–Jacobi equations. Trans Am Math Soc 277(1):1–42
11. Bardi M, Dolcetta I (1997) Optimal control and viscosity solutions of Hamilton–Jacobi–Bellman equations. Springer
12. Cimen T, Banks SP (2004) Global optimal feedback control for general nonlinear systems with nonquadratic performance criteria. Syst Control Lett 53(5):327–346
13. Cheng T, Lewis FL, Abu-Khalaf M (2007) A neural network solution for fixed-final time optimal control of nonlinear systems. Automatica 43(3):482–490
14. Cheng T, Lewis FL, Abu-Khalaf M (2007) Fixed-final-time-constrained optimal control of nonlinear systems using neural network HJB approach. IEEE Trans Neural Netw 18(6):1725–1737
15. Kar I, Adhyaru D, Gopal M (2009) Fixed final time optimal control approach for bounded robust controller design using Hamilton–Jacobi–Bellman solution. IET Control Theory Appl 3(9):1183–1195
16. Wang F, Jin N, Liu D, Wei Q (2011) Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with epsilon-error bound. IEEE Trans Neural Netw 22:24–36
17. Heydari A, Balakrishnan SN (2012) An optimal tracking approach to formation control of nonlinear multi-agent systems. In: Proceedings of AIAA guidance, navigation and control conference
18. Wang D, Liu D, Wei Q (2012) Finite-horizon neuro-optimal tracking control for a class of discrete-time nonlinear systems using adaptive dynamic programming approach. Neurocomputing 78(1):14–22
19. Zhao Q, Xu H, Jagannathan S (2015) Neural network-based finite-horizon optimal control of uncertain affine nonlinear discrete-time systems. IEEE Trans Neural Netw Learn Syst 26(3):486–499
20. Li C, Liu D, Li H (2015) Finite horizon optimal tracking control of partially unknown linear continuous-time systems using policy iteration. IET Control Theory Appl 9(12):1791–1801

21. Ge SS, Zhang J (2003) Neural-network control of nonaffine nonlinear system with zero dynamics by state and output feedback. IEEE Trans Neural Netw 14(4):900–918
22. Wang D, Liu D, Wei Q, Zhao D, Jin N (2012) Optimal control of unknown nonaffine nonlinear discrete-time systems based on adaptive dynamic programming. Automatica 48(8):1825–1832
23. Zhang X, Zhang H, Sun Q, Luo Y (2012) Adaptive dynamic programming-based optimal control of unknown nonaffine nonlinear discrete-time systems with proof of convergence. Neurocomputing 91:48–55
24. Liu D, Huang Y, Wang D, Wei Q (2013) Neural-network-observer-based optimal control for unknown nonlinear systems using adaptive dynamic programming. Int J Control 86(9):1554–1566
25. Bian T, Jiang Y, Jiang ZP (2014) Adaptive dynamic programming and optimal control of nonlinear nonaffine systems. Automatica 50(10):2624–2632
26. Yang X, Liu D, Wei Q, Wang D (2015) Direct adaptive control for a class of discrete-time unknown nonaffine nonlinear systems using neural networks. Int J Robust Nonlinear Control 25(12):1844–1861
27. Kiumarsi B, Kang W, Lewis FL (2016) H-∞ control of nonaffine aerial systems using off-policy reinforcement learning. Unmanned Syst 4(1):1–10
28. Song R, Wei Q, Xiao W (2016) Off-policy neuro-optimal control for unknown complex-valued nonlinear systems based on policy iteration. Neural Comput Appl 46(1):85–95
29. Lyashevskiy S, Meyer AU (1995) Control system analysis and design upon the Lyapunov method. In: Proceedings of the American control conference, vol 5, pp 3219–3223
30. Fahroo F, Ross IM (2008) Pseudospectral methods for infinite-horizon nonlinear optimal control problems. J Guid Control Dyn 31(4):927–936
31. Pickenhain S (2014) Hilbert space treatment of optimal control problems with infinite horizon. In: Bock GH, Hoang PX, Rannacher R, Schlöder PJ (eds) Modeling, simulation and optimization of complex processes - HPSC 2012: Proceedings of the fifth international conference on high performance scientific computing, 5–9 March 2012, Hanoi, Vietnam. Springer International Publishing, Cham, pp 169–182
32. Tauchnitz N (2015) The pontryagin maximum principle for nonlinear optimal control problems with infinite horizon. J Optim Theory Appl 167(1):27–48
33. Halkin H (1974) Necessary conditions for optimal control problems with infinite horizons. Econometrica pp 267–272
34. Aseev SM, Kryazhimskii A (2007) The pontryagin maximum principle and optimal economic growth problems. Proc Steklov Inst Math 257(1):1–255
35. Aseev SM, Veliov VM (2015) Maximum principle for infinite-horizon optimal control problems under weak regularity assumptions. Proc Steklov Inst Math 291(1):22–39
36. von Stryk O, Bulirsch R (1992) Direct and indirect methods for trajectory optimization. Ann Oper Res 37(1):357–373
37. Betts JT (1998) Survey of numerical methods for trajectory optimization. J Guid Control Dyn 21(2):193–207
38. Hargraves CR, Paris S (1987) Direct trajectory optimization using nonlinear programming and collocation. J Guid Control Dyn 10(4):338–342
39. Huntington GT (2007) Advancement and analysis of a gauss pseudospectral transcription for optimal control. Ph.D thesis, Department of Aeronautics and Astronautics, MIT
40. Rao AV, Benson DA, Darby CL, Patterson MA, Francolin C, Huntington GT (2010) Algorithm 902: GPOPS, A MATLAB software for solving multiple-phase optimal control problems using the Gauss pseudospectral method. ACM Trans Math Softw 37(2):1–39
41. Darby CL, Hager WW, Rao AV (2011) An hp-adaptive pseudospectral method for solving optimal control problems. Optim Control Appl Methods 32(4):476–502
42. Garg D, Hager WW, Rao AV (2011) Pseudospectral methods for solving infinite-horizon optimal control problems. Automatica 47(4):829–837
43. Freeman R, Kokotovic P (1995) Optimal nonlinear controllers for feedback linearizable systems. In: Proceedings of the American control conference, pp 2722–2726