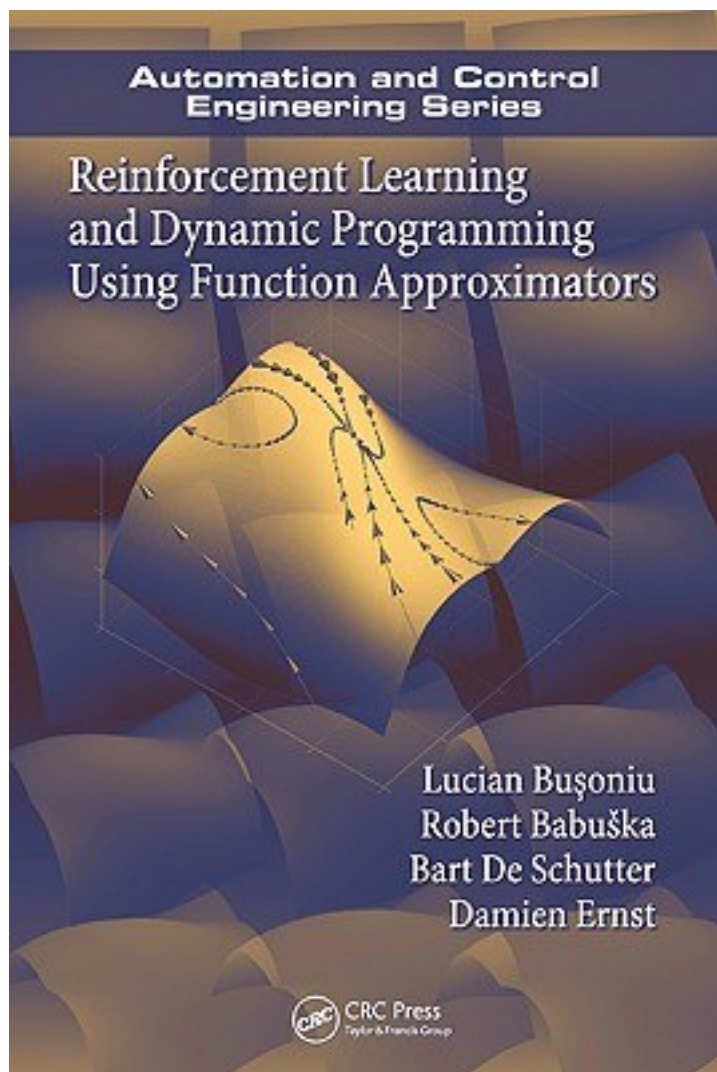


Reinforcement Learning and Dynamic Programming Using Function Approximators



[Reinforcement Learning and Dynamic Programming Using Function Approximators_下载链接1_](#)

著者:Busoniu, Lucian

出版者:CRC Press

出版时间:2017-7-28

装帧:

isbn:9781439821084

From household appliances to applications in robotics, engineered systems involving complex dynamics can only be as effective as the algorithms that control them. While Dynamic Programming (DP) has provided researchers with a way to optimally solve decision and control problems involving complex dynamic systems, its practical value was limited by algorithms that lacked the capacity to scale up to realistic problems.

However, in recent years, dramatic developments in Reinforcement Learning (RL), the model-free counterpart of DP, changed our understanding of what is possible. Those developments led to the creation of reliable methods that can be applied even when a mathematical model of the system is unavailable, allowing researchers to solve challenging control problems in engineering, as well as in a variety of other disciplines, including economics, medicine, and artificial intelligence.

Reinforcement Learning and Dynamic Programming Using Function Approximators provides a comprehensive and unparalleled exploration of the field of RL and DP. With a focus on continuous-variable problems, this seminal text details essential developments that have substantially altered the field over the past decade. In its pages, pioneering experts provide a concise introduction to classical RL and DP, followed by an extensive presentation of the state-of-the-art and novel methods in RL and DP with approximation. Combining algorithm development with theoretical guarantees, they elaborate on their work with illustrative examples and insightful comparisons. Three individual chapters are dedicated to representative algorithms from each of the major classes of techniques: value iteration, policy iteration, and policy search. The features and performance of these algorithms are highlighted in extensive experimental studies on a range of control applications.

The recent development of applications involving complex systems has led to a surge of interest in RL and DP methods and the subsequent need for a quality resource on the subject. For graduate students and others new to the field, this book offers a thorough introduction to both the basics and emerging methods. And for those researchers and practitioners working in the fields of optimal and adaptive control, machine learning, artificial intelligence, and operations research, this resource offers a combination of practical algorithms, theoretical analysis, and comprehensive examples that they will be able to adapt and apply to their own work.

Access the authors' website at www.dcsc.tudelft.nl/rlbook/ for additional material, including computer code used in the studies and information concerning new developments.

作者介绍:

Lucian Busoniu is a postdoctoral fellow at the Delft Center for Systems and Control of Delft University of Technology, in the Netherlands. He received his PhD degree (cum laude) in 2009 from the Delft University of Technology, and his MSc degree in 2003 from the Technical University of Cluj-Napoca, Romania. His current research interests include reinforcement learning and dynamic programming with function approximation, intelligent and learning techniques for control problems, and multi-agent learning.

Robert Babuska Robert Babuska is a full professor at the Delft Center for Systems and

Control of Delft University of Technology in the Netherlands. He received his PhD degree (cum laude) in Control in 1997 from the Delft University of Technology, and his MSc degree (with honors) in Electrical Engineering in 1990 from Czech Technical University, Prague. His research interests include fuzzy systems modeling and identification, data-driven construction and adaptation of neuro-fuzzy systems, model-based fuzzy control and learning control. He is active in applying these techniques in robotics, mechatronics, and aerospace.

Bart De Schutter Bart De Schutter is a full professor at the Delft Center for Systems and Control and at the Marine & Transport Technology department of Delft University of Technology in the Netherlands. He received the PhD degree in Applied Sciences (summa cum laude with congratulations of the examination jury) in 1996 from K.U. Leuven, Belgium. His current research interests include multi-agent systems, hybrid systems control, discrete-event systems, and control of intelligent transportation systems.

Damien Ernst Damien Ernst received the MSc and PhD degrees from the University of Lige in 1998 and 2003, respectively. He is currently a Research Associate of the Belgian FRS-FNRS and he is affiliated with the Systems and Modeling Research Unit of the University of Lige. Damien Ernst spent the period 2003--2006 with the University of Lige as a Postdoctoral Researcher of the FRS-FNRS and held during this period positions as visiting researcher at CMU, MIT and ETH. He spent the academic year 2006--2007 working at Suplec (France) as professor. His main research interests are in the fields of power system dynamics, optimal control, reinforcement learning, and design of dynamic treatment regimes.

目录: 1. Introduction

- 1.1 The dynamic programming and reinforcement learning problem
- 1.2 Approximation in dynamic programming and reinforcement learning
- 1.3 About this book
- 2. An introduction to dynamic programming and reinforcement learning
 - 2.1 Introduction
 - 2.2 Markov decision processes
 - 2.2.1 Deterministic setting
 - 2.2.2 Stochastic setting
 - 2.3 Value iteration
 - 2.3.1 Model-based value iteration
 - 2.3.2 Model-free value iteration and the need for exploration
 - 2.4 Policy iteration
 - 2.4.1 Model-based policy iteration
 - 2.4.2 Model-free policy iteration
 - 2.5 Policy search
 - 2.6 Summary and discussion
- 3. Dynamic programming and reinforcement learning in large and continuous spaces
 - 3.1 Introduction
 - 3.2 The need for approximation in large and continuous spaces
 - 3.3 Approximation architectures
 - 3.3.1 Parametric approximation
 - 3.3.2 Nonparametric approximation
 - 3.3.3 Comparison of parametric and nonparametric approximation
 - 3.3.4 Remarks
 - 3.4 Approximate value iteration
 - 3.4.1 Model-based value iteration with parametric approximation

- 3.4.2 Model-free value iteration with parametric approximation
- 3.4.3 Value iteration with nonparametric approximation
- 3.4.4 Convergence and the role of nonexpansive approximation
- 3.4.5 Example: Approximate Q-iteration for a DC motor
- 3.5 Approximate policy iteration
 - 3.5.1 Value iteration-like algorithms for approximate policy evaluation
 - 3.5.2 Model-free policy evaluation with linearly parameterized approximation
 - 3.5.3 Policy evaluation with nonparametric approximation
 - 3.5.4 Model-based approximate policy evaluation with rollouts
 - 3.5.5 Policy improvement and approximate policy iteration
 - 3.5.6 Theoretical guarantees
 - 3.5.7 Example: Least-squares policy iteration for a DC motor
- 3.6 Finding value function approximators automatically
 - 3.6.1 Basis function optimization
 - 3.6.2 Basis function construction
 - 3.6.3 Remarks
- 3.7 Approximate policy search
 - 3.7.1 Policy gradient and actor-critic algorithms
 - 3.7.2 Gradient-free policy search
 - 3.7.3 Example: Gradient-free policy search for a DC motor
- 3.8 Comparison of approximate value iteration, policy iteration, and policy search
- 3.9 Summary and discussion
- 4. Approximate value iteration with a fuzzy representation
 - 4.1 Introduction
 - 4.2 Fuzzy Q-iteration
 - 4.2.1 Approximation and projection mappings of fuzzy Q-iteration
 - 4.2.2 Synchronous and asynchronous fuzzy Q-iteration
 - 4.3 Analysis of fuzzy Q-iteration
 - 4.3.1 Convergence
 - 4.3.2 Consistency
 - 4.3.3 Computational complexity
 - 4.4 Optimizing the membership functions
 - 4.4.1 A general approach to membership function optimization
 - 4.4.2 Cross-entropy optimization
 - 4.4.3 Fuzzy Q-iteration with cross-entropy optimization of the membership functions
 - 4.5 Experimental study
 - 4.5.1 DC motor: Convergence and consistency study
 - 4.5.2 Two-link manipulator: Effects of action interpolation, and comparison with fitted Q-iteration
 - 4.5.3 Inverted pendulum: Real-time control
 - 4.5.4 Car on the hill: Effects of membership function optimization
 - 4.6 Summary and discussion
- 5. Approximate policy iteration for online learning and continuous-action control
 - 5.1 Introduction
 - 5.2 A recapitulation of least-squares policy iteration
 - 5.3 Online least-squares policy iteration
 - 5.4 Online LSPI with prior knowledge
 - 5.4.1 Online LSPI with policy approximation
 - 5.4.2 Online LSPI with monotonic policies
 - 5.5 LSPI with continuous-action, polynomial approximation
 - 5.6 Experimental study
 - 5.6.1 Online LSPI for the inverted pendulum
 - 5.6.2 Online LSPI for the two-link manipulator

- 5.6.3 Online LSPI with prior knowledge for the DC motor
- 5.6.4 LSPI with continuous-action approximation for the inverted pendulum
- 5.7 Summary and discussion
- 6. Approximate policy search with cross-entropy optimization of basis functions
 - 6.1 Introduction
 - 6.2 Cross-entropy optimization
 - 6.3 Cross-entropy policy search
 - 6.3.1 General approach
 - 6.3.2 Cross-entropy policy search with radial basis functions
 - 6.4 Experimental study
 - 6.4.1 Discrete-time double integrator
 - 6.4.2 Bicycle balancing
 - 6.4.3 Structured treatment interruptions for HIV infection control
 - 6.5 Summary and discussion
- Appendix A. Extremely randomized trees
 - A.1 Structure of the approximator
 - A.2 Building and using a tree
- Appendix B. The cross-entropy method
 - B.1 Rare-event simulation using the cross-entropy method
 - B.2 Cross-entropy optimization
- Symbols and abbreviations
- Bibliography
- List of algorithms
- Index
 - • • • • ([收起](#))

[Reinforcement Learning and Dynamic Programming Using Function Approximators_下载链接1](#)

标签

强化学习

增强学习

运筹学

数学

教材

动态规划

优化

评论

[Reinforcement Learning and Dynamic Programming Using Function Approximators](#) [下载链接1](#)

书评

[Reinforcement Learning and Dynamic Programming Using Function Approximators](#) [下载链接1](#)