Rollout, Policy Iteration, and Distributed Reinforcement Learning

by

Dimitri P. Bertsekas

Arizona State University
and
Massachusetts Institute of Technology

WWW site for book information and orders

http://www.athenasc.com

Athena Scientific, Belmont, Massachusetts

Athena Scientific
Post Office Box 805
Nashua, NH 03060
U.S.A.

Email: info@athenasc.com
WWW: http://www.athenasc.com

Cover photography by Dimitri Bertsekas.
Stars over the Stata Center at MIT (built on the location of the old Building 20 where Claude Shannon had his first office as a professor in 1956).

Publisher's Cataloging-in-Publication Data

Bertsekas, Dimitri P.
Rollout, Policy Iteration, and Distributed Reinforcement Learning
Includes Bibliography and Index
1. Mathematical Optimization. 2. Dynamic Programming. I. Title.
QA402.5 .B465 2020 519.703 00-91281

ISBN-10: 1-886529-07-8, ISBN-13: 978-1-886529-07-6

2nd Printing (includes revisions and updates)

ABOUT THE AUTHOR

Dimitri Bertsekas studied Mechanical and Electrical Engineering at the National Technical University of Athens, Greece, and obtained his Ph.D. in system science from the Massachusetts Institute of Technology. He has held faculty positions with the Engineering-Economic Systems Department, Stanford University, and the Electrical Engineering Department of the University of Illinois, Urbana. Since 1979 he has been teaching at the Electrical Engineering and Computer Science Department of the Massachusetts Institute of Technology (M.I.T.), where he is McAfee Professor of Engineering. In 2019, he joined the School of Computing, Informatics, and Decision Systems Engineering at the Arizona State University, Tempe, AZ, as Fulton Professor of Computational Decision Making.

Professor Bertsekas' teaching and research have spanned several fields, including deterministic optimization, dynamic programming and stochastic control, large-scale and distributed computation, artificial intelligence, and data communication networks. He has authored or coauthored numerous research papers and eighteen books, several of which are currently used as textbooks in MIT classes, including "Dynamic Programming and Optimal Control," "Data Networks," "Introduction to Probability," and "Nonlinear Programming."

Professor Bertsekas was awarded the INFORMS 1997 Prize for Research Excellence in the Interface Between Operations Research and Computer Science for his book "Neuro-Dynamic Programming" (co-authored with John Tsitsiklis), the 2001 AACC John R. Ragazzini Education Award, the 2009 INFORMS Expository Writing Award, the 2014 AACC Richard Bellman Heritage Award, the 2014 INFORMS Khachiyan Prize for Life-Time Accomplishments in Optimization, the 2015 MOS/SIAM George B. Dantzig Prize, and the 2022 IEEE Control Systems Award. In 2018 he shared with his coauthor, John Tsitsiklis, the 2018 INFORMS John von Neumann Theory Prize for the contributions of the research monographs "Parallel and Distributed Computation" and "Neuro-Dynamic Programming." Professor Bertsekas was elected in 2001 to the United States National Academy of Engineering for "pioneering contributions to fundamental research, practice and education of optimization/control theory, and especially its application to data communication networks."

ATHENA SCIENTIFIC

OPTIMIZATION AND COMPUTATION SERIES

1. Rollout, Policy Iteration, and Distributed Reinforcement Learning, by Dimitri P. Bertsekas, 2020, ISBN 978-1-886529-07-6, 480 pages
2. Reinforcement Learning and Optimal Control, by Dimitri P. Bertsekas, 2019, ISBN 978-1-886529-39-7, 388 pages
3. Abstract Dynamic Programming, 2nd Edition, by Dimitri P. Bertsekas, 2018, ISBN 978-1-886529-46-5, 360 pages
4. Dynamic Programming and Optimal Control, Two-Volume Set, by Dimitri P. Bertsekas, 2017, ISBN 1-886529-08-6, 1270 pages
5. Nonlinear Programming, 3rd Edition, by Dimitri P. Bertsekas, 2016, ISBN 1-886529-05-1, 880 pages
6. Convex Optimization Algorithms, by Dimitri P. Bertsekas, 2015, ISBN 978-1-886529-28-1, 576 pages
7. Convex Optimization Theory, by Dimitri P. Bertsekas, 2009, ISBN 978-1-886529-31-1, 256 pages
8. Introduction to Probability, 2nd Edition, by Dimitri P. Bertsekas and John N. Tsitsiklis, 2008, ISBN 978-1-886529-23-6, 544 pages
9. Convex Analysis and Optimization, by Dimitri P. Bertsekas, Angelia Nedić, and Asuman E. Ozdaglar, 2003, ISBN 1-886529-45-0, 560 pages
10. Network Optimization: Continuous and Discrete Models, by Dimitri P. Bertsekas, 1998, ISBN 1-886529-02-7, 608 pages
11. Network Flows and Monotropic Optimization, by R. Tyrrell Rockafellar, 1998, ISBN 1-886529-06-X, 634 pages
12. Introduction to Linear Optimization, by Dimitris Bertsimas and John N. Tsitsiklis, 1997, ISBN 1-886529-19-1, 608 pages
13. Parallel and Distributed Computation: Numerical Methods, by Dimitri P. Bertsekas and John N. Tsitsiklis, 1997, ISBN 1-886529-01-9, 718 pages
14. Neuro-Dynamic Programming, by Dimitri P. Bertsekas and John N. Tsitsiklis, 1996, ISBN 1-886529-10-8, 512 pages
15. Constrained Optimization and Lagrange Multiplier Methods, by Dimitri P. Bertsekas, 1996, ISBN 1-886529-04-3, 410 pages
16. Stochastic Optimal Control: The Discrete-Time Case, by Dimitri P. Bertsekas and Steven E. Shreve, 1996, ISBN 1-886529-03-5, 330 pages

Contents

Preface

We know the past but cannot control it. We control the future but cannot know it.

Claude Shannon

In this research monograph we discuss the solution of large and challenging multistage decision problems using methods of reinforcement learning (RL for short), also referred to by other names such as approximate dynamic programming and neuro-dynamic programming. We will focus on a subset of methods which are based on the idea of *policy iteration*, i.e., starting from some policy and generating one or more improved policies.

If just one improved policy is generated, this is called *rollout*, which, based on broad and consistent computational experience, appears to be one of the simplest and most reliable of all RL methods. Rollout is also well-suited for on-line model-free implementation and on-line replanning. Approximate policy iteration can be viewed as repeated application of rollout. This is one of the most prominent types of RL methods. It can be implemented using data generated by the system itself, a process known as *self-learning*, and value and policy approximation architectures, including neural networks. Both rollout and policy iteration are related to the classical Newton's method for iterative optimization, which in turn explains their associated large cost improvements and fast convergence.

Approximate policy iteration is more ambitious than rollout, but it is a strictly off-line method, and it is generally far more computationally intensive (of course rollout may also require a lot of on-line computation). This motivates the use of parallel and distributed computation. One of the purposes of the monograph is to discuss distributed (possibly asynchronous) methods that relate to rollout and policy iteration, both in the context of an exact and an approximate implementation involving neural networks or other approximation architectures.

One of the contributions of the monograph is to develop variants of rollout and policy iteration for problems with a multiagent structure, where the control consists of multiple components, each associated with a separate agent. In particular, we introduce a new approach to lookahead simplification through the use of *multiagent rollout*, which allows the dramatic reduction of the computational requirements for one-step lookahead when the control consists of multiple components, and connects with the theme of distributed asynchronous implementation.

Multiagent rollout also has a strong connection with a well-developed body of research with a long history: the theory of teams and the notion of person-by-person optimality. In particular, we develop an infinite horizon dynamic programming methodology, which includes value and policy iteration methods that converge to a person-by-person optimal policy. While our multiagent schemes are based on fully shared agent information, they are also well suited as a starting point for approximations, in the context of on-line autonomous decision making by multiple agents each coordinating in varying degrees with the other agents. In this context, agent information that is not shared by other agents, is appropriately estimated, with the estimates being treated as if they were exact.

Several of the ideas that we develop in some depth in this monograph have been central in the implementation of recent high profile successes, such as the AlphaZero program for playing chess, Go, and other games. In addition to the fundamental process of successive policy iteration/improvement, this program includes the use of deep neural networks for representation of both value functions and policies, the extensive use of large scale parallelization, and the simplification of lookahead minimization, through methods involving Monte Carlo tree search and pruning of the lookahead tree. In this monograph, we also focus on policy iteration, value and policy neural network representations, parallel and distributed computation, and lookahead simplification. Thus while there are significant differences, the principal design ideas that form the core of this monograph are shared by the AlphaZero architecture, except that we develop these ideas in a broader and less application-specific framework.

Another subject that we deal with in some detail is model predictive control (MPC for short), one of the most prominent control system design methods at present. One of the reasons is that classical forms of MPC are closely related to (and indeed can be viewed as) rollout algorithms, thereby providing a connection with reinforcement learning, which is beneficial in two ways. On one hand the MPC context provides rich crossfertilization opportunities with the analytical and algorithmic ideas of rollout and RL; for example the notion of sequential improvement in rollout is intimately connected to Lyapunov stability analysis in MPC, and the target tube ideas that are central in MPC may prove useful in the context of constrained rollout and policy iteration. On the other hand the dynamic programming and RL methodologies point the way to extensions of MPC based on self-learning, approximate policy iteration, simulation, the treatment of stochastic and set membership uncertainty, and the use of distributed computation.

In our development of several of the topics of this book we rely on methodology that is covered in greater depth in the 1996 neuro-dynamic programming book [BeT96] (jointly written with J. Tsitsiklis) as well as the author's recent RL book [Ber19a]. However, we aim to develop rollout and approximate policy iteration beyond the books [BeT96] and [Ber19a]. In particular, we present new research, relating to rollout variants, distributed asynchronous computation, partitioned architectures, and multiagent systems. We also indicate how our methods are well-suited for several types of challenging large scale optimization problems, such as combinatorial/discrete optimization, as well as partially observed Markov decision problems (POMDP).

This monograph took shape in the fall of 2019 and was largely based on two separate but related lines of research for distributed large-scale computation:

(a) My work on multiagent rollout, policy iteration, and value iteration, which was published in the papers [Ber19c], [Ber19d], [Ber20], [Ber21a]. It was based on my earlier work on rollout, which started in the mid 90s, in the context of the neuro-dynamic programming book, and continued for several years afterwards.

(b) My work on distributed policy iteration algorithms with state space-partitioned architectures. These ideas were extended, implemented, and applied to some large-scale POMDP problems in collaboration with my Arizona State University (ASU) colleagues Sushmita Bhattacharya, Sahil Badyal, Thomas Wheeler, and Stephanie Gil [BBW20]. This work is also connected with my joint research on asynchronous distributed state space-partitioned policy iteration with Huizhen Yu [BeY10], [BeY12], [YuB13], which is presented in Section 5.8 of this monograph.

Most of the book was written while teaching a research-oriented course at ASU, starting in January 2020. The hospitable and stimulating environment at ASU contributed much to my productivity during this period, and for this I am very thankful to several colleagues and students, including Stephanie Gil, Giulia Pedrielli, and Petr Sulc, and my teaching assistant, Sushmita Bhattacharya. I have also appreciated fruitful interactions with colleagues and students outside ASU, particularly Yuchao Li, who also provided valuable proofreading support.

Finally, I would like to dedicate this monograph to the creative genius of Claude Shannon, the father of information theory, but also the father of computer chess. His approximate dynamic programming ideas, which predated the work of Bellman, live on inside the AlphaZero program, the most impressive success story of reinforcement learning up to now.

Dimitri P. Bertsekas
July 2020

NOTE ABOUT THIS UPDATED PRINTING

This 2nd printing was prompted by the publication of the book in China, and by the use of the book for an on-line course at ASU in the Spring of 2021. See my website:

http://web.mit.edu/dimitrib/www/RLbook.html

Simultaneously with its publication in China, this 2nd printing will replace the 1st printing outside of China.

In addition to editorial corrections, I took the opportunity to make some more substantive revisions. One type of revision aimed to enrich the material on multiagent control systems, and to introduce enhancements in the form of signaling policies (see Section 5.3.4). Another type of revision aimed to highlight the connections of AlphaZero and related programs with approximations in value space and the on-line play idea that lies at the heart of rollout.

The significance of on-line policy improvement by multistep lookahead and rollout came into focus with the success of AlphaZero and the earlier, but just as impressive TD-Gammon program. Both of these programs involve an *off-line training* algorithm, and an *on-line play* algorithm that relies on the results of the off-line training. These two algorithms are different, but a key fact is that *the on-line player performs much better than the extensively trained off-line player*; see Section 1.1.

In practical problems of decision and control, a lot of analysis and/or off-line computation is often directed towards obtaining a policy, which is inevitably suboptimal, because of model imperfections, changing problem parameters, and overwhelming computational bottlenecks. The AlphaZero and TD-Gammon experience reinforces an important conclusion: despite the off-line effort that may have gone into the design of a policy, its performance may be greatly improved by on-line approximation in value space, with long lookahead (involving minimization or rollout with this policy), and terminal cost approximation.

This performance enhancement by on-line play goes well beyond the conventional control wisdom that "feedback corrects for noise, uncertainty, and modeling errors." It is embodied to some extent by the model predictive control methodology, but it also suggests a more broadly applicable paradigm, whose significance has yet to be fully recognized by the decision and control community.

In this revised printing I have also aimed to illustrate the ideas of on-line play, rollout, and policy iteration with intuitive figures that draw upon the use of abstract forms of the Bellman equation operators. In this way a direct line, couched on insightful visualization, can be drawn from AlphaZero to optimal, model predictive, and adaptive control.

To improve the didactic value of the book, I have also added some material on infinite horizon problems to Chapter 1. Furthermore, I have added exercises for the reader at the end of each chapter. Moreover, during the preceding year I supplemented the book with quite a few on-line extensions, including research papers, and lecture slides and videos. In particular, I posted a series of videolectures and slides from my 2021 course on reinforcement learning at ASU. This material is freely accessible from my website. The videolectures are also available at

https://www.youtube.com/playlist?list
=PLmH30BG15SIp79JRJ-MVF12uvB1qPtPzn

and at

https://space.bilibili.com/2036999141

Finally, I wish to thank students and colleagues for many helpful comments relating to the 1st printing and the ASU course. I am particularly thankful to Yuchao Li for proofreading support and numerous valuable suggestions.

Dimitri P. Bertsekas

July 2021