Quantitative Finance

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/rquf20

Reinforcement Learning and Stochastic Optimization: A Unified Framework for Sequential Decisions

by Warren B. Powell (ed.), Wiley (2022). Hardback. ISBN 9781119815051.

Igor Halperin

To cite this article: Igor Halperin (2022) Reinforcement Learning and Stochastic Optimization: A Unified Framework for Sequential Decisions, Quantitative Finance, 22:12, 2151-2154, DOI: 10.1080/14697688.2022.2135456

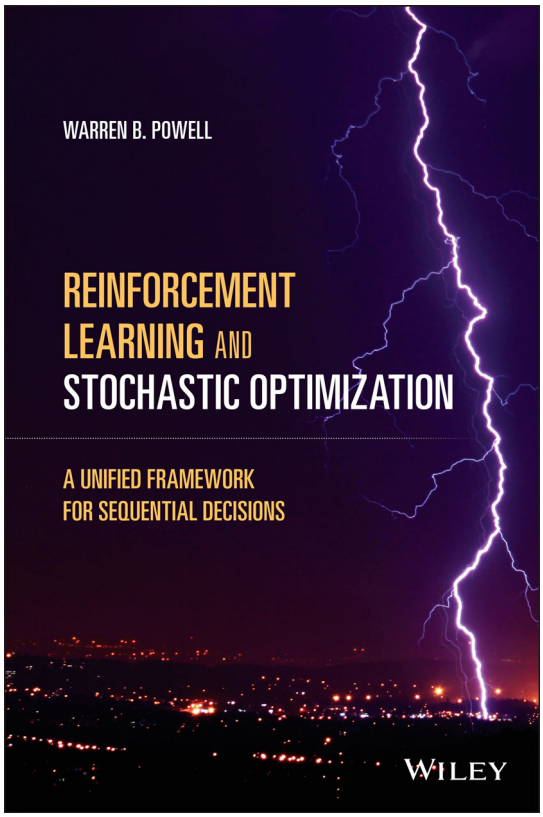
To link to this article: https://doi.org/10.1080/14697688.2022.2135456

Published online: 28 Nov 2022.

Quantitative Finance, 2022
Vol. 22, No. 12, 2151–2154, https://doi.org/10.1080/14697688.2022.2135456

Routledge
Taylor & Francis Group

Book review



Reinforcement Learning and Stochastic Optimization: A Unified Framework for Sequential Decisions, by Warren B. Powell (ed.), Wiley (2022). Hardback. ISBN 9781119815051.

What is reinforcement learning? How is reinforcement learning different from stochastic optimization? And finally, can it be used for applications to quantitative finance for my current or future projects? If two or more of these questions are in the scope of your interests, the new book 'Reinforcement learning and Stochastic Optimization: a Unified Framework for Sequential Decisions' by Warren Powell, Professor Emeritus at Princeton University, may become your good friend for years to come. This treatise, quite encyclopaedical in both the physical volume (1099 pages!) and the number of topics covered, summarizes more than 30 years of Prof. Powell's academic and industrial research across different industries, in the field that he himself prefers to call sequential decision problems.

But what is in the term 'sequential decision making', and why does it deserve such a prominent spot in the subtitle of the book? In my personal view, this term is far more self-explanatory and intuitive than the terms 'reinforcement learning' or 'stochastic optimization', to a novice in the field. Indeed, sequential decision problems are literally problems where an agent (a human or a robot) has to make decisions over some (finite or infinite) period of time. There are myriads of problems in the natural and social sciences, including in particular economics and finance, that amount to sequential decision-making. One can even say, and this is my own deep conviction, that most of quantitative finance amount to sequential decision problems, and this is exactly the reason for this book to potentially be of much interests for the readers of this journal. I will elaborate more on these points momentarily, but first let me go back to the title of the book, which is, quite remarkably, is *not* called 'Sequential Decision Problems'. Instead, it has the words 'reinforcement learning' and 'stochastic optimization' in the title. Why is that, and how are these terms related to sequential decision problems?

In Prof Powell's view, both terms, along with a few related ones, refer basically to different flavors, or different application domains developed by 15 different scientific communities! Naturally, this produces a tendency to develop a Babylon tower phenomenon, where different communities end up rediscovering essentially the same methods but under different names, and often with a narrow focus on a particular application or application area.

One reason that the title of the book contains the words 'reinforcement learning' rather than 'sequential decisions' is that the former is one of the most recognizable buzzwords in the communities of academics and practitioners alike, along with such terms as AI and deep learning. In author's words, the presence of words 'reinforcement learning' on the cover of the book is reflective of both a recognition and wide interest in reinforcement learning from communities of practitioners, academics and researchers.

Based on my own experience of interactions with people working in quantitative finance, I would assume that (i) more people have heard the term 'reinforcement learning' than 'sequential decision problems', and (ii) most people who have heard about reinforcement learning, heard it in the combination 'deep reinforcement learning', to the extent that the two are nearly identified in the mind of many people. In reality, the 'deep' part in the name simply points to the use of deep neural networks as a computational element performing function approximation for either a value function or a policy function.

I have had a few conversations in recent years with traders and quants who shared with me their experience with reinforcement learning. In many cases, it went like this: 'We tried reinforcement learning, and it did not work', or 'We tried it, and it worked sometimes, and sometimes it did not, and we were not able to find out what happens when it stops to work'.

To me, this sounded similar to 'We have tried calculus for our problem, but it did not work.' Reinforcement learning cannot 'not work', because it is a framework, and not a particular algorithm or an open-source library. At a closer inspection, I found that in all such cases, by 'trying reinforcement learning', people really meant: 'We tried one of the deep reinforcement learning libraries more or less verbatim out of the box, simply by switching to our own datasets, and playing with hyperparameters in a quest for convergence'.

Undoubtfully, the modern open source packages based on TensorFlow or PyTorch frameworks democratized access to deep reinforcement learning algorithms developed for video games and robotics by such companies as Google's DeepMind and OpenAI. An important remaining question before anyone interested in financial applications of reinforcement learning should feel ready to put their hands on the code is how much they need to know about reinforcement learning in the first place? Would be reading a Wikipedia article enough? Probably not. How about a short video course? This should probably give more confidence to a novice in the field. How about a 1099 pages long book (this book) that presents, among other things, a unifying view of all problems of sequential decision-making and produces a universal classification of all possible policies into 4 classes?

The latter fact that this book presents in-depth ontology of all possible decision-making problems and provides a universal modeling framework that unifies methods developed in 15 different communities may, in my view, have a different appeal to the readers, depending on their 'entry point'. First, one should mention the basic pre-requisites to the readers of the book, which are very common for most modern books on machine learning. Essentially, only the basic probability theory and basic linear algebra are needed. In addition, the reader is supposed to be able to implement the algorithms of this book in a software.

From this common starting point, the entry knowledge level of the reader may bifurcate and vary quite wildly. Readers with no prior knowledge of optimal control theory, stochastic optimization, reinforcement learning will enjoy a well-thought general approach to modeling sequential decisions under uncertainty. On the other hand, readers with some prior knowledge in one of these fields will find a bridge to it in chapter 2, which will undoubtfully help them to use it as an early reference point in navigating a unifying framework presented at length in this book.

In addition to these two sub-groups of readers, we can envision a yet third group of professionals that have already learned elements of reinforcement learning, e.g. from some video lectures or by reading some chapters from the classical book by Sutton and Barto, and already have a limited experience of applying available deep reinforcement learning libraries to real-world problems. What does this book have to offer to these readers? Would a conceptual framing of their prior knowledge into the general universal approach presented in this book be helpful in what they already do? Or would such an ontological foundation rather feel like a practically irrelevant nuisance? For example, referring to a financial example oriented towards the main audience of Quantitative Finance, if you already use deep hedging as a model-based policy optimization method to price and risk manage an option portfolio, how much of an added value you get from such ontological knowledge?

The famous poem 'The Centipede's Dilemma' tells the story of a centipede who lost its ability to run when it tried to figure out which one of its legs moved after which one. In psychology, this is known as hyper-reflection. Given that the intended audience of this book are newcomers in the field, as well as practitioners who are looking to solve real-world problems and implement their work in software, a related question we could ask ourselves is 'How much of the knowledge of this book I should consume to get a level of understanding of the field needed to guide me in trying it in practice?' In general, my answer to such questions would be 'the more the better'. Even if you are currently focused on a particular problem and believe you know the best tool, it may be beneficial to look at your problem within a unified framework perspective developed in this book, as well as get familiar with methods you were not familiar with prior to this book. The famed physicist Richard Feynman, one of the fathers of quantum electrodynamics and a Nobel prize winner, is known, among other things, for his continuous quest to 'view the world from another point of view'. In my view, this book offers a related experience, in the specific area of sequential decision-making, even to readers with prior knowledge and/or practical experience in the field.

This said, the recommendation to consume as much as possible of this book can be as general as nearly useless in practice, especially given that one of the target audience segments are practitioners who may not always be able to allocate a time budget sufficient for a back-to-back reading of this book. These readers can first read the four introductory chapters and then navigate to a chapter of a particular interest, say chapter 19 on direct lookahead policies. Thanks to a well-thought design and the presentation style of the book, I believe it can be used in a multitude of ways.

So, what will the reader find in the book? It is organized into six parts. The first one is an introduction and foundation for the rest of the book. It introduces canonical problems of sequential decision-making and then presents online learning and stochastic search problems. Elements of classical machine learning methods such as neural networks, support vector machines and kernel regression are introduced along the way in chapter 3. Part II deals with state-independent decision-making problems where the problem itself does not change over time (this is the case, e.g. for multiarmed bandit problems). For all such state-independent problems, decision-making reduces to pure learning. Part III treats the much richer class of state-dependent problems where the problem being solved depends on information or parameters that change over time. The objective function for such problem depends on a state variable whose evolution may either evolve exogenously to an agent (for example, weather) or instead, it can be partially driven by the agent's actions (for example, market impact of traders can be modeled along these lines). Part IV describes policies that fall in the 'policy search' class, which include PFAs (policy function approximations) and CFA (cost function approximation). The following (long and heavily loaded) part V presents policies based on lookahead approximations. Readers who are already familiar with the basics of dynamic programming and reinforcement learning will find the familiar algorithms such as value iteration, policy iteration and Q-learning. A less common textbook and monograph material is presented on the topic of DLAs (direct lookahead approximations. Examples of this class include model predictive control and Monte Carlo tree search (MCTS). This exposition is very valuable, given in particular the fact that many of the recent successes of reinforcement learning, such as DeepMind's AlphaZero and MuZero embed MCTS as a model component. The final part VI gives an overview of how the framework developed in the book can be extended to multiagent systems. In particular, it shows how two-agent problems can be cast as partially observable Markov decision processes. It further extends to cooperative multiagent problems with the need to model communication.

The book is written in a very digestible 'read short' style well suited for readers with engineering, physics or computer science backgrounds, which stays clear of the language of theorems and mathematical proofs. Sections that are more mathematically heavier than the rest of the text and which provide more technical insights are marked by a double asterisk and can be skipped at the first reading. Detailed reference lists are provided for each chapter, which lets the interested reader follow the original papers for further details. Each chapter is accompanied by a number of exercises that belong into various categories, including review questions, computational exercises, theory questions, problem-solving questions, and programming exercises with some Python code.

While providing a unified framework for sequential decision analytics, the book is abundant in real-world examples that immediately map the mathematical formalism onto practically important industrial applications such as inventory management under uncertainty, supply chain routing, COVID modeling, blood management systems, and so on. Some of these real-world examples are supplemented by Python modules (available via a separate web page), which enables experimenting with practical algorithms, checking their sensitivity to hyperparameters or uncertainty modeling assumptions, etc. With no doubt, the practically-oriented and theoretically solid approach of the book should provide strong foundations as well as further tips to practitioners working on their real-world problems.

Having talked about what is in the book, I would like to briefly overview topics that are *not* covered, focusing in particular on topics of potential interest for practitioners in quantitative finance. First, the book does not offer real-world examples from the quantitative finance field. The readers who are interested in, say, using reinforcement learning for portfolio construction and risk management, optimal trade execution, derivative pricing, or wealth management would not find any applications in these areas. These readers would need to resort to original research papers or other books that cover these topics. In particular, the latter topics are covered in our book, co-authored with Matthew Dixon and Paul Bilokon, entitled 'Machine Learning in Finance: from Theory to Practice' (Springer 2020). While our MLF book has an introductory chapter on reinforcement learning mostly centered around value-based approaches, Prof. Powell's book naturally gives a much more in-depth presentation of foundations and alternative numerical approaches. Readers with a focus on applications of reinforcement learning to quantitative finance can therefore be advised to use both books concurrently or in sequence.

Further to the absence of finance-specific use cases, the book also passes on some methods that are popular in the current research literature. In particular, the book does not explain entropy-based regularization approaches to value-based methods, such as (online or offline) MaxEnt reinforcement learning or G-learning, or related approaches to offline RL as inference (an introduction to these topics can be found in our MLF book). The book also stays clear of examples involving deep reinforcement learning—recall that a lion share of the current research literature deals with deep reinforcement learning. As the book states in sect. 3.10.4,

> As of this writing, it is not yet clear if deep learners will prove useful in stochastic optimization, partly because our data comes from iterations of an algorithm, and partly because the high dimensional capabilities of neural network raise the risk of overfitting in the context of stochastic optimization problems.

I personally find it hard to disagree at this point.

Finally, one more topic not covered in this book but potentially of interest for practitioners in quantitative finance is inverse reinforcement learning (IRL for short). IRL can be viewed as an inference problem which is essentially a twist

of the (direct) problem of reinforcement learning (or sequential decision-making). While in RL, one usually starts with a well-specified reward function for an agent to maximize, in the IRL setting, the observer has access to a historical (i.e. collected offline) dataset made of sequences of states and actions performed by an agent (or agents) over some period of time. The task of the observer in IRL is to find out a reward function that the agents were trying to maximize, or in other words, to rationalize the observed behavior of the agents. Arguably, the number of real-world examples where the available data has such a format is of a comparable scale or may even exceed the number of problems where the choice of the reward function is available as a part of the training data (see, e.g. our MLF book for an introduction to IRL and potential financial applications). Furthermore, IRL and direct RL can work jointly in the setting of offline policy optimization, where the reward function of the agent(s) is first learned in the IRL steps, and then the policy is optimized in the direct RL steps following one of the methods presented in this book.

Still, mentioning these topics that are missing in the book does not take anything away from this remarkable and encyclopaedical work, which should be viewed as a highly desirable addition to the working library of researchers and practitioners in the field alike. You will very likely find yourself coming back to this book many times after the first read.

Igor Halperin
Financial Engineering, Fidelity, Boston, MA, USA

Igor Halperin is an AI researcher and the Group Data Science leader at Fidelity Investments. His research focuses on using methods of reinforcement learning, information theory, and physics for financial problems such as portfolio optimization, dynamic risk management, and inference of sequential decision-making processes of financial agents. Igor has an extensive industrial and academic experience in statistical and financial modeling, in particular in the areas of option pricing, credit portfolio risk modeling, and portfolio optimization. Prior to joining Fidelity, Igor worked as a Research Professor of Financial Machine Learning at NYU Tandon School of Engineering. Before that, Igor was an Executive Director of Quantitative Research at JPMorgan, and a quantitative researcher at Bloomberg LP. Igor has published numerous articles in finance and physics journals, and is a frequent speaker at financial conferences. He has co-authored the books "Machine Learning in Finance: From Theory to Practice" (Springer, 2020) and "Credit Risk Frontiers" (Bloomberg LP, 2012). Igor has a Ph.D. in theoretical high energy physics from Tel Aviv University, and a M.Sc. in nuclear physics from St. Petersburg State Technical University. In February 2022, Igor was named the Buy-Side Quant of the Year by RISK magazine.