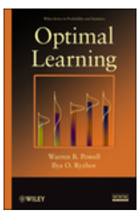
four constituent papers. Along with including the aforementioned additions, the organization of the book emphasizes a few important features of these problems and their solutions (mainly the need for non-anticipative scheduling policies and the role of early start schedules) early and often, drawing attention to the contrast and similarities between different problem types based on these features. To its great advantage, the book flows very well, with problems carefully explained and models and assumptions discussed and, for the most part, carefully justified.

References

[1] W. Wiesemann, R. Hochreiter, and D. Kuhn. A stochastic programming approach for qos-aware service composition. In T. Priol, editor, *2008 8th IEEE International Symposium on Cluster Computing and the Grid (CCGrid 2008)*, pages 226–233. IEEE, 2008.
[2] W. Wiesemann, D. Kuhn, and B. Rustem. Maximizing the net present value of a project under uncertainty. *European Journal of Operational Research*, 202(2):356–367, 2010.
[3] W. Wiesemann, D. Kuhn, and B. Rustem. Multi-resource allocation in stochastic project scheduling. *Annals of Operations Research*, 193:193–220, 2012.
[4] W. Wiesemann, D. Kuhn, and B. Rustem. Robust resource allocations in temporal networks. *Mathematical Programming, Series A*, pages 1–35, To appear. Available online at http://dx.doi.org/10.1007/s10107-011-0478-7.



Optimal Learning, by Warren B. Powell and Ilya O. Ryzhov, Wiley, 2012.
See http://www.wiley.com. ▷

Reviewed by: Stephen E. Chick, INSEAD, Technology and Operations Management Area, stephen.chick@insead.edu.

The fields of operations research, computer science and statistics have a rich history of exploring optimal decision making. In some cases, complete information is available to support decisions and the resulting problem is deterministic in nature. In other cases, there is uncertainty involved, but we may know completely the probability distributions of the uncertainties involved, e.g., the selection of the optimal quantity of inventory to procure when the distribution of demand is known. Such problems involve optimization with stochastic outcomes. The book *Optimal learning* is focused on a third interesting class of problems, when the distributions that describe the uncertainty in the decision problem are themselves uncertain, and when auxiliary action can be taken sequentially to learn something about those distributions in order to improve future decisions.

Optimal learning builds a case for the importance of this third class of problems, which may range from very simple to very complex. For example, the demand for a product might be known to have a Poisson distribution but the mean demand might be unknown—what if information can be obtained to obtain a better estimate of that mean before inventory commitments are made. Or suppose that several alternative service system designs might be under consideration, that each design may be complex and stochastic, and that the alternative with the highest profit (in expectation) is to be implemented. If simulations can provide information about the performance of each alternative, at a cost, in what sequence should alternatives be sampled and for how long before an alternative is selected for implementation? In a third example, what if several chemical compounds are being considered as a drug therapy, the results of the treatment are stochastic, and a sequential trial is sought whereby patients are both treated (with associated costs and benefits) and the distribution of values of each treatment is inferred with each patient tested? Each of these examples presents an interesting structure: the probability distributions that describe uncertainty in a decision problem are unknown, but there may be an opportunity to collect information, with a potential cost, to reduce uncertainty about those distributions. That uncertainty reduction, in turn, can inform better future decisions.

Optimal learning provides background, theory, algorithms, and modeling ideas to address the interesting and general question of how to balance the cost of learning with the benefit of the information it brings. There are several reasons why this book is of particular interest for students, practitioners and researchers in the operations research, computer science, and applied probability communities. One reason is that the book is structured so as to be very useful in an eponymous senior-level project-oriented course on the application of operations research. There are a wide variety of applications that are described in the book that motivate the importance of optimal learning problems and that show how one might go about modeling those problems. There is also a web site that has a number of examples and usable code for implementing the concepts (key algorithms and examples are provided that make of java called from a spreadsheet interface, as well as some matlab code). There are overviews of theory development and citations to relevant literature that go more deeply into the theory.

Because of the broad generality of optimal learning problems, there has been quite a bit of research done on them. One strength of this book is that it tries to connect the applications in these several fields, and how the frameworks from those fields compare. For example, the first chapter describes a battery of applications where information acquisition and learning problems can be found: transportation, energy and the environment, science and engineering, health and medicine, sports, business, and other domains. They provide context for how such problems have been approached in different ways by several different communities: including streams in simulation optimization, ranking and selection, sequential sampling, bandit problems, global optimization of expensive functions in engineering, economics, design of experiments, and approximate dynamic programming. The book illustrates how different approaches from different fields compare both conceptually and in numerical experiments. For example, Chapter 4 compares algorithms that use pure exploration, pure exploitation, Botlzmann exploration, epsilon-greedy exploration, interval estimation, and Chernoff interval estimation and shows how they differ from the knowledge gradient approach. The book provides pointers to papers from different fields that may be helpful for cross-fertilization of ideas across these domains.

The book is structured in three main parts, the first two of which provide the framework of a senior-level course. The first part of the book, aptly entitled "Fundamentals," provides the motivation, intuition, and basic framework for optimal learning. After the "what" and "why" of optimal learning is presented in the first chapter, Chapter 2 describes what is meant by adaptive learning (the decision of what to observe next can depend on prior observations) and describes a Bayesian and expected value of information framework for doing so. Chapter 3 motivates and illustrates the notion of economics of information that is central to the book: that sampling can be used to improve decisions, as measured by potential future economic benefit, but may come at a cost. Chapter 4 applies these ideas to the area of statistical ranking and selection: the sequential sampling from several alternatives until a single alternative is selected as best, where best is determined by the expected reward of the alternative. Chapter 5 formalizes the knowledge gradient approach, a key workhorse in the book. As the name might imply, the knowledge gradient is an approximation of the expected value of information per additional sample to be observed. The knowledge gradient idea is applied in several contexts, and its relation to closely related concepts studied elsewhere, such as the expected improvement idea of Bayesian global optimization, and the expected linear loss approach to Bayesian ranking and selection. Chapter 6 describes another application of optimal learning—the multi-armed bandit problem. Chapter 7 provides a framework for modeling and classifying optimal learning problems that is motivated by the applications and theory in the preceding chapters.

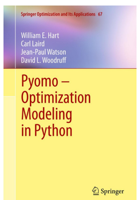
Part two, entitled "Extensions and Applications," demonstrates how complexities that may arise in actual applications might be approached by tweaking the tools that were introduced in part one. In their course, the authors have covering the material in the second part of the book in parallel to having the students work on projects that apply the concepts from the first part. Chapter 8 describes a neat use of linear models to support learning by sampling one alternative to make inferences about related alternatives. Examples from dynamic pricing in eCommerce, drug therapy response optimization in medicine, housing loan decision, and other areas are included. Chapter 9 adapts the learning framework to the context of subset selection (rather than selection of an individual alternative), with reference to applications in energy portfolio selection, assortment planning, and shortest path selection. Chapter 10 explores the optimization of a scalar function when samples are observed with noise. Chapter 11 applies the sequential learning ideas to optimal bidding strategies and pricing decisions. Chapter 12 makes the link to optimal stopping problems such as the secretary problem. Taken as a whole, the five chapters in part 2 of the book demonstrate the flexibility of the knowledge gradient approach to be adapted to a variety of contexts, even when samples from one alternative can give indirect information about the mean performance of other alternatives. This part provides a nice basis for discussion about the modeling process itself, not just the mathematical adaptation of the knowledge gradient tool for optimal learning problems.

Part three of the book has five chapters that focus on more advanced topics: active learning in statistics; simulation optimization; learning in mathematical programming; optimizing over continuous measurements; learning with a physical state. These chapters are more appropriate as a reference for researchers in the field, and are less appropriate for undergraduate or first year masters students. Part three makes interesting connections between the proposed learning framework, linear programming when some parameters are unknown but might be inferred through observation, approximate dynamic programming, learning in network flow problems, and other areas. This third part provides references to related books and papers in other fields, but will be easier to approach if concepts like entropy and experimental design are already somewhat familiar to the reader.

The book uses a number of simple examples to illustrate points, such as with decision trees and the well-studied case of samples with independent normal distributions and conjugate prior distributions. These very approachable examples are nicely complemented with additional meat that shows how the approach can apply in other contexts, such as with correlated beliefs about the mean performance of alternatives and other sampling distributions such as Bernoulli, exponential, and uniformly distributed samples. Derivations of some more advanced results are relegated to optional subsections, which makes the book amenable to a broader range of levels in the classroom—one can imagine a master's level course using the text, and the text is great self-reading for PhD students that are wishing to develop a good overview of the optimal learning field.

The title of the book is *Optimal Learning*, yet the techniques in the book do not always provide optimal solutions. The knowledge gradient, a central concept in the book, is an approximation to optimal learning. That said, its versatility as a modeling construct and the strong performance of variations on this concept shows that the knowledge gradient is a powerful tool that can be used as a building block in many applications.

The authors and others that have contributed to the development of the knowledge gradient have made some exciting progress in the area of sequential sampling and learning, and this book is an enjoyable, useful, and approachable summary of both basics and some interesting advance topics in this field. This book collects a number of interesting ideas in optimal learning, allows for connections to be made across disciplines, and is a welcome addition to my bookshelf.



Pyomo – Optimization Modeling in Python, by W. E. Hart, C. Laird, J. P. Watson, and D. L. Woodruff, Springer, 2012. See http://www.springer.com. ▷

Reviewed by: Nedialko B. Dimitrov, Operations Research Department, Naval Postgraduate School, ned@nps.edu.

If a simple, intuitive tool for a task exists, the task is done more often, by more people. This basic principle is as true for gardening and gadgets, as it is for computation in operations research. The book, *Pyomo – Optimization Modeling in Python*, documents a simple, yet versatile tool for modeling and solving optimization problems.

Pyomo, which stands for Python Optimization Modeling Objects, is an *algebraic modeling language* (AML) developed by a diverse set of researchers and hosted at Sandia National Laboratories. An AML allows a user to specify the algebraic form of an optimization problem, independent of its solution algorithm. Further, separating algebraic problem specification and data allows a user to solve varied instances of the same problem with relatively little overhead.

To solve an optimization problem, a researcher has two basic options: to use a high level modeling language that is separate from, but works in conjunction with, a general purpose solver, or to interface with a solver, possibly custom-made, directly through solver-specific libraries. The advantages of a modeling language are ease of problem specification, and the ability to switch solvers with no overhead. The advantage of a direct interface with a solver is the configurability of the optimization algorithm. Pyomo stands apart because it combines these two advantages.

The core difference between Pyomo and other AMLs is its development. Typically, languages like AMPL and GAMS are created with special-purpose syntax for defining optimization problems. This leads to relative ease for defining standard optimization problems, but invariably, for complex problems, a user requires complex logic for data pre-processing, data post-processing, and even solving. Existing AMLs add some standard programming language constructs to allow for these tasks, but using them is cumbersome to a user. Pyomo, on the other hand, starts with a popular, general purpose, modern programming language—Python—and adds AML constructs. In doing so, the complex pre-processing, post-processing, and solving logic can be easily implemented by a user while still maintaining the mathematical specification and solver independence of an AML. The idea of building an AML on top of a standard programming language is appealing and other projects also follow that model: CVX, CVXPY, PuLP-OR.

The book, by Bill Hart, Carl Laird, Jean-Paul Watson, and David Woodruff, is essential to the usability of Pyomo, serving as *the* Pyomo documentation. The main alternative to learning the power and flexibility of Pyomo is to read example code, or dive into the open-source software's internals. Because Pyomo is only a few years old and under active development, it does not yet