

## Book reviews

---

**Handbook of Learning and Approximate Dynamic Programming**, Jennie Si, Andrew G. Barto, Warren B. Powell and Donald Wunsch II, (eds), IEEE Press and J. Wiley, Hoboken, New Jersey, 2004, ISBN 0 471 66054 x

As computers become more and more powerful, as algorithms improve, more and more computing power is behind the engineering systems' behavior. As a result of this increasing sophisticated data processing and decision making, many current systems exhibit more and more intelligent behavior. However, no matter how intelligent they may seem, in comparison with the previous less sophisticated systems, true intelligence also means learning, and many existing systems, from many sophisticated robots to automatic control systems for power grid, do not have the ability to learn.

One reason for this is the fact that learning is difficult. For simple static systems, when the output  $y$  depends on the few inputs  $x_1, \dots, x_n$ , we can learn the appropriate dependence  $y = f(x_1, \dots, x_n)$ , e.g., by using artificial neural networks that use backpropagation or a similar rule to learn. For many real life systems, the situation is not so easy. For example, for controlling a power grid, we must make decisions  $y(t)$  based not only on the current values  $x_i(t)$  of power production and power demands, but also on the past values  $x_i(t')$  ( $t' < t$ ) of the corresponding parameters. The objective is to maximize the discounted utility  $\sum q^t \cdot u(t)$ , where  $u(t)$  is the overall utility at time  $t$  and  $q < 1$  is a discounting coefficient. To describe a general control of such a system, we must describe the dependence of the current control parameter  $y(t)$  on many different variables  $x_i(t)$ ,  $x_i(t-1)$ , etc. With a large number of inputs, not only the neural network training becomes difficult, but, as Paul Werbos, the author of backpropagation, has shown, in such case, often, backpropagation leads to an irrelevant local minimum instead of the correct values of the weights. How can we make training feasible for such dynamic systems?

One problem is that we have both uncertainty in how the system would react to different controls – the uncertainty that requires neural network training to find

this out – and the complex dynamic character of the system. We have already mentioned that when the system is not complex, the problem becomes efficiently solvable – we can use traditional neural networks to train it. In the other simplified case, when the system is dynamic but its behavior is known, the problem also becomes efficiently solvable: namely, we can use dynamic programming to solve the corresponding optimization problem.

It is therefore desirable to try to combine the methods for solving these two simplified cases, i.e., neural networks and dynamic programming, to handle the optimization of a dynamic system in the situation when the behavior of this system is not known exactly and need to be learned.

This combination is not as wishful as it may sound because the backpropagation methods for training neural networks and dynamic programming methods for optimizing nonlinear systems actually have an idea in common. Indeed, the main idea behind dynamic programming is that to optimize the decision making, at a given moment to time  $t$ , we select a control  $u(t)$  that optimizes a special auxiliary function  $J(t, x(t))$ . To be more precise, we first go “backwards” in time and compute the value of the function  $J(t, x)$  for  $t = T, T-1, \dots$ ; after this, we go forward in time and compute the controls  $u(0), u(1), \dots$ . A similar alternation of forward and backward steps is what makes backpropagation computationally efficient: first, we go forward, from the input to the output, and use the current weights to compute the output; then, we compare the results with the desired results and use these errors to propagate back to the inputs and update the weights along the way.

This handbook appeared as a result of the workshop organized in 2002 by the National Science Foundation to enhance the collaboration between dynamic programming and computer learning communities. This was the first serious joint meeting of the two research communities that really boosted the collaboration. The papers resulting from this collaboration form this handbook.

This book already has a lot of interesting results and even a lot of practical applications ranging from helicopter flight control to power system control. In my opinion, however, the most interesting part of this book is that it is a snapshot on a new interesting area of research collaboration, with numerous possibilities. Some readers will read this book and find algorithms for solving their problems, but even more readers will

hopefully find ideas that will help them in their problems as well.

Vladik Kreinovich  
Book Review Editor  
*Journal of Intelligent & Fuzzy Systems*

**Static and Dynamic Neural Networks**, Madan M. Gupta, Liang Jin and Noriyasu Homma, (eds), J. Wiley, Hoboken, New Jersey, 2003, ISBN 0 471 219 48 7

This book is a well-written textbook, with numerous helpful exercises and an accompanying ftp site. It starts with a detailed description of biological neurons, and how different features of biological neurons inspired different types of artificial neural networks (Chapters 2 and 3). The book then describes traditional neuron models, from linear to nonlinear, explains the need for multiple layers and the main ideas behind backpropagation (Chapter 4). After that, Chapter 5 overviews possible modifications of the standard backpropagation algorithm, such as:

- robust methods that minimize  $l^p$ -norm  $\sum |e_i|^p$  instead of the traditional means squares  $\sum |e_i|^2$ ,
- regularization techniques – that, crudely speaking, make learning smoother,
- network pruning techniques that simplify the network by deleting connections with low weights that do not contribute to the results, and
- second order versions of backpropagation that sometimes speed up the learning process.

Chapter 6 describes radial basis function neural net-

works, and Chapter 7 overviews different universal approximation results for neural networks.

Part III describes dynamic neural networks, in which, similar to how our brains handle difficult problems, we do not simply follow a straightforward step from input to output; instead, neurons communicate with each other and change each other's states, probably for a reasonable time, until the system finally stabilizes. The resulting stable state is the solution to the original problem. An important aspect of such neural networks is their stability that is analyzed in Chapters 11 and 12. Part IV includes advanced topics, including the practically useful topic of fuzzy neural networks that is rarely covered in introductory neural textbooks.

Overall, the exposition is very clear, the exercises help a lot. This book can be used not only as a textbook, but also as a reference book both for practitioners who want to use neural networks (and go beyond clicking the buttons in Matlab neural package) and for researchers who are interested in different aspects of neural data processing.

Vladik Kreinovich  
Book Review Editor  
*Journal of Intelligent & Fuzzy Systems*

**Virtual Reality Technology**, Grigore C. Burdea and Philippe Coiffet, 2nd edition, Wiley, Hoboken, New Jersey, 2003, ISBN 0 471 36089 9

Virtual Reality is such a fast growing field that in effect, this second edition is really a new book. The necessity for such a book is even more clear if we take into account that in the early 1990s, when the first edition appeared, the field of Virtual Reality was over-hyped by the media – which, predictably, led to a public disappointment in VR. It is time for an honest re-evaluation of this maturing field.

This book is a textbook, largely non-technical, that explains the basics of all aspects of virtual reality (VR) and lists its current and potential applications. Chapter 1 is devoted to VR's history – it is older than most people think, predating computers by a few decades – and an overview of the field.

Chapters 2–4 describe VR hardware. Chapter 2 describes different input devices: trackers that track where the user looks, gloves that track what the user touches and how, etc. Chapter 3 describes different output devices: graphic displays, sound generators, and haptic devices that provide the tactile feedback. Chapter 4 describes architectures of different computing systems.

Chapters 5 and 6 describe VR software. Chapter 5 describes the models that are used in VR computing,

including:

- geometric models that describe the location of the simulated objects and
- force models and dynamic models that describe the interaction between these objects.

Chapter 6 overviews different software tools available for VR programming.

Chapter 7 describes human aspects of VR such as:

- health and safety issues – including the much-discussed cybersickness, and
- possible societal consequences of VR applications.

Finally, Chapters 8 and 9 describe applications. Chapter 8 overviews current applications, to medical training, military training, and entertainment. Chapter 9 describes emerging applications in manufacturing, robotics, and data visualization.

The book is supplied with a CD that has VR examples in VRML and Java 3D. The book is non-technical but references to technical aspects are provided for interested readers.

Vladik Kreinovich  
Book Review Editor  
*Journal of Intelligent & Fuzzy Systems*

**Fuzzy Relational Systems: Foundations and Principles**, Radim Bělohlávek, Kluwer Academic/Plenum Publishers, New York, 2002, ISBN 0 306 467777 1

Traditional introductory books on fuzzy logic and fuzzy set theory usually start with the most widely used set of truth values – the interval  $[0,1]$ . This approach is reasonable for books designed for practitioners, because it enables them to start using fuzzy tools without the need to first go beyond simple math.

On the other hand, for readers who are more interested in foundations of fuzzy techniques, for readers who are not yet convinced that these tools are appropriate for solving their problems, it is definitely desirable to first explain the motivations and general foundations, so that the formulas and techniques are justified and not simply given in a usual ad hoc manner. For such readers, Radim Bělohlávek has provided a very well written book.

The book starts with motivations behind residuated lattice, MV-algebras, and similar structures. These terms should be familiar to the readers who have ever glanced through papers on mathematics of fuzzy sets, but the corresponding algebraic structures are rarely motivated in these papers. This book provides a convincing motivation for the properties that define these structures, and explains how these structures (and the mathematical results about these structures) are related to the main motivation behind fuzzy logic: description and formalization of natural language-related human reasoning.

After the detailed introduction, the author describes his contribution to the field. Again, his motivations are very convincing. For example, in the traditional approach to fuzzy, a fuzzy subset  $A$  of a universal set  $X$  – describing a fuzzy property of elements of  $X$  – is described as an arbitrary function from  $X$  to the set  $L$  of truth values (usually, the set  $[0,1]$ ). Not all such functions, however, make sense from the viewpoint of the real-life human “fuzzy” reasoning, because for real “fuzzy” words  $A$  like “big”, there is an additional important property: e.g., if  $x$  is close to  $y$ , and  $x$  is big, then  $y$  is also rather big. In other words, in practice, we do not simply have a universal set on which all kinds of properties can be defined. On the set  $X$ , we also have a natural relation “close”  $\sim$ , and only those properties  $A$  make practical sense for which  $A(x)$  and  $x \sim y$  imply  $A(y)$  (in some reasonable sense).

Similarly, not all functions make practical sense, only functions for which, if  $x$  is close to  $y$ , then  $f(x)$  is close to  $f(y)$ . For functions of several variables, if  $x_1 \sim y_1, \dots$ , and  $x_n \sim y_n$ , then the result  $f(x_1, \dots, x_n)$  of applying this function  $f$  to the original values  $x_1, \dots, x_n$  should be close to the result  $f(y_1, \dots, y_n)$  of applying the same function  $f$  to close values  $y_1, \dots, y_n$ . Similarly, not all relations are possible, etc.

Even operations of fuzzy logic, i.e., operations that enables us to estimate the degree of belief in, say,  $A \& B$  based on our degrees of belief in  $A$  and  $B$ , must be consistent with some natural notion of closeness.

This general restriction enables the author to avoid pathological (practically useless) examples of general fuzzy structures, and to provide reasonable motivations for the selections that are usually made on a very ad hoc basis. In other words, the transition from fuzzy set to fuzzy relational systems (on which relations like  $\sim$  are pre-defined) make foundations more natural.

On the other hand, the mathematical analysis of the corresponding systems becomes more complicated. It is no longer possible to simply define, e.g., a composition of the two functions: we must make sure that the resulting composition agrees (in the above sense) with the underlying closeness structures. Similarly, when we try to decompose a fuzzy relation into a composition (or represent a fuzzy function as a combination of several simpler fuzzy functions), it is no longer sufficient to provide the mathematical decomposition, we must also make sure that the component relations (correspondingly, component functions) are also consistent with the closeness structures. All these questions lead to a rich and mathematically sophisticated theory that forms the bulk of the book.

In general, this book is aimed at foundationally-minded readers, but readers who are interested in applications will definitely also benefit from reading this book: they will learn not only the fuzzy formulas, they will also learn where these formulas come from, and this knowledge will hopefully help in applications where traditional fuzzy techniques are not sufficient.

Vladik Kreinovich  
Book Review Editor  
*Journal of Intelligent & Fuzzy Systems*