space search and game playing, and then introduces Lisp and Prolog and discusses how to implement search mechanisms using them. Having covered some basic principles the book then addresses other methods of knowledge representation, rule-based expert systems, natural language understanding, automated reasoning (including resolution and the relation between state space search and resolution techniques), and a number of methods of machine learning. Then, as before, some of the theory is turned into programs with the implementations including some Prolog meta-interpreters and a natural language parser, along with a Lisp expert system shell and a version of the ID3 learning algorithm. Object-oriented programming is also discussed.

The stress that is placed on implementation is clearly a very good thing if one is interested in teaching artificial intelligence in a "hands on" fashion. With the code that is available, it would be possible to choose a path through the book such that every concept covered was illustrated by a piece of code that students could play with and yet sufficient material was covered that they ended up with a good grasp of the subject. Given that the publishers will supply the code on request, along with a public domain PROLOG interpreter, using the book as a basis for an introductory course on artificial intelligence is an attractive idea.

The introduction to the volumes *Knowledge engineering* edited by Hojjat Adeli claim that they "are suitable as primary reference books in introductory courses on applied AI and knowledge engineering." In my opinion this is simply not true. What Adeli has done is to invite various luminaries in various sub-fields of artificial intelligence to write a chapter on a particular topic. These chapters are then gathered together in no particular order, and with no linking material, so that they are much closer to being the usual collection of papers than a book that every student should have. Of course, there is nothing wrong with collecting together a set of papers, and many of the papers are very fine, but they do not constitute any sort of coherent whole and they certainly don't make a textbook. Of the twenty papers that make up the two volumes I believe only a quarter could be considered introductory, and these include some of the worst—a couple giving the impression that they were thrown together in the course of an afternoon. However, having dealt with these the rest are considerably better. A number of the papers give authoritative surveys of particular areas that are of sufficient standard and depth to be of interest to postgraduate students just starting out in the field as well as being engaging for people "doing" work in other areas of artificial intelligence. However, the best papers are the kind that one would find in any collection of papers, namely detailed descriptions of the author's own work, and this fact in my opinion finishes off any claims that *Knowledge engineering* might have to being a textbook. However interesting descriptions of a particular piece of work are to a given writer and those interested in their work, they are hardly relevant to or useful for students who have just begun to study the subject. However, these same papers may well prove interesting to those with more experience of the field, and it is to them that the books are most likely to appeal.

References

Nilsson, NJ, 1991. "Logic and artificial intelligence". *Artificial Intelligence* **47** 31–56.Birnbaum, L, 1991. "Rigor mortis: a response to Nilsson's 'Logic and artificial intelligence'. *Artificial Intelligence* **47** 57–78.

Reviewed by Simon Parsons, Advanced Computation Laboratory, Imperial Cancer Research Fund, London, UK

Learning in embedded systems by Leslie Pack Kaelbling, Bradford Books. MIT Press, USA, 1993, pp 176, $29.95, ISBN 0-262-11174-8.

An *embedded system* is one which "has an ongoing interaction with a dynamic external world" (p. 1—all references are to *Learning in Embedded Systems*). It has perceptual inputs and produces effector outputs, and can thus partly influence its environment. It may have internal states which will allow its *action mapping*, the mapping from inputs to outputs, to depend on any, all, or none of

the input history. *Learning in Embedded Systems* explores the learning of such action maps, and focuses on techniques for learning them "without building state-transition models as an intermediate stage" (p. 5).

Kaelbling develops a number of new algorithms, exhibits their historical and theoretical connections to existing algorithms, reports on her extensive empirical testing of the various algorithms, and examines in detail their complexity in space and time. The balance between these disparate components is excellent; indeed, the book could be read profitably for this overall balance alone. The book develops nicely as various features of the algorithms are added successively. Kaelbling shows an excellent knowledge of the literature; the references to past and present works are comprehensive and scholarly.

Kaelbling focuses sharply on reinforcement learning in which each state of the environment is accompanied by a measure of how desirable the state is for the system, its reinforcement value. The system's goal is to maximize the overall reinforcement value over time. A major issue resulting from this approach, and one motivating much of Kaelbling's work, is the tension in such systems, when in a state of imperfect knowledge, between choosing to perform an action to gain knowledge about its reinforcement value, and performing an action already known to have a high reinforcement value in order to maximize the overall reinforcement.

The book is relevant to researchers in artificial intelligence and robotics in general, but will be of most interest to those working directly in machine learning. For such readers, it is more or less self-contained. It is suitable for students only at the advanced level. The book is very well written; it is somewhat terse, but not overly so. The transition between the most elementary and the advanced sections are very good, though some readers might find a little more background helpful at some points. The index is surprisingly inadequate.

Having formulated the reinforcement learning problem, giving the foundational concepts, and surveyed some existing algorithms for simple, two-action reinforcement learning, Kaelbling presents a new approach: the interval estimation algorithm. It uses second-order information to estimate the reinforcement value, as well as the confidence level of that estimate, of an action not yet performed. Kaelbling uses these estimates to balance the conflicting aims of acting to gain information and acting to gain reinforcement. The method is compared empirically to six known algorithms. Kaelbling shows that this method of learning action maps for two-action systems can be generalized to multi-action systems by cascading instances of the two-action learner.

Kaelbling presents two algorithms for learning functions in k-DNF (DNF with at most k conjunctive terms) from reinforcement. LARCKDNF is partly based on known techniques of linear associative reinforcement comparison, whereas IEKDNF is based on her interval estimation method. Both perform better than previous methods in both time and the range of functions that are thus learnable.

The generate-and-test reinforcement learning Algorithm (GTRL), which is more flexible than those above, represents the high point of Kaelbling's current work. It "performs a restricted, real-time, generate-and-test search in a syntactic space of Boolean-function descriptions" (p. 12). The algorithm is highly parameterized, and can be tuned in a wide variety of ways relating to the structure of the search space and to how it is to be searched. The search method is essentially a form of beam search. In particular, it can be tuned to trade off the generality of the problem against the algorithmic efficiency. It is superior to all of the algorithms previously considered and employs some symbolic as well as statistical methods. The space and time complexities are shown to be linear in L, the "length of the longest formula that can be constructed by the algorithm" (p. 101). This is a noteworthy result. Kaelbling does not address the question of whether, as the problem instances increase in size, the algorithmic performance it can be expected to suffer unless larger and larger L values are employed.

Kaelbling considers an extension to GTRL intended to address the problem of *perceptual aliasing*, which results from the fact that a system's perceptual limitations may cause it to be unable to discern objectively different states of the world. She deems her attempt as having only limited value, and the problem as needing further work. She also considers methods of optimizing the

reinforcement over a series of future actions, as opposed to merely over the next action. Such systems can learn actions for which the reward is delayed. Existing techniques are reviewed and extended using the interval estimation method and the algorithms are compared experimentally. Again, Kaelbling deems the interval estimation method as having only limited value and the problem as needing further work. Finally, some of the algorithms developed are applied to an actual mobile robot with mixed but overall very encouraging results.

This is an important and impressive work. It is scholarly, richly detailed, and presents systematically a number of new algorithms in reinforcement learning. it should be studied by anyone with an interest in learning; it will influence the development of reinforcement learning methods for some time.

Reviewed by Richard Wyatt, Department of Mathematics and Computer Science, West Chester University, USA

Information systems development: A database approach (2nd ed) by D. E. Avison, Blackwell Scientific, 1992, £16.99, pp 329, ISBN 0-632-03028-3.

Information systems, according to the foreword in this book, is a larger view of information technology, including the technology but considering its development, use and social or business consequences. Blackwell Scientific provides a series on Information Systems for this obviously interdisciplinary area, and this particular text is intended for students of computer science and business. It provides a readable, readily comprehensible overview of the subject, from a user's perspective. The reader could easily come from a business background or be a technically minded manager involved with information systems, or a student in applications of computer science.

The reader is given an easy and broad introduction to database technology and system implementation, using what is termed a "data-oriented methodology". The book's content is based around database concepts but focuses on this particular methodology and its implementation. At a general level it provides the essentials of entity-relation or conceptual modelling, explaining the conceptual schema, data analysis and the relational model. A sketch is provided, for the intended wide readership, of relational algebra and calculus and logical schema. Similarly, the basics of relational, hierarchical, network and object-oriented databases are explained clearly and comparatively.

This book is well structured with extensive internal cross-referencing, and it makes good use of bold type and emphases. It is well illustrated with figures, and features simple but often real examples. For the more interested reader there are references for further reading at the end of each chapter.

The author is realistic in describing the book's content. He is not over ambitious in its claims to depth but admits readily to simplification and brevity. The explanations are very clear and the subject matter is treated to a systematic overview. Avison explains and describes the many database terms and concepts clearly and well, but cannot avoid including the acronyms that plague this discipline.

Notions such as the software engineering lifecycle are described in applications development and database implementation. Points of theory and practice in information systems are made concrete with examples, criticisms and implementation details. The information systems requirement for databases is described in so far as it is met by current and old technology, which is appropriate for the applied approach of the book. Similarly, the shortness of the section on object-oriented (OO) databases reflects the publication date (1992), and is an honest comment on the lack of widespread usage of OO in business.

Since this is not a solely technical book it mixes discussion of business objectives and management structures with design issues of modelling and hardware aspects of file access. This