The Knowledge Engineering Review, Vol. 10:1, 1995, 101–114

Book reviews

Fuzzy logic for the management of uncertainty edited by Lotfi Zadeh and Janusz Kacprzyk, John Wiley & Sons, New York, 1992, pp 676, £47.50, ISBN 0-471-54799-9.

Fuzzy sets and fuzzy logic provide techniques for modelling vagueness and imprecision, and approximate reasoning, respectively. Zadeh's pioneering work of the 1960s and 70s has developed into a rich and fertile field of research with wide application. To be pedantic, whilst imprecision and vagueness *are* matters of uncertainty management, fuzzy logic is strictly a logic of graded properties and not of uncertainty. That is, a conclusion is true to a degree, rather than there being an expression of the likelihood of a conclusion being true. So the title is perhaps a little misleading.

Fuzzy Logic for the Management of Uncertainty is primarily a research text, with the majority of the papers concentrating on theoretical issues. There is a selection of papers describing more application oriented work, which give a good flavour for the state of the art in the application of fuzzy logic to knowledge-based systems. The applications are, however, slanted towards systems for diagnosis and prognosis. With the exception of a paper on a VLSI fuzzy chip, there is relatively little reference to fuzzy process controllers or imaging systems; both of which are in widespread use in commercial "white goods". The section on fuzzy logic for knowledge representation and elicitation also provides a number of papers which will be of interest to the application builder, as well as theoreticians.

With the exception of one or two survey papers, most of the papers in this volume would be quite at home in research journals. This does provide for a wide sweeping survey of the state-of-the-art, with contributions from many of the leading practitioners the world over. However, an introductory survey together with perhaps an additional commentary on each section by one or both of the editors would have added immensely to the accessibility and value of this book as a self contained volume. In addition, there is a level of typographical errors which would be intolerable in any self-respecting journal.

Reviewed by Paul J. Krause, Imperial Cancer Research Fund, London, UK

Foundations of fuzzy systems by Rudolf Kruse, Jörg Gebhardt and Frank Klawonn, John Wiley & Sons, Chichester, 1994, pp 265, £29.95, ISBN 0-471-94243-X.

This is an English translation of an original German edition which appeared about a year ago. The translation was carried out by professional translators with the support of the publishers, and is of a very high standard. The German edition was extremely well received as soon as it appeared, so this English language edition (produced as a slightly higher quality hardback) is to be welcomed as making the book accessible to a much wider audience.

Fuzzy sets and fuzzy logic still remain somewhat controversial in the west. There are still some lingering doubts about the applicability of this work, and the poor quality of some of the earlier treatises on fuzzy set theory has left many still suspicious of the integrity of the underlying formalism. This book must help in countering this, as the authors' technical ability is without question, *and* their work has well established links with industry which has given them direct experience with the development of successful applications.

After a short introduction, the book opens with a comprehensive account of the elements of fuzzy set theory and fuzzy logic. The exposition is clear, precise, nicely paced and comprehensive. For ease of reading, references are collected together in a final section which includes some supplementary remarks. These cover both historical material, an example application treated in depth, and pointers to research issues which arise from the earlier material. This final section concludes with some exercises which are designed to clarify and deepen understanding of the technical material presented in the chapter.

The same format is followed for the remaining two chapters. These are on possibilistic reasoning, including details on knowledge representation and efficient implementation, and fuzzy controllers, one of the major application domains.

The sum total is a very high quality text which advanced students, researchers and practitioners with an interest in the theory and application of fuzzy sets will find invaluable. The authors also address one of the remaining criticisms of the fuzzy set approach by including detailed analyses of the semantics of the underlying concepts as they are introduced. So, those that are still sceptical of the "fuzzy paradigm" might also benefit from reading this book!

Reviewed by Paul J. Krause, Advanced Computation Laboratory, Imperial Cancer Research Fund, 61 Lincoln's Inn Fields, London WC2A 3PX, UK

The Gödel programming language by Patricia Hill and John W. Lloyd, The MIT Press, 1994, pp 337, £40.50/$60.75, ISBN 0-262-08229-2.

For more than 20 years, the logic programming world has been dominated by the Prolog programming language and its many more specialized offspring. While this has, in some ways, led to a usefully focused community of users and researchers, it has also given rise to some false assumptions and conclusions both within and outside the logic programming community.

The most insidious of these assumptions is that Prolog and its siblings are, somehow, the be-all and end-all of logic programming languages, which is surely not the case. While Prolog is very well suited for many styles of programming work, such as rapid prototyping, knowledge-rich applications, and so on, it has drawbacks which cannot be ignored and which should not be thought of as representative of logic programming in the large. One obvious such drawback is floundering under negation, where a variable becomes bound during a program execution, but the binding is hidden from the rest of the program, leading to incorrect answers. Another is the lack of a declared language for logic programs: I dread to think how many hours I have wasted debugging trivial spelling mistakes in the names of (undeclared) Prolog atoms.

A more subtle problem, particularly for users not familiar with the ethos of logic programming, is the fixed left-right computation rule, which allows us to view the procedural, rather than the declarative, interpretation of programs as paramount. Reliance on the fixed computation strategy often results in Prolog programs which might as well have been implemented in C. This is, of course, missing the point of Kowalski's original equation "Algorithm = Logic + Control" (Kowalski, 1979); in logic programming, we want to think of the logic only, and let the control be looked after as automatically as possible.

There is clearly a long way to go before we reach Kowalski's vision of a programming system where issues of efficiency can be left entirely to the computer and the task of programming becomes that of simply stating the problem to be solved as clearly as possible. Even so, the current state of the art in programming technology means that we can do a lot better than Prolog, and the Gödel language, described in this new volume from MIT Press, is one example of how to do so.

Gödel is aimed at superceding the "core" versions of Prolog. That is to say, it is a mainstream, non-specialized programming language, which can and (I do not doubt) will be specialized in various ways in future. The main difference between Prolog and Gödel is that the Gödel system is significantly more carefully thought out in theoretical terms. Any program written in Prolog can be