

There are some good things to say about this book—the range of examples in the overview Chapter 1 seem well chosen, and the explanations in the chapter on knowledge representation are clear and easy to understand, as is the section on search methods in Chapter 4. In Chapter 5, the section choosing the development tool contains some practical advice—indeed, the whole chapter indicates the authors' appreciation of the practical problems in knowledge engineering.

Chapter 6, on the inference engine, likewise is clear and well-constructed, with the concepts argued in simple terms (in particular, the section on “dealing with uncertainty”). The user interface is discussed in Chapter 7, whilst in Chapter 8 we turn our attention to software engineered expert systems development integration—the first of the practical, rather than conceptual, chapters. Here there is more good advice in the sections on “Putting the expert system together” and “Integration of the expert system into the organization”, though annoying inconsistencies in the use of bold type spoil an otherwise readable text. Whilst I am sympathetic to the problems of “open-ended” system development, I think it is rather too strong to advise students that “The knowledge engineer should not allow management to say ‘Well, that looks fine . . . but I would like the system also to do . . .’.”

Nowhere in the chapter is there a statement of what software engineering *is*.

What these sections lack, is an entertaining style with *interesting* or stimulating examples. In the main the examples used are inane, and the complex examples worked using symbols are not easily accessible for the business reader.

The tutorial modules are generally well worked, and clearly illustrated. However, not having read the VP Expert user manual or documentation it is difficult to judge if they offer a large benefit over consulting the supplier's own literature. Irritations occur here too—the instructions on installing the software on hard disk are tucked away in an Appendix and not referred to from the text, so are only discovered when having devoured the whole book. The authors also attempt to stretch beyond its limit the very simple example of an expert system developed—advice on selection of a word processing package—and further annoy the reader by the inclusion of inane questions such as “estimate the amount of money that you can afford?”, ‘low’ or ‘high’, and “what kind of technical support do you want for your word processor?”, ‘fair’ or ‘good’.

The module on “Using databases and spreadsheets” with the knowledge base is clear, though the examples worked are trivial. The authors indicate the limitations of the software in its use of called data. There is only sketchy detail on how to create so-called ‘smart forms’ which adapt to the user's input, and even less on the use of the HYPERTEXT feature of VP Expert.

The final sections of the book include the VP Expert command hierarchy, a table of ASCII characters, and answers to exercises—many of which simply appear to repeat the content of the chapters (including a word-for-word repetition of the dictionary definition from Webster's New World Dictionary, with which the book opens!).

The book concludes with a rather short list of references, and an index of less than three pages for this 400 page student text book. The indexing is incomplete and inconsistent in style.

In conclusion, this is a disappointing, unsatisfying book. The title is misleading, since it is aimed at students, the context is variable in quality, and in terms of its textual and typographical errors it is appalling. I do not recommend it.

Knowledge based systems for test and diagnosis by G Saucier, A Ambler, M A Breuer (eds.), North Holland, Amsterdam, 1990, pp 256, G135.

Seeing a book title always induces certain expectations regarding the book's content. These expectations are based on the background knowledge of the reader. The title *Knowledge based systems for test and diagnosis* may raise the expectation that the book will contain a collection of papers/chapters describing KBSs to test devices and machines, or perhaps even students, as well as to perform a diagnostic task, for example in medicine or in repair/maintenance situations.

After reading the preface of the book, one soon comes to the conclusion that the book covers a much narrower domain, viz. the application of KBS for testing and diagnosis of electronic chips and boards.

The book contains four sections. It starts with an introduction in which some 20 systems are briefly described. There is, however, no synthesis of the common approaches, nor an analysis of the differences. The descriptions are also rather shallow (about half a page each).

The core of the book is divided into three chapters in which papers are grouped according to the functions they perform. The first chapter, taking about half the pages of the book, describes systems which are—according to the chapter heading—related to “test strategy planning and test expertise”. A majority of the contributions in this chapter describe systems which are utilized during the design phase of VLSI chips. More specifically, they deal with systems for the analysis of the testability of the design, as well as with proposing modifications in the design to make the final products testable.

Specifically, the 45 page long paper of Breuer et al. gives a good overview of their system as well as the various issues relevant during the design of testable VLSI chips. Also, the papers of Buonanno and Sciuto, Dislis et al. and Crastes de Paulet et al. present different approaches to the same problem. The advice such systems can give range from critiquing the proposed design up to proposals for modifications of the design. The last two contributions in this chapter are rather shallow and do not add much to the other papers.

The second chapter deals with “Knowledge-based diagnosis”. The first paper in this chapter is titled “Survey of diagnosis expert systems”. I would hardly consider this chapter as a survey. As a matter of fact, only two paradigms regarding knowledge representation for such systems are discussed: shallow knowledge and deep knowledge representation. Any survey of existing systems is lacking.

The next two papers in this chapter contain more relevant information. Malabocchia describes knowledge engineering issues in constraint-based diagnosis. This paper discusses some issues that are also relevant in domains other than electronics engineering. The paper of Kennett and Totton gives valuable insight in the process of building a diagnostic expert system. These two papers discuss applications at the level of diagnosis of failures of components on a printed circuit. The last paper describes a tool for building expert systems for technical diagnosis and maintenance. There is no report on experiences with the tool for building actual systems.

The last chapter deals with “Rule-based design verification and maintenance”. The first paper by Bolsens et al. describes in a formal way a system for the verification of MOS digital circuits. It gives in a comprehensive one way overview of the verification strategy, the approach to partitioning the design in lower level blocks and some issues for finding errors in the design. In the end some results are reported from test runs with the system. The next paper describes a similar system, but in lesser detail than the previous paper. The last paper addresses the issue of developing maintenance programs. Robach et al. describe an approach for determining optimal test sequences such that all components of a printed circuit are tested, taking into account the time for execution of the test as well as the test complexity. This paper also is relevant for other domains.

In conclusion, the book describes a number of KBSs in the domain of design and testing of chips and printed circuits. Therefore, it is much narrower in scope than the title may suggest. Readers who are familiar with terms such as BIST, BILBO, TDM, PLA, ADFT, FSM—they all appear in the introduction—may find valuable information in this book. For others, only a few papers may have valuable information.

A final remark is that many of the papers are written in rather poor English. I think that the editors would have served their readers better if they had also paid attention to this aspect of the publication.

Neural network design and the complexity of learning by J Stephen Judd, The MIT Press, Cambridge, MA, 1990, pp 134, £22.50.

Neural networks as a computational paradigm are currently of wide interest, discussion of this model no longer restricted to mathematicians, computer scientists, psychologists, and the like, but now often read about in science sections of newspapers, and mentioned on technology-orientated