judged according to the problem statement; and E-type (embedded) which solve a problem or implement an application in some real world domain, and are judged according to acceptability, value and level of satisfaction. Large scale systems are nearly always E-type, and therefore the notion of correctness does not apply. Interactive systems are another important group of E-type programs, and methodologies have been developed which require an initial theoretical model, and follow an iterative cycle of implementation and evaluation (e.g., Boehm, 1988; Carroll et al., 1991; Giddings, 1984; Sharples et al., 1989). Note that such methodologies differ from Partridge's Run-Understand-Debug-Edit (RUDE) approach in both the initial, empirically validated model, and the emphasis on a stage of empirical evaluation in the cycle. What is important in such methodologies is not that the problem be formally specifiable, but that the requirements be explicitly and precisely stated, and furthermore, that they be updated as understanding of the problem changes. Formal methods can be applied to assist in the rigourous analysis of these requirements, but are not the be-all and end-all of software engineering.

To summarize, on the plus side this book is a compendium of useful information about AI. It attempts to provide a balanced critique, in that for all the criticisms, AI is still presented as a worthwhile endeavour. However, the book is poorly structured, and poorly written, and so is irritating to read, and difficult to use as a reference book. Furthermore, some of the arguments are misleading, and some are misconceived. In particular, the chapter on methodology takes a particularly simplistic and caricatured view of computer science, as an excuse to present a dubious methodology which takes no account of recent work in software engineering.

References

Boehm, BW, 1988, "A spiral model of software development and enhancement" *IEEE Computer* **21** (5): 61–72.

Carroll, JM, Kellog, WA and Rosson, MB, 1991, "The task-artefact cycle" in: JM Carroll, ed., *Designing Interaction: Psychology at the Human-Computer Interface.* Cambridge University Press.

Giddings, RV, 1984, "Accommodating uncertainty in software design" *Communications of the ACM* **27** (5): 428–434.

Lehman, MM, 1980, "Programs, life cycles, and laws of software evolution" *Proceedings of the IEEE* **68** (9): 1060–1076.

Sharples, M, Goodlet, JS and Pemberton, L, 1989, "Developing a writer's assistant" in: N Williams and P Holt, eds., *Computers and Writing.* Intellect.

Formal methods in artificial intelligence by Allan Ramsay, Cambridge University Press, Cambridge, 1991, pp 289, £14.95 (paperback). ISBN 0 521 42421 6.

Formalism in AI and computer science by Philip Leith, Ellis Horwood, Chichester, 1990, pp 225, £29.95. ISBN 0 13 325549 2.

Formal techniques in artificial intelligence: a sourcebook by RB Banerji (Ed.), North Holland, Amsterdam, 1990, pp 437, Dfl 160.00. ISBN 0 444 88130 1.

A father and his son were driving to a football match. As they were crossing a level crossing the car stalled. The lights warned of an approaching train. The father frantically tried to restart the engine, but in his panic had flooded the carburettor and the car was hit by the train. An ambulance was called to the scene within minutes, but on the way to the hospital the father died. The son was just alive when the ambulance reached the hospital, but his condition was very serious and he needed immediate surgery. He was wheeled into the emergency operating room and the surgeon summoned. On seeing the boy, the surgeon blanched and stammered "I can't operate on this boy—he's my son".

I have paraphrased this puzzle from Douglas Hofstadter's book *Metamagical Themas.* He asks the reader to resolve the puzzle, if indeed there is a puzzle, bearing in mind that the surgeon is not lying, and there are no tricks of reincarnation or adoption or whatever. It is quite a well known

puzzle now, but if you have not seen it before, how long did it take you to solve it? And how did you feel about yourself when you did solve it if you didn't solve it straight away? (Typical reactions; one minute and a bit deflated, respectively). Now think about how you solved the puzzle and what made you make the assumptions that led you to see this puzzle in the first place. I think that in my case it was the word "default" in the chapter heading that triggered my thinking. Had I made a default assumption? We do this all the time. Yes, indeed I had. What was it? Well, in the society we live in, if I am told I am going to be introduced to a successful professional, I will assume that I am going to be introduced to a man.

There was some good traditional logical reasoning going on in my head. I had worked out from the accident scenario that the boy's father was not alive. I had then worked out from the statement of the surgeon that the boy's father was alive. I had then concluded that there was a contradiction (although I did not then proceed to conclude that everything was true as I seem to be entitled to do in classical logic). I then decided that I had to resolve the contradiction. Where was there a weakness in my reasoning? I had used a default rule that I would expect a surgeon to be man. The surgeon merely claimed to be a parent. I could reject my default and conclude that as the father was dead, the surgeon must be the mother. There I am, contradiction resolved, but still feeling a victim of rampant chauvinism.

But then I had to know that the son could not have two fathers before I knew there was a contradiction. If I was living in a more enlightened society, on the other hand, I would not have used my default rule (which with the present state of Britain's medical profession is usually quite a safe rule to use). I have used a bit of logic. Then I have reflected on my conclusions and found I am a victim of society; I've looked at my arguments and I have rejected one of them. This is all getting quite involved for a silly little problem.

So how are we getting along at formalising intelligence then? Here are three books with "formal" and "AI" in their titles. Two seem to think we are doing quite nicely; one disagrees.

Allan Ramsay's book, *Formal Methods in Artificial Intelligence,* is a new paperback edition of a well respected book which has been around for three or four years now. It is somewhat more wordy than many of the books in the Cambridge Tracts in Theoretical Computer Science series. This makes it more accessible than some, but it lacks the spark of insight of, for example, *Proofs and Types* or *Topology via Logic.* The book provides quite a useful introduction to propositional and predicate calculus, and automatic theorem proving therein, for the less mathematically inclined. Drawing on these for example, the discussion is then extended to modal logic and temporal and non-monotonic reasoning (my mistaken use of a default in the above is an example of non-monotonic reasoning).

It is a book that is ripe for final year undergraduates, MSc students in AI, perhaps even the (scientifically oriented) mythical general reader. But, I don't know, for a non-mathematical look at formal logical approaches to reasoning it wasn't that much fun to read. Nevertheless, if you are not too mathematically inclined and want to find out a bit more about the logicist approach to AI, this is a good book to start with.

So how far have we got with formalizing intelligence? Well, classical logic is useful, I can do a lot with that. I use defaults and there is something about that in this book. But how did I know which was the wrong default to have used? Why did I invent that default? And I do a lot of hand waving, and plausible reasoning. There is little about that in this book.

If you have drifted into computing science from mathematics or one of the "hard" sciences, the next book will drive you mad. *Formalism in AI and Computer Science,* by Philip Leith, is a text on "the philosophy and sociology of computer science". He admits that it is a somewhat contentious proposition that a foundation for the nature and practice of computing can be based on social theory. Very true. But there are some lessons that can be learned from the book if you are prepared to persevere through the somewhat jaundiced diatribes against contemporary formalists as reborn medieval pedants. Philip Leith has a strong research interest in the use of expert systems in the legal profession. There is a major difficulty there, as law is very much a social phenomenon and one may question using the rigidity of a rule-based system to model the reasoning of a legal expert. How

much of the day to day expert reasoning is a social phenomenon? Just as an expert's choices and decisions may be influenced by his or her interactions with colleagues, should not a machine be able to learn and to critique its reasoning? To interact with its colleagues, both mechanical and otherwise?

I baulked a bit at the extension of anti-formalist ideas to software engineering. There are grounds for looking at the sociology of the interactions between team members in large scale engineering projects. But constructing software is as much an engineering discipline as constructing aeroplanes or bridges, and needs formal tools for analysis and modelling as much as they do too.

Philip Leith has to shout loudly to be heard. The current state of the art in formalisms is a beginning but not an end. We do have to start looking at the social contexts in which computer software is used and built, and I hope the valuable points made in this book will gain an audience. I also fear he shouts a bit too loud sometimes.

Back to the mainstream. The final book, *Formal Techniques in Artificial Intelligence: A Sourcebook,* is an attempt to survey the theoretical basis to contemporary AI. It is not really a sourcebook, as most of the papers included in this volume are introductory survey papers rather than reprints of papers presenting original work. It is a rather idiosyncratic selection of papers, reflecting (quite naturally) the editor's main research interests, with a strong leaning towards formal logic, logic programming and automatic theorem proving. Approximate, uncertain and plausible reasoning are not really addressed, although there is some discussion of non-monotonic logics in Przymusinska and Przymusinski's paper on "Semantic Issues in Deductive Databases and Logic Programs". There is also a very readable paper by Grantham and Ungar on Qualitative Physics—basically an attempt to formalize hand waving arguments.

Actually, uncertain reasoning does receive some coverage in Philip Laird's chapter on machine learning. I've been trying to think of some way of succinctly categorizing the contents of the book, or of saying what is not included, and failed. Nevertheless, it is a valuable collection of survey papers, each introducing the state of the art in formalizing (not necessarily mathematicizing) some aspect of artificial intelligence. Several of them do assume a reasonably high degree of mathematical literacy, so this book is more for researchers or putative researchers in foundational aspects of AI; rather harder work for those who just want to use it to get a feel for what the theoreticians are about. But they are all clearly presented, with good motivational material and often having an informal presentation of the more technical results which appear later. Certainly a book which I would not like to be without.

Artificial intelligence has matured into a good hard scientific discipline. The reason for opening this review with a puzzle was just to emphasize that we should not lose sight of what it is we are trying to do. There is a danger, as Philip Leith argues, that the theoretical aspects may gain a momentum of their own and become entrenched in technicalisms which may have a beauty to the initiated, but be inessential to the task of moving forward the frontiers of the functionality of practical artificial intelligence. We still need visionaries!