

**Michael R. Genesereth and Nils J. Nilsson, *Logical Foundations of Artificial Intelligence* (Morgan Kaufmann, Los Altos, CA, 1987); 406 + xviii pages.**

Reviewed by: John F. Sowa

*IBM Systems Research, 500 Columbus Avenue,  
Thornwood, NY 10594, U.S.A.*

---

Every student of AI should be familiar with the material presented in this book. It covers important topics on logic in AI that no other textbook covers as well. Yet the book is not suitable for a first course in AI since many major topics are hardly mentioned. Nowhere, for example, will you get the idea that expert systems are the biggest commercial force that is driving interest in AI; the term "expert system" is mentioned on page 1, but never used again. LISP and PROLOG are mentioned in one sentence each on page 5. Computer vision and natural language processing are not discussed at all. Despite these omissions, the book would be good for a second course devoted to the use of logic in AI.

The book has thirteen chapters, each with a few pages of exercises and suggested readings. At the end are answers to all the exercises, a 38-page bibliography, and a 5-page index. The suggested readings and answers to exercises are especially valuable for a textbook. Following is a brief review of each chapter:

(1) *Introduction*. Most chapters entitled "Introduction" are written only because the authors didn't know where to begin with what they really wanted to say. This one has only four pages of overview followed by three pages of historical and bibliographical remarks, which are more lively and informative than the overview.

(2) *Declarative knowledge*. This chapter introduces the first-order predicate calculus and model theory. It applies the notation to examples from the blocks world, digital circuits, algebra, and simple English sentences. For a student who already knows logic, it is a good review, but it is too short for a beginner. An instructor who uses this book as a text should either require logic as a prerequisite or spend a lot of class time amplifying the text.

(3) *Inference*. This is a relatively short chapter on the rules of inference typically used for nonautomated theorem proving. The emphasis in the book is on the automated techniques presented in the next two chapters.

(4) *Resolution*. This is a standard treatment of resolution theorem proving. But the authors tend to give too much abstract discussion before introducing examples. In this chapter, there are 9 pages of detailed examples, but they follow 13 pages of abstract presentation of conversion to clausal form, unifica-

tion, resolution, and unsatisfiability. Following the examples are a section on soundness and completeness and a section on resolution with equality.

(5) *Resolution strategies*. This is one of the most readable chapters in the book. It is not an exhaustive treatment of all possible strategies, but it introduces the basic ideas and explains them clearly.

(6) *Nonmonotonic reasoning*. This is another good chapter, but the authors should present some examples earlier. Page 118, for example, has three paragraphs: the first one gives an informal definition of the closed world assumption; the second gives a formidable formal definition replete with Greek letters, subscripts, and parenthetical remarks; finally, the third paragraph gives a rather simple example. The presentation could be improved by moving the example ahead of the definition and showing what each Greek letter corresponds to in terms of the example. The treatment of circumscription bogs down on page 133: “Let  $M|\Delta|$  and  $M^*|\Delta|$  be two models of  $\Delta$ . (You may want to refer to Chapter 2 to review the definition of a model in logic.)” A reader who needs to review the definition of model will have problems in deciphering the barrage of symbols in the subsequent two paragraphs. An example at this point would be welcome. Circumscription is the only nonmonotonic approach treated in any depth. Default reasoning is only given three pages, and autoepistemic logic cited in the readings. These latter two systems deserve more coverage.

(7) *Induction*. This chapter is not hard to read, but it is too short. Many commercial expert system shells support automatic induction of rules from examples. This chapter should describe how they work with some typical examples. The two pages of historical remarks mention several learning systems in the AI literature. Some examples from those systems would have made the abstract description more useful.

(8) *Reasoning with uncertain beliefs*. The first three sections of this chapter are devoted to probabilities, Bayes’ theorem, and uncertainty in expert systems. The remaining five sections are based on a paper by one of the coauthors [1]. For an introductory text, that seems unbalanced, especially since they left out fuzzy logic and Dempster–Shafer’s evidential reasoning. They say that probabilistic logic is general enough to encompass these other methods. They may be true, but the book should show how they are related.

(9) *Knowledge and belief*. The first half of this chapter is fairly clear, but the presentation becomes hard to follow towards the end. One reason for the complexity is that the authors never explain the usual modal logics for necessity and possibility, but plunge directly into Kripke semantics for the knowledge operator.

(10) *Metaknowledge and metareasoning*. Metareasoning is an important topic that most textbooks ignore. However, the use of quoted strings for the object language makes the treatment more complex than it should be, since the metarules must parse the strings while interpreting them. A major advantage

of AI languages like LISP and PROLOG is the ability to represent embedded statements in a form that is already parsed and ready to evaluate—as lists in LISP or as functional forms in PROLOG. By using that technique, it is possible to write a simple PROLOG interpreter in only three PROLOG rules. Such an interpreter is more than a curiosity, as Kowalski [2] showed for a variety of problems. Starting with the simple three-rule interpreter, Wilson [3] showed how to add features to support explanations or use a different reasoning strategy; he even showed how to “compile” the object rules into the metalanguage to generate rules that run at the speed of the metalanguage, but simulate the reasoning strategy of the desired object language. Kowalski and Wilson present metareasoning as a practical programming technique, but Genesereth and Nilsson make it seem so inefficient that their readers would be discouraged from using it.

(11) *State and change*. This chapter formalizes the notions of state and action and discusses the frame problem. This is one of the few chapters built around an example; unfortunately, it is the rather trivial one of building stacks of three blocks on a table. The chapter gives no indication of how useful the approach might be for anything more complex.

(12) *Planning*. Like Chapter 11, this chapter is also limited to describing plans for building stacks of three blocks. Both of these chapters are clearly written, but they are not as detailed as Nilsson’s earlier books [4, 5], in which he solved a wider range of problems including the tower of Hanoi, the 8-puzzle, and symbolic integration. The older books, however, did not use logic for planning; instead, they used state search techniques with emphasis on the A\* algorithm. The new book merely mentions A\* as “a more complicated ordering rule.”

(13) *Intelligent-agent architecture*. This chapter describes a series of simple robot-like agents of increasing complexity. The simplest is a memoryless agent that selects an action that depends only on its immediate input. The richest are the deliberate agents that observe the environment, record its state in their database, perform some inference to decide upon an action, execute the action, and go back to observe.

Although I like the book, I wish the authors had highlighted some of the ideas and provided better motivation for them. PROLOG, for example, has given a strong impetus to logic-based techniques in AI. Yet PROLOG is first introduced in one sentence in the historical remarks at the end of Chapter 1. The next reference to it occurs in Chapter 5: “Depth-first backward resolution is the strategy used in PROLOG.” A student who never heard of PROLOG before would not remember the first sentence long enough to link it to the second. Horn clauses are mentioned several times, but nowhere does the book say that PROLOG happens to be based on the Horn clause subset of logic. Connections like that are the conceptual pegs on which students hang the details of the

subject matter. The book should explain their practical and theoretical significance and provide the reminders and cross-references that help students remember them.

Although model theory is introduced very early in the book, many of the relevant ideas are not highlighted. Section 2.3, with the title “Semantics,” develops the notion of interpretation in some detail, but it doesn’t introduce the word *model* until five pages later. I would have preferred to see “Model theory” as the title of that section. The first paragraph should have an informal discussion about models and their role in defining the semantics of a logical system. It would also help to point out that for propositional logic, model theory reduces to the theory of truth tables; that comment usually sparks a glimmer of recognition in students, who usually feel more comfortable with truth tables than with more general models. Since Chapter 6 on nonmonotonic reasoning requires the notion of minimal model, Section 2.3 should prepare the way by saying what it means for one model to be larger or smaller than another. In defining minimal model in Chapter 6, the authors talk about the extension of a relation, but the term *extension* is never defined. One of the exercises asks the student to distinguish  $p \Rightarrow q$ ,  $p \vdash q$ , and  $p \models q$ . But that distinction is so important that it should be emphasized in the text instead of being left for an exercise. I also dislike the authors’ term *logical implication* for  $p \models q$ ; I would prefer the term *semantic consequence*, which emphasizes the connection with models. The distinction is further blurred by a statement on page 58: “Earlier, it was suggested that there is a close connection between provability and logical implication. In fact, they are equivalent.” That is true only for standard first-order logic. The point of Gödel’s incompleteness theorem is that they are not equivalent for higher-order logic. For nonmonotonic logic and reasoning about knowledge and belief, the distinctions become important, and the authors should emphasize them.

There are other places where the choice of terminology or presentation may lead to confusion. I especially dislike the authors’ use of the term *database* to mean “a finite collection of sentences,” including arbitrarily complex quantified formulas. A conventional database, however, is normally restricted to ground level unit clauses. For most applications, it is important to distinguish the ground level clauses, which are subject to frequent updates, from the more general constraints, which change only when the problem or system is completely redefined. On page 36, the authors gloss over serious issues with the sentence “The easiest way to encode information about the number of objects having a property is to state the cardinality of the set of all objects having that property.” To talk about sets and permit quantification over them would require a major extension to first-order logic. They develop model theory with a set  $I$  of individuals and never say whether one individual might contain other individuals as parts. Yet in their examples, they introduce lists and permit quantification over them. The word “cardinality,” by the way, is one that I would like to banish from the computer science literature. It is necessary in

mathematics for talking about orders of infinity. But when all sets are finite, there is a perfectly respectable one-syllable synonym: *count*.

Although the book has a more complete treatment of logic and AI than anything else available, a number of important topics are missing. It should include truth maintenance and belief revision, which are active topics in the research literature that are also being implemented in commercial expert system shells. It should also say more about PROLOG and its relationship to pure first-order logic. At the very least, Chapter 5 on resolution strategies should have a section on PROLOG. Constraint logic programming is another outgrowth of the logic programming movement that is closer to pure logic than PROLOG. Right after Chapter 5, the book should have a full chapter on logic programming that starts with PROLOG, discusses other variations, and concludes with constraint logic programming. Chapter 9 on knowledge and belief should start with the modal logics of possibility and necessity. When teaching modal logic to students, I emphasize the applications to databases:

- A conventional database is a collection of atomic facts (ground level unit clauses) that describe some state of the world.
- Database constraints correspond to laws that determine which combinations of facts may be true in a possible world.
- Kripke's accessibility relation between possible worlds corresponds to a database update that does not violate any constraint.
- A proposition is necessarily true if it is provable from the laws; semantically, it must be true of all databases that are accessible from the current one by a permissible sequence of updates.
- A proposition is possible if it is consistent with the laws; semantically, it must be true in at least one database that is accessible by updates from the current one.
- Different axioms for modal logic correspond to different assumptions about the way the laws and facts vary from one database to another. Axiom S5, for example, corresponds to the assumption that all databases have the same laws. The weaker axioms would permit the laws to change from one database to another.

The accessibility relation is an abstraction that students find hard to understand. Relating it to database updates makes it much clearer. The formal basis for this treatment is taken from a paper by Dunn [6].

A book on the logical foundations of AI should also say something about what is built on those foundations. The historical and bibliographical remarks refer to many important applications; a few examples taken from the literature and explained in detail would help to show how logic can be used in practice. The book should also show how logic can clarify the reasoning strategies underlying other approaches. An analysis of the inference engine underlying some expert system shell or other AI program would be interesting. Section 2.9 mentions semantic networks and frames, but the versions that it describes are

trivial ones that do not capture any of the richness of the real systems used for practical applications. It says that frames include “provision for the association of procedural knowledge with the declarative knowledge stored as slot values. This allows us to express knowledge beyond that discussed. Unfortunately, it does not allow us to express this knowledge in declarative form.” That is a crucial issue. Why do frame systems require procedural knowledge? Can purely declarative systems based on logic handle the same kinds of applications equally well? If so, how? In order to make a serious case for logic, the book must come to grips with the most difficult problems handled by other approaches.

Finally, the authors ignore one of the most challenging problems in AI: knowledge acquisition for large, practical AI systems. The closest they come to discussing knowledge acquisition is in four pages on conceptualization at the beginning of Chapter 2. The problem of mapping natural language into logic is an important aspect of knowledge acquisition, but they discuss it in less than two pages in Section 2.8, which includes only four sample sentences. The authors certainly realize the importance of dealing with large applications, since they have spent years working at places that are developing major AI systems, such as the Stanford Knowledge Systems Laboratory and SRI International. I wish they had drawn on that experience to show how the techniques in this book can be scaled up to problems that are more than toys.

In summary, this is a good book for a course on logic in AI. The middle chapters are the best, with good presentations of resolution theorem proving, nonmonotonic logic, reasoning with uncertainty, and reasoning about knowledge and belief. Chapter 10 on metareasoning is more complicated than it should be, and the last three chapters are rather thin. With some revisions and extra material, the authors could produce a second edition that would raise the level from good to outstanding. Following are my suggestions for a future edition:

- Expand the bibliographical remarks in Chapter 1 into an essay on the role of logic in AI. It should address the controversies head on: “scruffies” versus “neats,” and proceduralists versus declarativists. Whereas the present book ignores everything not based on logic, the new chapter should give the “scruffies” their due; many of them have implemented major systems that actually work. The new chapter should discuss successes and failures of both the “scruffies” and the “neats” and analyze the underlying causes. By not discussing the limitations of current logic-based techniques, the book gives a lopsided view of AI. It could also lead to disillusionment if a student tried to apply the techniques to a problem that wasn’t suitable.

- Section 2.1 on conceptualization should become the central theme of Chapter 2; the section on model theory should be deferred to Chapter 3 or be made into a separate chapter by itself. The new version of Chapter 2 should address the problems of knowledge acquisition with examples from real

applications in addition to the toy examples there now. The work in naive physics would provide an excellent source of examples. Besides examples of successful mapping from language to logic, it should also present some of the failures, which are often more instructive. One natural language system, for example, mapped the sentences *Sam is a good musician* and *Sam is a bad cook* into the following formulas in logic:

$\text{good}(\text{Sam}) \wedge \text{musician}(\text{Sam}).$

$\text{bad}(\text{Sam}) \wedge \text{cook}(\text{Sam}).$

From these propositions, it answered yes to all of the following questions: *Is Sam a good cook? Is Sam a bad musician? Is Sam a good bad musician cook?* Besides being amusing, this anecdote teaches some basic points about both adjectives and nouns: some adjectives, like *good* and *bad*, do not describe a property of the individual, but of the role that the individual is playing; some nouns, like *musician* and *cook*, represent role types, rather than natural types [7]. McCawley [8] presents a wealth of examples that could be used to illustrate the issues in mapping language to logic.

– The remaining chapters could be augmented with some of the topics mentioned above. The scope and limitations of the techniques should be clearly stated. Chapter 12, for example, should give a detailed comparison of search algorithms like A\* with logic-based planning. If a procedural approach is better for some of these problems, then the book should say so. It should characterize the kinds of problems that logic can handle well and the problems that are better handled by other approaches. The only way to make a convincing case for logic is to be absolutely fair to other approaches as well.

There are also a number of misprints that I hope will be corrected before the second edition. Some are merely annoying, but others occur in formulas where they could be serious.

#### REFERENCES

1. Nilsson, N.J., Probabilistic logic, *Artificial Intelligence* **28** (1986) 71–87.
2. Kowalski, R.A., *Logic for Problem Solving* (North-Holland, New York, 1979).
3. Wilson, W.G., Programming techniques in Prolog, in: A. Walker, M. McCord, J.F. Sowa and W.G. Wilson (Eds.), *Knowledge Systems and Prolog* (Addison-Wesley, Reading, MA, 1987) 109–217.
4. Nilsson, N.J., *Problem-Solving Methods in Artificial Intelligence* (McGraw-Hill, New York, 1971).
5. Nilsson, N.J., *Principles of Artificial Intelligence* (Tioga, Palo Alto, CA, 1980).
6. Dunn, J.M., A truth value semantics for modal logic, in: H. Leblanc (Ed.), *Truth, Syntax, and Modality* (North-Holland, Amsterdam, 1973) 87–100.
7. Sowa, J.F., *Conceptual Structures: Information Processing in Mind and Machine* (Addison-Wesley, Reading, MA, 1984).
8. McCawley, J.D., *Everything that Linguists Have Always Wanted to Know about Logic* (Chicago University Press, Chicago, IL, 1981).