LOGICAL FOUNDATIONS of ARTIFICIAL INTELLIGENCE



MICHAEL R. GENESERETH NILS J. NILSSON Logical Foundations of Artificial Intelligence

Logical Foundations of Artificial Intelligence

Michael R. Genesereth and Nils J. Nilsson Stanford University

Morgan Kaufmann Publishers, Inc.

Editor and President Michael B. Morgan Production Manager Jennifer M. Ballentine Cover Designer Irene Imfeld Composition Arthur Ogawa and Laura Friedsam Book Design Beverly Kennon-Kelley Copy Editor Lyn Dupré Graduate Assistant Jamison Gray Production Assistant Todd R. Armstrong

Library of Congress Cataloging-in-Publication Data

```
Genesereth, Michael, 1948-
Logical foundations of artificial intelligence.
Bibliography: p.
Includes index.
1. Artificial intelligence. I. Nilsson, Nils,
1933- . II. Title.
Q335.G37 1986 006.3 87-5461
ISBN 0-934613-31-1
```

Reprinted with corrections May, 1988

Morgan Kaufmann Publishers, Inc. P.O. Box 50490, Palo Alto, CA 94303 © 1987 by Morgan Kaufmann Publishers Inc. All rights reserved. Printed in the United States of America

ISBN: 0-934613-31-1

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means—electronic, mechanical, photocopying, recording, or otherwise—without the prior written permission of the publisher.

90 89 88 5 4 3 2

to Maureen and Karen

Acknowledgments

WE ARE GRATEFUL for the support of Stanford University and of our many colleagues and students there. The second author also thanks SRI International for many years of an excellent research environment and the Palo Alto Laboratory of the Rockwell Scientific Center for valued help.

Many people read early drafts of this book. Some suggested major changes; some spotted small but dangerous errors. We thank them all and hope that the list below doesn't leave too many of them unmentioned.

James Allen William Greene Yoram Moses Mario Aranha **Benjamin Grosof** Karen Myers Marianne Baudinet Haym Hirsh Pandu Nayak Edward Brink Jane Hsu Eunok Paek Peter Cheeseman Joseph Jacobs Judea Pearl Jens Christensen Leslie Pack Kaelbling Donald Perlis Lai-Heng Chua Michael Kharitonov Liam Peyton Michael Dixon **Donald Knuth Charles Restivo** Kurt Konolige Stan Rosenschein **David Etherington David Fogelsong** Ami Kronfeld **Dave Singhal** Vladimir Lifschitz David Smith **Bob Flovd** John Lowrance Devika Subramanian Peter Friedland Matthew Ginsberg Kim McCall Tom Strat Bill McClung **Richard Waldinger** Andrew Golding Andres Modet Elizabeth Wolf Jamie Grav John Mohammed

Preface

THIS BOOK rests on two main assumptions. First, scientific and engineering progress in a discipline requires the invention and use of appropriate mathematical apparatus with which to express and unify good ideas. Second, symbolic logic forms a most important part of the mathematics of Artificial Intelligence (AI). Perhaps each of these assumptions needs some defense.

One might think that our first tenet should be noncontroversial. Yet in new fields, where the knowledge corpus is still mainly tied to practice and case studies, there often is substantial resistance to attempts at mathematicization. (One of us, for example, remembers hearing some electrical engineers during the 1950s complain about the irrelevance of differential equations to the study of electrical circuits and control systems!) We do not claim that knowing the mathematical ideas and techniques of a field is *all* that is needed to succeed in that field—either in research or in practice. We do note, however, that successful preparation in mature fields of science and engineering always includes a solid grounding in the mathematical apparatus of that field. This preparation provides the all-important framework needed to interpret, understand, and build the discipline.

Because the field of AI is relatively new, it is not surprising then that there are spirited debates between the "formalists" and the "experimentalists." The formalists claim that the experimentalists would progress faster if the latter had a more thorough understanding of various AI theoretical ideas. The experimentalists claim that the formalists would be more helpful if the latter were less concerned with form and more with content. Even if we were to grant that most of the advances in AI (or in any field of engineering) are inspired by experimentalists and that formalists serve mainly to "tidy up," it is nevertheless our opinion that the important new results in AI will be achieved by those researchers whose experiments are launched from the high platform of solid theory.

The theoretical ideas of older branches of engineering are captured in the language of mathematics. We contend that mathematical logic provides the basis for theory in AI. Although many computer scientists already count logic as fundamental to computer science in general, we put forward an even stronger form of the logic-is-important argument. In Chapters 1 and 2, we claim that AI deals mainly with the problem of representing and using *declarative* (as opposed to *procedural*) knowledge. Declarative knowledge is the kind that is expressed as sentences, and AI needs a language in which to state these sentences. Because the languages in which this knowledge usually is originally captured (natural languages such as English) are not suitable for computer representations, some other language with the appropriate properties must be used. It turns out, we think, that the appropriate properties include at least those that have been uppermost in the minds of logicians in their development of logical languages such as the predicate calculus. Thus, we think that any language for expressing knowledge in AI systems must be at least as expressive as the first-order predicate calculus.

If we are going to use a predicate-calculus-like language as a knowledgerepresentation language, then the theory that we develop about such systems must include parts of proof theory and model theory in logic. Our view is rather strong on this point: Anyone who attempts to develop theoretical apparatus relevant to systems that use and manipulate declaratively represented knowledge, and does so without taking into account the prior theoretical results of logicians on these topics, risks (at best) having to repeat some of the work done by the brightest minds of the twentieth century and (at worst) getting it wrong!

Given these two assumptions, then, the book develops the major topics of AI using the language and techniques of logic. These main topics are knowledge representation, reasoning, induction (a form of learning), and architectures for agents that reason, perceive, and act. We do not treat the various applications of these ideas in expert systems, naturallanguage processing, or vision. Separate books have been written about these application areas, and our goal here has been to concentrate on the common, fundamental ideas with which people in all these other areas ought to be familiar.

We propose the first-order predicate calculus as a language in which to represent the knowledge possessed by a reasoning agent about its world. We imagine that the agent exists in a world of objects, functions, and relations that form the basis for a *model* of the agent's predicate-calculus sentences. We propose that deductive inference is the major reasoning technique employed by an intelligent agent. Thus, we devote Chapters 1 through 5 of the book to a brief but complete presentation of the syntax and semantics of first-order predicate calculus, of logical deduction in general, and of resolution-refutation methods in particular.

The material in Chapters 1 through 5 and in Chapters 11 and 12 (on reasoning about actions and plans) is by now almost classical in AI. Much of the rest of the book is much closer to the current research frontier. We have attempted to draw together those recent research results that we think will, in time, become classical. We suspect that ours is the first textbook that treats these ideas. These topics include nonmonotonic reasoning, induction, reasoning with uncertain information, reasoning about knowledge and belief, metalevel representations and reasoning, and architectures for intelligent agents. We think that a field advances when important ideas migrate from research papers into textbooks. We are aware of the fact (and the reader should be too) that one takes one's chances with early migrations.

We should say something about why there is almost no mention in this book of the subject of *search*. Search is usually thought to be a cornerstone of AI. (One of us, in an earlier book, acknowledged the primacy of search in AI.) Nevertheless, as its title implies, this book is not intended to be a general introduction to the entire field of AI. A discussion of search would have detracted from the emphasis on logic that we wanted this book to have. In any case, search is well treated in other books on AI.

The book is aimed at advanced college seniors and graduate students intending to study further in AI. It assumes some knowledge of ideas of computer programming, although one does not have to program in order to learn from the book. The book also assumes mathematical sophistication. The reader who has already encountered some probability theory, logic, matrix algebra, list notation, and set theory will have an easier time with some parts of the book than will people less acquainted with these topics. Some of the more advanced sections might be skipped on a first reading; they are indicated by an asterisk (*) following the section title.

Exercises are included at the end of each chapter. (Solutions to the exercises are given at the end of the book.) Some ideas not presented in the text itself are introduced in the exercises. Most of these problems have been used successfully in classes taught by the authors at Stanford. The reader who is using this book for independent study is especially encouraged to work through the exercises. Even if the reader does not do the exercises, he should at least study those the solutions of which we have worked out—treating them as additional examples of concepts introduced in the book.

We briefly discuss important citations at the end of each chapter in sections entitled "Bibliographical and Historical Remarks." References to all cited works are collected at the end of the book. With these citations, Chapters 6 through 10 and 13 especially can be considered as a thorough introduction to the literature of these advanced topics.

At least three different languages are used in this book, and we have attempted to follow rigorously certain typographical conventions to help inform the reader which language is being used. Ordinary English is set in ordinary Roman type font (with italics for emphasis). Sentences in the predicate calculus are set in a typewriter-style font. Mathematical equations and formulas are set in a mathematical italic font. An explanation of these conventions with examples is given beginning on page xvii. The authors are grateful to Donald Knuth for inventing T_EX and to Leslie Lamport for developing IAT_EX. We used these typesetting systems from the very first days of preparing the manuscript, and they helped us immensely in dealing with the book's complex typography.

The authors would appreciate suggestions, comments, and corrections, which can be sent to them directly or in care of the publisher.

Contents

Typographical Conventions xv	ii
1 Introduction	1
1.1 Bibliographical and Historical Remarks	5
Exercises	8
2 Declarative Knowledge	9
2.1 Conceptualization	9
2.2 Predicate Calculus	3
2.3 Semantics	1
2.4 Blocks World Example	8
2.5 Circuits Example 2.5 2.5	9
2.6 Algebraic Examples	2
2.7 List Examples	3
2.8 Natural-Language Examples	5
2.9 Specialized Languages	6
2.10 Bibliographical and Historical Remarks	0
Exercises	2
3 Inference 4	5
3.1 Derivability	5
3.2 Inference Procedures	9

	3.3	Logical Implication	53
	3.4	Provability	55
	3.5	Proving Provability	58
	3.6	Bibliographical and Historical Remarks	62
	Exerc	eises	62
4 Re	solutio	n	63
	4.1	Clausal Form	63
	4.2	Unification	66
	4.3	Resolution Principle	69
	4.4	Resolution	71
	4.5	Unsatisfiability	75
	4.6	True-or-False Questions	75
	4.7	Fill-in-the-Blank Questions	76
	4.8	Circuits Example	78
	4.9	Mathematics Example	84
	4.10	Soundness and Completeness	85
	4.11	Resolution and Equality	89
	4.12	Bibliographical and Historical Remarks	92
	Exerc	\dot{ises}	92
5 Re	solutio	on Strategies	95
	5.1	Deletion Strategies	95
	5.2	Unit Resolution	98
	5.3	Input Resolution	99
	5.4	Linear Resolution	99
	5.5	Set of Support Resolution	100
	5.6	Ordered Resolution	102
	5.7	Directed Resolution	102
	5.8	Sequential Constraint Satisfaction	108
	5.9	Bibliographical and Historical Remarks	112
	Exerc	\dot{z} is estimated as \dot{z} and \dot{z} an	113
6 No	nmon	otonic Reasoning	115
	6.1	The Closed-World Assumption	117
	6.2	Predicate Completion	122
	6.3	Taxonomic Hierarchies and Default Reasoning	128
	6.4	Circumscription	132
	6.5	More General Forms of Circumscription	147
	6.6	Default Theories	152
	6.7	Bibliographical and Historical Remarks	155
	Exerc	zises	157
7 Inc	ductio	n	161
	7.1	Induction	. 161

7.2	Concept Formation	165
7.3	Experiment Generation	170
7.4	Bibliographical and Historical Remarks	. 174
Exerc	cises	176
8 Reasonir	ng with Uncertain Beliefs	177
8.1	Probabilities of Sentences	. 178
8.2	Using Bayes' Rule in Uncertain Reasoning	. 180
8.3	Uncertain Beasoning in Expert Systems	186
8.4	Probabilistic Logic	189
8.5	Probabilistic Entailment	193
8.6	Computations Appropriate for Small Matrices	197
87	Desling with Large Matrices	201
88	Probabilities Conditioned on Specific Information	201
0.0 9.0	Ribliographical and Historical Romarks	200
0.9 Even	Dibilographical and Historical Itemarks	204
Exerc		. 200
Knowled	ge and Belief	207
9.1	Preliminaries	208
9.2	Sentential Logics of Belief	. 209
9.3	Proof Methods	. 212
9.4	Nested Beliefs	. 214
9.5	Quantifying-In	. 216
9.6	Proof Methods for Quantified Beliefs	218
9.7	Knowing What Something Is	221
9.8	Possible-Worlds Logics	. 222
9.9	Properties of Knowledge	225
9.10	Properties of Belief	. 229
9 11	Group Knowledge	230
0.12	Equality Quantification and Knowledge	232
9.12	Bibliographical and Historical Remarks	234
5.15 Exerc	cises	236
0 Metakn	nowledge and Metareasoning	239
10.1	Metalanguage	. 240
10.2	Clausal Form	. 243
10.3	Resolution Principle	. 244
10.4	Inference Procedures	. 246
10.5	Derivability and Belief	. 247
10.6	Metalevel Reasoning	. 248
10.7	Bilevel Reasoning	. 251
10.8	Reflection*	. 255
10.9	Bibliographical and Historical Remarks	. 261
Exerc	cises	. 262

11 State ar	nd Change	263
11.1	States	263
11.2	Actions	267
11.3	The Frame Problem	271
11.4	Action Ordering	272
11.5	Conditionality	274
11.6	Bibliographical and Historical Remarks	280
Exerc	zises	282
12 Plannin	g	285
12.1	Initial State	285
12.2	Goals	286
12.3	Actions	287
12.4	Plans	289
12.5	Green's Method	290
12.6	Action Blocks	290
12.7	Conditional Plans	293
12.8	Planning Direction	294
12.9	Unachievability Pruning	296
12.10	State Alignment	297
12.11	Frame-Axiom Suppression	298
12.12	Goal Regression	299
12.13	State Differences	301
12.14	Bibliographical and Historical Remarks	304
Exerc	cises	305
13 Intellige	ent-Agent Architecture	307
13.1	Tropistic Agents	307
13.2	Hysteretic Agents	311
13.3	Knowledge-Level Agents	313
13.4	Stepped Knowledge-Level Agents	318
13.5	Fidelity*	320
13.6	Deliberate Agents*	325
13.7	Bibliographical and Historical Remarks	327
Exerc	cises \ldots	327
Answers to	o Exercises	329
A.1	Introduction	329
A.2	Declarative Knowledge	330
A.3	Inference	334
A.4	$Resolution \ . \ . \ . \ . \ . \ . \ . \ . \ . \ $	336
A.5	Resolution Strategies	340
A.6	Nonmonotonic Reasoning	342
A.7	Induction	347
A.8	Reasoning with Uncertain Beliefs	347

A.9	Knowledge and Belief				•		•	•	•		351
A.10	Metaknowledge and Metareasoning			•	•			 •		•	353
A.11	State and Change				•			 	•	•	355
A.12	Planning	•		•	•				•	•	359
A.13	Intelligent-Agent Architecture	•	•	•	•	• •		 	•	•	361
References											363
Index											401

Typographical Conventions

(1) Elements of a conceptualization—objects, functions, and relations are written in italics, as in the following example:

The extension of the on relation is the set $\{\langle a, b \rangle, \langle b, c \rangle, \langle d, e \rangle\}$.

(2) Expressions and subexpressions in predicate calculus are written in boldface, "typewriter" font, as in:

 $(\forall x Apple(x)) \lor (\exists x Pear(x))$

(3) We use lowercase Greek letters as metavariables ranging over predicate-calculus expressions and subexpressions. The use of these variables is sometimes mixed with actual predicate-calculus expressions, as in the following:

$$(\phi(\alpha) \lor P(\mathbf{A}) \Rightarrow \psi)$$

Sometimes, for mnemonic clarity, we use Roman characters, in mathematical font, as metavariables for relation constants and object constants, as in the following sample text:

Suppose we have a relation constant P and an object constant A such that $P(A) \Rightarrow P \land Q(B)$.

(4) We use uppercase Greek letters to denote sets of predicate calculus formulas, as in:

If there exists a proof of a sentence ϕ from a set Δ of premises and the logical axioms using Modus Ponens, then ϕ is said to be *provable* from Δ (written as $\Delta \vdash \phi$).

Since clauses are sets of literals, we also use uppercase Greek letters as variables ranging over clauses, as in:

Suppose that Φ and Ψ are two clauses that have been standardized apart.

(5) We use ordinary mathematical (not typewriter) font for writing metalogical formulas *about* predicate-calculus statements, as in:

```
If \sigma is an object constant, then \sigma^{I} \epsilon |I|.
```

Sometimes, metalogical formulas might contain predicate-calculus expressions:

 $\mathbf{A}^{I} = a$

- (6) We use an uppercase script \mathcal{T} to denote a predicate-calculus "theory."
- (7) Algorithms and programs are stated in typewriter font:

```
Procedure Resclution (Gamma)
Repeat Termination(Gamma) ==> Return(Success),
Phi <- Choose(Gamma), Psi <- Choose(Gamma),
Chi <- Choose(Resolvents(Phi,Psi)),
Gamma <- Concatenate(Gamma,[Chi])
End</pre>
```

(8) We use the notation {x/A} to denote the substitution in which the object constant A is substituted for the variable x. We use lowercase Greek letters for variables ranging over substitutions, as in:

Consider the combination of substitutions $\sigma \rho$.

(9) Lowercase ps and qs are used to denote probabilities:

```
p(P \land Q)
```

- (10) Sets of possible worlds are denoted by uppercase script letters, such as \mathcal{W} .
- (11) Vectors and matrices are denoted by boldface capital letters, such as \mathbf{V} and \mathbf{P} .
- (12) We also use boldface capital letters (and sequences of capital letters) to denote modal operators, such as **B** and **K**.

CHAPTER 1 Introduction

ARTIFICIAL INTELLIGENCE (AI) is the study of intelligent behavior. Its ultimate goal is a theory of intelligence that accounts for the behavior of naturally occurring intelligent entities and that guides the creation of artificial entities capable of intelligent behavior. Thus, AI is both a branch of science and a branch of engineering.

As engineering, AI is concerned with the concepts, theory, and practice of building intelligent machines. Examples of machines already within the reach of AI include expert systems that give advice about specialized subjects (such as medicine, mineral exploration, and finance), questionanswering systems for answering queries posed in restricted but large subsets of English and other natural languages, and theorem-proving systems for verifying that computer programs and digital hardware meet stated specifications. Ahead lie more flexible and capable robots, computers that can converse naturally with people, and machines capable of performing much of the world's "knowledge work."

As science, AI is developing concepts and vocabulary to help us to understand intelligent behavior in people and in other animals. Although there are necessary and important contributions to this same scientific goal by psychologists and by neuroscientists, we agree with the statement made by the sixteenth-century Italian philosopher Vico: Certum quod factum (one is certain of only what one builds). Aerodynamics, for