

Cambridge University Press

978-0-521-89943-7 - Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations

Yoav Shoham and Kevin Leyton-brown

Frontmatter

[More information](#)

Multiagent Systems

Multiagent systems combine multiple autonomous entities, each having diverging interests or different information. This comprehensive overview of the field offers a computer science perspective but also draws on ideas from game theory, economics, operations research, logic, philosophy, and linguistics. It will serve as a reference for researchers in each of these fields and be used as a text for advanced undergraduate or graduate courses.

The authors emphasize foundations to create a broad and rigorous treatment of their subject, with thorough presentations of distributed problem solving, non-cooperative game theory, multiagent communication and learning, social choice, mechanism design, auctions, cooperative game theory, and modal logics of knowledge and belief. For each topic, basic concepts are introduced, examples are given, proofs of key results are offered, and algorithmic considerations are examined. An appendix covers background material in probability theory, classical logic, Markov decision processes, and mathematical programming.

Yoav Shoham is a professor of computer science at Stanford University.

Kevin Leyton-Brown is an associate professor of computer science at the University of British Columbia.

Cambridge University Press
978-0-521-89943-7 - Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations
Yoav Shoham and Kevin Leyton-brown
Frontmatter
[More information](#)

Cambridge University Press

978-0-521-89943-7 - Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations

Yoav Shoham and Kevin Leyton-brown

Frontmatter

[More information](#)

Multiagent Systems

Algorithmic, Game-Theoretic,
and Logical Foundations

YOAV SHOHAM

Stanford University

KEVIN LEYTON-BROWN

University of British Columbia



CAMBRIDGE
UNIVERSITY PRESS

CAMBRIDGE
UNIVERSITY PRESS

32 Avenue of the Americas, New York NY 10013-2473, USA

Cambridge University Press is part of the University of Cambridge.
It furthers the University's mission by disseminating knowledge in the pursuit of
education, learning and research at the highest international levels of excellence.

www.cambridge.org
Information on this title: www.cambridge.org/9780521899437

© Yoav Shoham and Kevin Leyton-Brown 2009

This publication is in copyright. Subject to statutory exception
and to the provisions of relevant collective licensing agreements,
no reproduction of any part may take place without the written
permission of Cambridge University Press.

First published 2009

A catalogue record for this publication is available from the British Library
Library of Congress Cataloguing in Publication data

Shoham, Yoav.
Multiagent systems : algorithmic, game-theoretic, and logical
foundations / Yoav Shoham, Kevin Leyton-Brown.

p. cm.
Includes index.

ISBN 978-0-521-89943-7 (hardback)

1. Intelligent agents (Computer software) 2. Electronic data processing – Distributed
processing. I. Leyton-Brown, Kevin, 1975– II. Title.

QA76.76.I58575 2008
006.3 – dc22 2008012063

ISBN 978-0-521-89943-7 Hardback

Cambridge University Press has no responsibility for the persistence or accuracy of
URLs for external or third-party internet websites referred to in this publication,
and does not guarantee that any content on such websites is, or will remain, accurate
or appropriate.

Cambridge University Press
978-0-521-89943-7 - Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations
Yoav Shoham and Kevin Leyton-brown
Frontmatter
[More information](#)

To my wife, Noa, and my daughters, Maia, Talia, and Ella
—YS

To Jude
—KLB

Cambridge University Press
978-0-521-89943-7 - Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations
Yoav Shoham and Kevin Leyton-brown
Frontmatter
[More information](#)

Contents

<i>Credits and Acknowledgments</i>	<i>page xv</i>
<i>Introduction</i>	<i>xvii</i>
1 Distributed Constraint Satisfaction	1
1.1 Defining distributed constraint satisfaction problems	2
1.2 Domain-pruning algorithms	4
1.3 Heuristic search algorithms	8
1.3.1 The asynchronous backtracking algorithm	9
1.3.2 A simple example	11
1.3.3 An extended example: the four queens problem	13
1.3.4 Beyond the ABT algorithm	15
1.4 History and references	17
2 Distributed Optimization	19
2.1 Distributed dynamic programming for path planning	19
2.1.1 Asynchronous dynamic programming	19
2.1.2 Learning real-time A*	21
2.2 Action selection in multiagent MDPs	23
2.3 Negotiation, auctions, and optimization	27
2.3.1 From contract nets to auction-like optimization	27
2.3.2 The assignment problem and linear programming	29
2.3.3 The scheduling problem and integer programming	36
2.4 Social laws and conventions	43
2.5 History and references	45
3 Introduction to Noncooperative Game Theory: Games in Normal Form	47
3.1 Self-interested agents	47
3.1.1 Example: friends and enemies	48
3.1.2 Preferences and utility	49
3.2 Games in normal form	53
3.2.1 Example: the TCP user's game	54
3.2.2 Definition of games in normal form	55
3.2.3 More examples of normal-form games	56
3.2.4 Strategies in normal-form games	58
3.3 Analyzing games: from optimality to equilibrium	60
	vii

viii	<i>Contents</i>
3.3.1	Pareto optimality 60
3.3.2	Defining best response and Nash equilibrium 61
3.3.3	Finding Nash equilibria 62
3.3.4	Nash's theorem: proving the existence of Nash equilibria 64
3.4	Further solution concepts for normal-form games 71
3.4.1	Maxmin and minmax strategies 72
3.4.2	Minimax regret 75
3.4.3	Removal of dominated strategies 77
3.4.4	Rationalizability 79
3.4.5	Correlated equilibrium 81
3.4.6	Trembling-hand perfect equilibrium 83
3.4.7	ϵ -Nash equilibrium 83
3.5	History and references 85
4	Computing Solution Concepts of Normal-Form Games 87
4.1	Computing Nash equilibria of two-player, zero-sum games 87
4.2	Computing Nash equilibria of two-player, general-sum games 89
4.2.1	Complexity of computing a sample Nash equilibrium 89
4.2.2	An LCP formulation and the Lemke–Howson algorithm 91
4.2.3	Searching the space of supports 99
4.2.4	Beyond sample equilibrium computation 101
4.3	Computing Nash equilibria of n -player, general-sum games 102
4.4	Computing maxmin and minmax strategies for two-player, general-sum games 105
4.5	Identifying dominated strategies 106
4.5.1	Domination by a pure strategy 106
4.5.2	Domination by a mixed strategy 107
4.5.3	Iterated dominance 109
4.6	Computing correlated equilibria 110
4.7	History and references 111
5	Games with Sequential Actions: Reasoning and Computing with the Extensive Form 113
5.1	Perfect-information extensive-form games 113
5.1.1	Definition 113
5.1.2	Strategies and equilibria 115
5.1.3	Subgame-perfect equilibrium 117
5.1.4	Computing equilibria: backward induction 119
5.2	Imperfect-information extensive-form games 125
5.2.1	Definition 125
5.2.2	Strategies and equilibria 126
5.2.3	Computing equilibria: the sequence form 129
5.2.4	Sequential equilibrium 136
5.3	History and references 139

<i>Contents</i>	ix
6 Richer Representations: Beyond the Normal and Extensive Forms	141
6.1 Repeated games	142
6.1.1 Finitely repeated games	143
6.1.2 Infinitely repeated games	144
6.1.3 “Bounded rationality”: repeated games played by automata	147
6.2 Stochastic games	153
6.2.1 Definition	153
6.2.2 Strategies and equilibria	154
6.2.3 Computing equilibria	155
6.3 Bayesian games	156
6.3.1 Definition	157
6.3.2 Strategies and equilibria	160
6.3.3 Computing equilibria	163
6.3.4 <i>Ex post</i> equilibrium	165
6.4 Congestion games	166
6.4.1 Definition	166
6.4.2 Computing equilibria	167
6.4.3 Potential games	168
6.4.4 Nonatomic congestion games	170
6.4.5 Selfish routing and the price of anarchy	172
6.5 Computationally motivated compact representations	176
6.5.1 The expected utility problem	177
6.5.2 Graphical games	179
6.5.3 Action-graph games	181
6.5.4 Multiagent influence diagrams	183
6.5.5 GALA	186
6.6 History and references	187
7 Learning and Teaching	189
7.1 Why the subject of “learning” is complex	189
7.1.1 The interaction between learning and teaching	189
7.1.2 What constitutes learning?	190
7.1.3 If learning is the answer, what is the question?	192
7.2 Fictitious play	195
7.3 Rational learning	200
7.4 Reinforcement learning	204
7.4.1 Learning in unknown MDPs	205
7.4.2 Reinforcement learning in zero-sum stochastic games	205
7.4.3 Beyond zero-sum stochastic games	208
7.4.4 Belief-based reinforcement learning	209
7.5 No-regret learning and universal consistency	209
7.6 Targeted learning	211
7.7 Evolutionary learning and other large-population models	212
7.7.1 The replicator dynamic	213

x	<i>Contents</i>
7.7.2	Evolutionarily stable strategies 216
7.7.3	Agent-based simulation and emergent conventions 219
7.8	History and references 221
8	Communication 223
8.1	“Doing by talking” I: cheap talk 223
8.2	“Talking by doing”: signaling games 227
8.3	“Doing by talking” II: speech-act theory 229
8.3.1	Speech acts 229
8.3.2	Rules of conversation 230
8.3.3	A game-theoretic view of speech acts 232
8.3.4	Applications 235
8.4	History and references 238
9	Aggregating Preferences: Social Choice 241
9.1	Introduction 241
9.1.1	Example: plurality voting 241
9.2	A formal model 242
9.3	Voting 244
9.3.1	Voting methods 244
9.3.2	Voting paradoxes 246
9.4	Existence of social functions 247
9.4.1	Social welfare functions 248
9.4.2	Social choice functions 251
9.5	Ranking systems 255
9.6	History and references 258
10	Protocols for Strategic Agents: Mechanism Design 261
10.1	Introduction 261
10.1.1	Example: strategic voting 261
10.1.2	Example: buying a shortest path 262
10.2	Mechanism design with unrestricted preferences 263
10.2.1	Implementation 264
10.2.2	The revelation principle 265
10.2.3	Impossibility of general, dominant-strategy implementation 267
10.3	Quasilinear preferences 268
10.3.1	Risk attitudes 269
10.3.2	Mechanism design in the quasilinear setting 271
10.4	Efficient mechanisms 276
10.4.1	Groves mechanisms 276
10.4.2	The VCG mechanism 280
10.4.3	VCG and individual rationality 282
10.4.4	VCG and weak budget balance 283
10.4.5	Drawbacks of VCG 284
10.4.6	Budget balance and efficiency 288

<i>Contents</i>	xi
10.4.7 The AGV mechanism	288
10.5 Beyond efficiency	289
10.5.1 What else can be implemented in dominant strategies?	290
10.5.2 Tractable Groves mechanisms	292
10.6 Computational applications of mechanism design	293
10.6.1 Task scheduling	294
10.6.2 Bandwidth allocation in computer networks	296
10.6.3 Multicast cost sharing	298
10.6.4 Two-sided matching	301
10.7 Constrained mechanism design	307
10.7.1 Contracts	308
10.7.2 Bribes	309
10.7.3 Mediators	310
10.8 History and references	311
11 Protocols for Multiagent Resource Allocation: Auctions	315
11.1 Single-good auctions	315
11.1.1 Canonical auction families	316
11.1.2 Auctions as Bayesian mechanisms	318
11.1.3 Second-price, Japanese, and English auctions	319
11.1.4 First-price and Dutch auctions	321
11.1.5 Revenue equivalence	323
11.1.6 Risk attitudes	326
11.1.7 Auction variations	327
11.1.8 “Optimal” (revenue-maximizing) auctions	328
11.1.9 Collusion	330
11.1.10 Interdependent values	333
11.2 Multiunit auctions	336
11.2.1 Canonical auction families	336
11.2.2 Single-unit demand	337
11.2.3 Beyond single-unit demand	340
11.2.4 Unlimited supply: random sampling auctions	342
11.2.5 Position auctions	344
11.3 Combinatorial auctions	346
11.3.1 Simple combinatorial auction mechanisms	348
11.3.2 The winner determination problem	349
11.3.3 Expressing a bid: bidding languages	352
11.3.4 Iterative mechanisms	357
11.3.5 A tractable mechanism	359
11.4 Exchanges	361
11.4.1 Two-sided auctions	361
11.4.2 Prediction markets	362
11.5 History and references	364

xii	<i>Contents</i>
12 Teams of Selfish Agents: An Introduction to Coalitional Game Theory	367
12.1 Coalitional games with transferable utility	367
12.1.1 Definition	368
12.1.2 Examples	368
12.1.3 Classes of coalitional games	370
12.2 Analyzing coalitional games	371
12.2.1 The Shapley value	372
12.2.2 The core	374
12.2.3 Refining the core: ϵ -core, least core, and nucleolus	378
12.3 Compact representations of coalitional games	381
12.3.1 Weighted majority games and weighted voting games	381
12.3.2 Weighted graph games	382
12.3.3 Capturing synergies: a representation for superadditive games	384
12.3.4 A decomposition approach: multi-issue representation	385
12.3.5 A logical approach: marginal contribution nets	386
12.4 Further directions	388
12.4.1 Alternative coalitional game models	388
12.4.2 Advanced solution concepts	389
12.5 History and references	390
13 Logics of Knowledge and Belief	393
13.1 The partition model of knowledge	393
13.1.1 Muddy children and warring generals	393
13.1.2 Formalizing intuitions about the partition model	394
13.2 A detour to modal logic	396
13.2.1 Syntax	398
13.2.2 Semantics	398
13.2.3 Axiomatics	399
13.2.4 Modal logics with multiple modal operators	399
13.2.5 Remarks about first-order modal logic	400
13.3 S5: An axiomatic theory of the partition model	400
13.4 Common knowledge, and an application to distributed systems	403
13.5 Doing time, and an application to robotics	406
13.5.1 Termination conditions for motion planning	407
13.5.2 Coordinating robots	410
13.6 From knowledge to belief	412
13.7 Combining knowledge and belief (and revisiting knowledge)	413
13.8 History and references	418
14 Beyond Belief: Probability, Dynamics, and Intention	421
14.1 Knowledge and probability	421
14.2 Dynamics of knowledge and belief	425
14.2.1 Belief revision	426

<i>Contents</i>	xiii
14.2.2 Beyond AGM: update, arbitration, fusion, and friends	430
14.2.3 Theories of belief change: a summary	436
14.3 Logic, games, and coalition logic	436
14.4 Towards a logic of “intention”	438
14.4.1 Some preformal intuitions	438
14.4.2 The road to hell: elements of a formal theory of intention	440
14.4.3 Group intentions	443
14.5 History and references	445
 Appendices: Technical Background	 447
A Probability Theory	449
A.1 Probabilistic models	449
A.2 Axioms of probability theory	449
A.3 Marginal probabilities	450
A.4 Conditional probabilities	450
 B Linear and Integer Programming	 451
B.1 Linear programs	451
B.2 Integer programs	453
 C Markov Decision Problems (MDPs)	 455
C.1 The model	455
C.2 Solving known MDPs via value iteration	455
 D Classical Logic	 457
D.1 Propositional calculus	457
D.2 First-order logic	458
<i>Bibliography</i>	459
<i>Index</i>	473

Cambridge University Press
978-0-521-89943-7 - Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations
Yoav Shoham and Kevin Leyton-brown
Frontmatter
[More information](#)

Credits and Acknowledgments

We should start off by explaining the order of authorship. Yoav conceived of the project and started it, in late 2001, working on it alone and with several colleagues (see below). Sometime in 2004 Yoav realized he needed help if the project were ever to come to conclusion, and he enlisted the help of Kevin. The result was a true partnership and a complete overhaul of the material. The current book is vastly different from the draft that existed when the partnership was formed—in depth, breadth, and form. Yoav and Kevin have made equal contributions to the book; the order of authorship reflects the history of the book, but nothing else.

In six years of book-writing we accumulated many debts. The following is our best effort to acknowledge those. If we omit any names, it is due solely to our poor memories and record keeping, and we apologize in advance.

When the book started out, Teg Grenager served as a prolific ghost writer. While little of the original writing remains (though some does, for example, in Section 8.3.1 on speech acts), the project would not have gotten off the ground without him.

Several past and present graduate students made substantial contributions. Chapter 12 (coalitional games) is based entirely on writing by Sam Leong, who was also closely involved in the editing. Section 3.3.4 (the existence of Nash equilibria) and parts of Section 6.5 (compact game representations) are based entirely on writing by Albert Xin Jiang, who also worked extensively with us to refine the material. Albert also contributed to the proof of Theorem 3.4.4 (the minmax theorem). Some of the material in Chapter 4 on computing solution concepts is based on writing by Ryan Porter, who also contributed much of the material in Section 6.1.3 (bounded rationality). The material in Chapter 7 (multiagent learning) is based in part on joint work with Rob Powers, who also contributed text. Section 10.6.4 (mechanisms for matching) is based entirely on text by Baharak Rastegari, and David R. M. Thompson contributed material to Sections 10.6.3 (mechanisms for multicast routing) and 6.3.4 (*ex post* equilibria). Finally, all of the past and present students listed here offered invaluable comments on drafts. Other students also offered valuable comments. Samantha Leung deserves special mention; we also received useful feedback from Michael Cheung, Matthew Chudek, Farhad Ghassemi, Ryan Golbeck, James Wright, and Erik Zawadzki. We apologize in advance to any others whose names we have missed.

Several of our colleagues generously contributed material to the book, in addition to lending their insight. They include Geoff Gordon (Matlab code to generate Figure 3.13, showing the saddle point for zero-sum games), Carlos Guestrin (material on action selection in distributed MDPs in Section 2.2, and Figure 1.1,

showing a deployed sensor network), Michael Littman (Section 5.1.4 on computing all subgame-perfect equilibria), Amnon Meisels (much of the material on heuristic distributed constraint satisfaction in Chapter 1), Marc Pauly (material on coalition logic in Section 14.3), Christian Shelton (material on computing Nash equilibria for n -player games in Section 4.3), and Moshe Tennenholtz (material on restricted mechanism design in Section 10.7). We thank Éva Tardos and Tim Roughgarden for making available notes that we drew on for our proofs of Lemma 3.3.14 (Sperner's lemma) and Theorem 3.3.21 (Brouwer's fixed-point theorem for simplices), respectively.

Many colleagues around the world generously gave us comments on drafts, or provided counsel otherwise. Felix Brandt and Vince Conitzer deserve special mention for their particularly detailed and insightful comments. Other colleagues to whom we are indebted include Alon Altman, Krzysztof Apt, Navin A. R. Bhat, Ronen Brafman, Yiling Chen, Yossi Feinberg, Jeff Fletcher, Nando de Freitas, Raul Hakli, Joe Halpern, Jason Hartline, Jean-Jacques Herings, Ramesh Johari, Bobby Kleinberg, Daphne Koller, Fangzhen Lin, David Parkes, David Poole, Maurice Queyranne, Tim Roughgarden, Tuomas Sandholm, Peter Stone, Nikos Vlasis, Mike Wellman, Bob Wilson, Mike Wooldridge, and Dongmo Zhang.

Many others pointed out errors in the first printing of the book through our errata wiki: B. J. Buter, Nicolas Dubebout, Marco Guazzzone, Joel Kammet, Nicolas Lambert, Nimalan Mahendran, Mike Rogers, Ivomar Brito Soares, Michael Styer, Sean Sutherland, Grigorios Tsoumakas, Steve Wolfman, and James Wright.

Several people provided critical editorial and production assistance of various kinds. Most notably, David R. M. Thompson overhauled our figures, code formatting, bibliography, and index. Chris Manning was kind enough to let us use the $\text{\LaTeX}$ macros from his own book, and Ben Galin added a few miracles of his own. Ben also composed several of the examples, found some bugs, drew many figures, and more generally for two years served as an intelligent jack-of-all-trades on this project. Erik Zawadzki helped with the bibliography and with some figures. Maia Shoham helped with some historical notes and bibliography entries, as well as with some copy-editing.

We thank all these friends and colleagues. Their input has contributed to a better book, but of course they are not to be held accountable for any remaining shortcomings. We claim sole credit for those.

We also thank Cambridge University Press for publishing the book, and for their enlightened online-publishing policy, which has enabled us to provide the broadest possible access to it. Specific thanks to Lauren Cowles, an editor of unusual intelligence, good judgment, and sense of humor.

Last, and certainly not the least, we thank our families, for supporting us through this time-consuming project. We dedicate this book to them, with love.

Introduction

Imagine a personal software agent engaging in electronic commerce on your behalf. Say the task of this agent is to track goods available for sale in various online venues over time, and to purchase some of them on your behalf for an attractive price. In order to be successful, your agent will need to embody your preferences for products, your budget, and in general your knowledge about the environment in which it will operate. Moreover, the agent will need to embody your knowledge of other similar agents with which it will interact (e.g., agents who might compete with it in an auction or agents representing store owners)—including their own preferences and knowledge. A collection of such agents forms a multiagent system. The goal of this book is to bring under one roof a variety of ideas and techniques that provide foundations for modeling, reasoning about, and building multiagent systems.

Somewhat strangely for a book that purports to be rigorous, we will not give a precise definition of a multiagent system. The reason is that many competing, mutually inconsistent answers have been offered in the past. Indeed, even the seemingly simpler question—What is a (single) agent?—has resisted a definitive answer. For our purposes, the following loose definition will suffice: Multiagent systems are those systems that include multiple autonomous entities with either diverging information or diverging interests, or both.

Scope of the book

The motivation for studying multiagent systems often stems from interest in artificial (software or hardware) agents, for example software agents living on the Internet. Indeed, the Internet can be viewed as the ultimate platform for interaction among self-interested, distributed computational entities. Such agents can be trading agents of the sort discussed above, “interface agents” that facilitate the interaction between the user and various computational resources (including other interface agents), game-playing agents that assist (or replace) human players in a multiplayer game, or autonomous robots in a multi-robot environment. However, although the material is written by computer scientists with computational sensibilities, it is quite interdisciplinary, and the material is in general fairly abstract. Many of the ideas apply to—and indeed are often taken from—inquiries about human individuals and institutions.

The material spans disciplines as diverse as computer science (including artificial intelligence, theory, and distributed systems), economics (chiefly

microeconomic theory), operations research, analytic philosophy, and linguistics. The technical material includes logic, probability theory, game theory, and optimization. Each of the topics covered easily supports multiple independent books and courses, and this book does not aim to replace them. Rather, the goal has been to gather the most important elements from each discipline and weave them together into a balanced and accurate introduction to this broad field. The intended reader is a graduate student or an advanced undergraduate, prototypically, but not necessarily, in computer science.

Because the umbrella of multiagent systems is so broad, the questions of what to include in any book on the topic and how to organize the selected material are crucial. To begin with, this book concentrates on foundational topics rather than surface applications. Although we will occasionally make reference to real-world applications, we will do so primarily to clarify the concepts involved; this is despite the practical motivations professed earlier. And so this is the wrong text for the reader interested in a practical guide to building this or that sort of software. The emphasis is rather on important concepts and the essential mathematics behind them. The intention is to delve in enough detail into each topic to be able to tackle some technical material, and then to point the reader in the right directions for further education on particular topics.

Our decision was thus to include predominantly established, rigorous material that is likely to withstand the test of time, and to emphasize computational perspectives where appropriate. This still left us with vast material from which to choose. In understanding the selection made here, it is useful to keep in mind the following keywords: *coordination*, *competition*, *algorithms*, *game theory*, and *logic*. These terms will help frame the chapter overview that follows.

Overview of the chapters

Starting with issues of coordination, we begin in **Chapter 1** and **Chapter 2** with distributed problem solving. In these multiagent settings there is no question of agents' individual preferences; there is some global problem to be solved, but for one reason or another it is either necessary or advantageous to distribute the task among multiple agents, whose actions may require coordination. These chapters are thus strongly algorithmic. The first one looks at distributed constraint-satisfaction problems. The latter addresses distributed optimization and specifically examines four algorithmic methods: distributed dynamic programming, action selection in distributed MDPs, auction-like optimization procedures for linear and integer programming, and social laws.

We then begin to embrace issues of competition as well as coordination. Whereas the area of multiagent systems is not synonymous with game theory, there is no question that game theory is a key tool to master within the field, and so we devote several chapters to it. **Chapters 3, 5, and 6** constitute a crash course in noncooperative game theory. They cover, respectively, the normal form, the extensive form, and a host of other game representations. In these chapters, as in others that draw on game theory, we culled the material that in our judgment

is needed in order to be a knowledgeable consumer of modern-day game theory. Unlike traditional game theory texts, we also include discussion of algorithmic considerations. In the context of the normal-form representation, that material is sufficiently substantial to warrant its own chapter, **Chapter 4**.

We then switch to two specialized topics in multiagent systems. In **Chapter 7** we cover multiagent learning. The topic is interesting for several reasons. First, it is a key facet of multiagent systems. Second, the very problems addressed in the area are diverse and sometimes ill understood. Finally, the techniques used, which draw equally on computer science and game theory (as well as some other disciplines), are not straightforward extensions of learning in the single-agent case.

In **Chapter 8** we cover another element unique to multiagent systems: communication. We cover communication in a game-theoretic setting, as well as in cooperative settings traditionally considered by linguists and philosophers (except that we see that there too a game-theoretic perspective can creep in).

Next is a three-chapter sequence that might be called “protocols for groups.” **Chapters 9** covers social-choice theory, including voting methods. This is a nonstrategic theory, in that it assumes that the preferences of agents are known, and the only question is how to aggregate them properly. **Chapter 10** covers mechanism design, which looks at how such preferences can be aggregated by a central designer even when agents *are* strategic. Finally, **Chapter 11** looks at the special case of auctions.

Chapter 12 covers coalitional game theory, in recent times somewhat neglected within game theory and certainly underappreciated in computer science.

The material in Chapters 1–12 is mostly Bayesian and/or algorithmic in nature. And thus the tools used in them include probability theory, utility theory, algorithms, Markov decision problems (MDPs), and linear/integer programming. We conclude with two chapters on logical theories in multiagent systems. In **Chapter 13** we cover modal logic of knowledge and belief. This material hails from philosophy and computer science, but it turns out to dovetail very nicely with the discussion of Bayesian games in Chapter 6. Finally, in **Chapter 14** we extend the discussion in several directions—we discuss how beliefs change over time, logical models of games, and how one might begin to use logic to model motivational attitudes (such as “intention”) in addition to the informational ones (knowledge, belief).

Required background

The book is rigorous and requires mathematical thinking, but only basic background knowledge. In much of the book we assume knowledge of basic computer science (algorithms, complexity) and basic probability theory. In more technical parts we assume familiarity with Markov decision problems (MDPs), mathematical programming (specifically, linear and integer programming), and classical logic. All of these (except basic computer science) are covered briefly in **appendices**, but those are meant as refreshers and to establish notation, not as a

substitute for background in those subjects. This is true in particular of probability theory. However, above all, a prerequisite is a capacity for clear thinking.

How to teach (and learn) from this book

There are partial dependencies among the 13 chapters. To understand them, it is useful to think of the book as consisting of the following “blocks.”

- **Block 1**, Chapters 1–2: Distributed problem solving
- **Block 2**, Chapters 3–6: Noncooperative game theory
- **Block 3**, Chapter 7: Learning
- **Block 4**, Chapter 8: Communication
- **Block 5**, Chapters 9–11: Protocols for groups
- **Block 6**, Chapter 12: Coalitional game theory
- **Block 7**, Chapters 13–14: Logical theories

Within every block there is a sequential dependence (except within Block 1, in which the sections are largely independent of each other). Among the blocks, however, there is only one strong dependence: Blocks 3, 4, and 5 each depend on some elements of noncooperative game theory and thus on block 2 (though none requires the entire block). Otherwise there are some interesting local pairwise connections between blocks, but none that require that both blocks be covered, whether sequentially or in parallel.

Given this weak dependence among the chapters, there are many ways to craft a course out of the material, depending on the background of the students, their interests, and the time available. The book’s Web site

<http://www.masfoundations.org>

contains several specific syllabi that have been used by us and other colleagues, as well as additional resources for both students and instructors.

On pronouns and gender

We use male pronouns to refer to agents throughout the book. We debated this between us, not being happy with any of the alternatives. In the end we reluctantly settled on the “standard” male convention rather than the reverse female convention or the grammatically dubious “they.” We urge the reader not to read patriarchal intentions into our choice.