MULTIAGENT SYSTEMS

Algorithmic, Game-Theoretic, and Logical Foundations

Yoav Shoham
Stanford University

Kevin Leyton-Brown
University of British Columbia

© all rights reserved

March 28, 2008
# Contents

*Credits and Acknowledgments*  
 xi

*Introduction*  
 xiii

## 1 Distributed Constraint Satisfaction  
1.1 Defining distributed constraint satisfaction problems  
2  
1.2 Domain-pruning algorithms  
5  
1.3 Heuristic search algorithms  
9  
\subsection*{1.3.1} The asynchronous backtracking algorithm  
10  
1.3.2 A simple example  
12  
1.3.3 An extended example: the four-queen problem  
13  
1.3.4 Beyond the ABT algorithm  
17  
1.4 History and references  
18

## 2 Distributed Optimization  
19  
2.1 Distributed dynamic programming for path planning  
19  
\subsection*{2.1.1} Asynchronous dynamic programming  
19  
\subsection*{2.1.2} Learning real-time A*  
21  
2.2 Action selection in multiagent MDPs  
22  
2.3 Negotiation, auctions and optimization  
28  
\subsection*{2.3.1} Introduction: from contract nets to auction-like optimization  
28  
\subsection*{2.3.2} The assignment problem and linear programming  
31  
\subsection*{2.3.3} The scheduling problem and integer programming  
37  
2.4 Social laws and conventions  
45  
2.5 History and references  
47

## 3 Introduction to Noncooperative Game Theory: Games in Normal Form  
49  
3.1 Self-interested agents  
49  
\subsection*{3.1.1} Example: friends and enemies  
50  
\subsection*{3.1.2} Preferences and utility  
51  
3.2 Games in normal form  
56  
\subsection*{3.2.1} Example: the TCP user’s game  
56
### Contents

3.2.2 Definition of games in normal form 57  
3.2.3 More examples of normal-form games 58  
3.2.4 Strategies in normal-form games 61  

3.3 Analyzing games: from optimality to equilibrium 63  
3.3.1 Pareto optimality 63  
3.3.2 Defining best response and Nash equilibrium 64  
3.3.3 Finding Nash equilibria 65  
3.3.4 Nash’s theorem: proving the existence of Nash equilibria 68  

3.4 Further solution concepts for normal-form games 75  
3.4.1 Maxmin and minmax strategies 76  
3.4.2 Minimax regret 79  
3.4.3 Removal of dominated strategies 81  
3.4.4 Rationalizability 84  
3.4.5 Correlated equilibrium 86  
3.4.6 Trembling-hand perfect equilibrium 88  
3.4.7 \( \epsilon \)-Nash equilibrium 88  

3.5 History and references 90  

4 Computing Solution Concepts of Normal-Form Games 93  
4.1 Computing Nash equilibria of two-player, zero-sum games 93  
4.2 Computing Nash equilibria of two-player, general-sum games 95  
4.2.1 Complexity of computing a sample Nash equilibrium 96  
4.2.2 An LCP formulation and the Lemke–Howson algorithm 97  
4.2.3 Searching the space of supports 106  
4.2.4 Beyond sample equilibrium computation 109  
4.3 Computing Nash equilibria of \( n \)-player, general-sum games 110  
4.4 Computing maxmin and minmax strategies for two-player, general-sum games 113  
4.5 Identifying dominated strategies 114  
4.5.1 Domination by a pure strategy 115  
4.5.2 Domination by a mixed strategy 116  
4.5.3 Iterated dominance 118  
4.6 Computing correlated equilibria of \( n \)-player normal-form games 119  
4.7 History and references 121  

5 Games with Sequential Actions: Reasoning and Computing with the Extensive Form 123  
5.1 Perfect-information extensive-form games 123  
5.1.1 Definition 124  
5.1.2 Strategies and equilibria 125  
5.1.3 Subgame-perfect equilibrium 127  
5.1.4 Computing equilibria: backward induction 130  
5.2 Imperfect-information extensive-form games 136  
5.2.1 Definition 136  

© Shoham and Leyton-Brown, 2008
### Contents

7.4.4 Belief-based reinforcement learning 227  
7.5 No-regret learning and universal consistency 227  
7.6 Targeted learning 228  
7.7 Evolutionary learning and other large-population models 230  
7.7.1 The replicator dynamic 231  
7.7.2 Evolutionarily stable strategies 235  
7.7.3 Agent-based simulation and emergent conventions 237  
7.8 History and references 240  

8 **Communication** 243  
8.1 “Doing by talking” I: cheap talk 243  
8.2 “Talking by doing”: signaling games 247  
8.3 “Doing by talking” II: speech-act theory 250  
8.3.1 Speech acts 250  
8.3.2 Rules of conversation 251  
8.3.3 A game-theoretic view of speech acts 253  
8.3.4 Applications 256  
8.4 History and references 260  

9 **Aggregating Preferences: Social Choice** 263  
9.1 Introduction 263  
9.1.1 Example: plurality voting 263  
9.2 A formal model 264  
9.3 Voting 266  
9.3.1 Voting methods 266  
9.3.2 Voting paradoxes 268  
9.4 Existence of social functions 270  
9.4.1 Social welfare functions 270  
9.4.2 Social choice functions 274  
9.5 Ranking systems 277  
9.6 History and references 281  

10 **Protocols for Strategic Agents: Mechanism Design** 283  
10.1 Introduction 283  
10.1.1 Example: strategic voting 283  
10.1.2 Example: buying a shortest path 284  
10.2 Mechanism design with unrestricted preferences 285  
10.2.1 Implementation 286  
10.2.2 The revelation principle 288  
10.2.3 Impossibility of general, dominant-strategy implementation 290  
10.3 Quasilinear preferences 290  
10.3.1 Risk attitudes 291  
10.3.2 Mechanism design in the quasilinear setting 294  
10.4 Efficient mechanisms 298  
10.4.1 Groves mechanisms 299  

© Shoham and Leyton-Brown, 2008
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.4.2</td>
<td>The VCG mechanism</td>
<td>303</td>
</tr>
<tr>
<td>10.4.3</td>
<td>VCG and individual rationality</td>
<td>305</td>
</tr>
<tr>
<td>10.4.4</td>
<td>VCG and weak budget balance</td>
<td>307</td>
</tr>
<tr>
<td>10.4.5</td>
<td>Drawbacks of VCG</td>
<td>308</td>
</tr>
<tr>
<td>10.4.6</td>
<td>Budget balance and efficiency</td>
<td>312</td>
</tr>
<tr>
<td>10.4.7</td>
<td>The AGV mechanism</td>
<td>312</td>
</tr>
<tr>
<td>10.5</td>
<td>Beyond efficiency</td>
<td>313</td>
</tr>
<tr>
<td>10.5.1</td>
<td>What else can be implemented in dominant strategies?</td>
<td>314</td>
</tr>
<tr>
<td>10.5.2</td>
<td>Tractable Groves mechanisms</td>
<td>316</td>
</tr>
<tr>
<td>10.6</td>
<td>Computational applications of mechanism design</td>
<td>318</td>
</tr>
<tr>
<td>10.6.1</td>
<td>Task scheduling</td>
<td>318</td>
</tr>
<tr>
<td>10.6.2</td>
<td>Bandwidth allocation in computer networks</td>
<td>321</td>
</tr>
<tr>
<td>10.6.3</td>
<td>Multicast cost sharing</td>
<td>323</td>
</tr>
<tr>
<td>10.6.4</td>
<td>Two-sided matching</td>
<td>327</td>
</tr>
<tr>
<td>10.7</td>
<td>Constrained mechanism design</td>
<td>332</td>
</tr>
<tr>
<td>10.7.1</td>
<td>Contracts</td>
<td>333</td>
</tr>
<tr>
<td>10.7.2</td>
<td>Bribes</td>
<td>335</td>
</tr>
<tr>
<td>10.7.3</td>
<td>Mediators</td>
<td>336</td>
</tr>
<tr>
<td>10.8</td>
<td>History and references</td>
<td>337</td>
</tr>
</tbody>
</table>

11 **Protocols for Multiagent Resource Allocation: Auctions** | 341 |
| 11.1    | Single-good auctions | 341  |
| 11.1.1  | Canonical auction families | 342  |
| 11.1.2  | Auctions as Bayesian mechanisms | 344  |
| 11.1.3  | Second-price, Japanese, and English auctions | 345  |
| 11.1.4  | First-price and Dutch auctions | 347  |
| 11.1.5  | Revenue equivalence | 350  |
| 11.1.6  | Risk attitudes | 352  |
| 11.1.7  | Auction variations | 354  |
| 11.1.8  | “Optimal” (revenue-maximizing) auctions | 356  |
| 11.1.9  | Collusion | 357  |
| 11.1.10 | Interdependent values | 361  |
| 11.2    | Multiunit auctions | 364  |
| 11.2.1  | Canonical auction families | 364  |
| 11.2.2  | Single-unit demand | 365  |
| 11.2.3  | Beyond single-unit demand | 368  |
| 11.2.4  | Unlimited supply: random sampling auctions | 370  |
| 11.2.5  | Position auctions | 372  |
| 11.3    | Combinatorial auctions | 375  |
| 11.3.1  | Simple combinatorial auction mechanisms | 377  |
| 11.3.2  | The winner determination problem | 378  |
| 11.3.3  | Expressing a bid: bidding languages | 381  |
| 11.3.4  | Iterative mechanisms | 386  |
| 11.3.5  | A tractable mechanism | 389  |

*Multiagent Systems, draft of March 28, 2008*
11.4 Exchanges 390
11.4.1 Two-sided auctions 391
11.4.2 Prediction markets 392
11.5 History and references 393

12 Teams of Selfish Agents: An Introduction to Coalitional Game Theory 397
12.1 Coalitional games with transferable utility 397
12.1.1 Definition 398
12.1.2 Examples 398
12.1.3 Classes of coalitional games 400
12.2 Analyzing coalitional games 401
12.2.1 The Shapley value 402
12.2.2 The core 405
12.2.3 Refining the core: e-core, least core, and nucleolus 408
12.3 Compact representations of coalitional games 411
12.3.1 Weighted majority games and weighted voting games 411
12.3.2 Weighted graph games 412
12.3.3 Capturing synergies: a representation for superadditive games 414
12.3.4 A decomposition approach: multi-issue representation 415
12.3.5 A logical approach: marginal contribution nets 416
12.4 Further directions 418
12.4.1 Alternative coalitional game models 418
12.4.2 Advanced solution concepts 420
12.5 History and references 421

13 Logics of Knowledge and Belief 423
13.1 The partition model of knowledge 423
13.1.1 Muddy children and warring generals 423
13.1.2 Formalizing intuitions about the partition model 424
13.2 A detour to modal logic 427
13.2.1 Syntax 428
13.2.2 Semantics 429
13.2.3 Axiomatics 429
13.2.4 Modal logics with multiple modal operators 430
13.2.5 Remarks about first-order modal logic 430
13.3 S5: An axiomatic theory of the partition model 431
13.4 Common knowledge, and an application to distributed systems 434
13.5 Doing time and an application to robotics 438
13.5.1 Termination conditions for motion planning 438
13.5.2 Coordinating robots 442
13.6 From knowledge to belief 444
13.7 Combining knowledge and belief (and revisiting knowledge) 445
13.8 History and references 451

14 Beyond Belief: Probability, Dynamics and Intention 453

© Shoham and Leyton-Brown, 2008