Game Theory and Decision Theory in Multi-Agent Systems

Simon Parsons (s.d.parsons@csc.liv.ac.uk)
Department of Computer Science, University of Liverpool,
Liverpool L69 7ZF, United Kingdom

Michael Wooldridge (m.j.wooldridge@csc.liv.ac.uk)
Department of Computer Science, University of Liverpool,
Liverpool L69 7ZF, United Kingdom

Abstract. In the last few years, there has been increasing interest from the agent community in the use of techniques from decision theory and game theory. Our aims in this article are firstly to briefly summarise the key concepts of decision theory and game theory, secondly to discuss how these tools are being applied in agent systems research, and finally to introduce this special issue of *Autonomous Agents and Multi-Agent Systems* by reviewing the papers that appear.

Keywords: game theory, decision theory.

1. Introduction

In the last few years, there has been increasing interest in the use of techniques from decision theory and game theory for analysing and implementing agent systems. Our aims in this article are firstly to briefly summarise the key concepts of decision theory and game theory, secondly to discuss how these tools are being applied in agent systems research, and finally to introduce this special issue of *Autonomous Agents and Multi-Agent Systems* by briefly discussing the papers that appear.

Broadly speaking, decision theory [20] is a means of analysing which of a series of options should be taken when it is uncertain exactly what the result of taking the option will be. Decision theory concentrates on identifying the "best" decision option, where the notion of "best" is allowed to have a number of different meanings, of which the most common is that which maximises the expected utility of the decision maker. Decision theory provides a powerful tool with which to analyse scenarios in which an agent must make decisions in an unpredictable environment.

Game theory [1] is a close relative of decision theory, which studies interactions between self-interested agents. In particular, it studies the problems of how interaction *strategies* can be designed that will maximise the welfare of an agent in a multi-agent encounter, and how *protocols* or *mechanisms* can be designed that have certain desirable properties. Notice that decision theory can be considered to be the study of *games against nature*, where nature is an opponent that does not seek to gain the best payout, but rather acts randomly. Given this brief description, it comes as no surprise to learn that many of the applications of game theory in agent systems have been to analyse multi-agent interactions, particularly those involving negotiation and co-ordination.

This increasing interest in game theory and decision theory in the agents community led us to believe that the time was ripe to hold a workshop which focused on these matters. This workshop was held in London on the 3rd July 1999, in conjunction with the Fifth European Conference on Quantitative and Symbolic Approaches to Reasoning and Uncertainty [9]. The special issue of *Autonomous Agent and Multi-Agent Systems* you are now reading includes revised and polished versions of the best papers that were presented at this workshop. Our aims in this article are to provide very brief introductions to the areas of decision theory and game theory in general, then to put the papers that appear in this special issue into some context, and finally to point to further reading on this exciting and rapidly expanding subject.

2. Decision Theory in Agent Systems

Classical decision theory, so called to distinguish it from a number of non-classical theories which have grown up in the last few years, is a set of mathematical techniques for making decisions about what action to take when the outcomes of the various actions are not known. Although the area grew up long before the concept of an intelligent agent was conceived, such agents are canonical examples of the decision makers which can usefully employ classical decision theory.

An agent operating in a complex environment is inherently uncertain about that environment; it simply does not have enough information about the enviroment to know either the precise current state of its environments, nor how that environment will evolve. Thus, for every variable X_i which captures some aspect of the current state of the environment, all the agent typically knows is that each possible value x_{i_j} of each X_i has some probability $\Pr(x_{i_j})$ of being the current value of X_i. Writing $\mathbf{x}$ for the set of all x_{i_j}, we have:

$$\Pr : x \in \mathbf{x} \mapsto [0, 1]$$

and

$$\sum_j \Pr(x_{i_j}) = 1$$

In other words, the probability $\Pr(x_{i_j})$ is a number between 0 and 1 and the sum of the probabilities of all the possible values of X_i is 1. If X_i is known to have value x_{i_j} then $\Pr(x_{i_j}) = 1$ and if it is known not to have value x_{i_j} then $\Pr(x_{i_j}) = 0$.

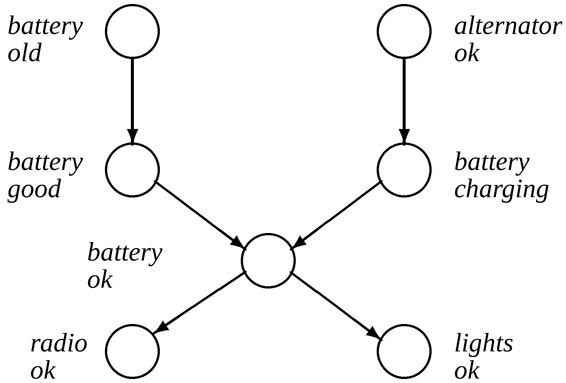

Figure 1. An example Bayesian network

Given two of these variables, X_1 and X_2, then the probabilities of the various values of X_1 and X_2 may be related to one another. If they are not related, a case we distinguish be referring to X_1 and X_2 as being *independent*, then for any two values x_{1_i} and x_{2_j}, we have:

$$\Pr(x_{1_i} \wedge x_{2_j}) = \Pr(x_{1_i}) \Pr(x_{2_j})$$

If the variables are not independent, then:

$$\Pr(x_{1_i} \wedge x_{2_j}) = \Pr(x_{1_i}|x_{2_j}) \Pr(x_{2_j})$$

where $\Pr(x_{1_i}|x_{2_j})$ is the probability of X_1 having value x_{1_i} given that X_2 is known to take value x_{2_j}. Such *conditional probabilities* capture the relationship between X_1 and X_2, representing, for instance, the fact that x_{1_i} (the value “wet”, say, of the variable “state of clothes”) becomes much more likely when x_{2_j} (the value “raining” of the variable “weather condition”) is known to be true.

If we take the set of these X_i of which the agent is aware, the set $\mathbf{X}$, then for each pair of variables in $\mathbf{X}$ we can establish whether the pair are independent or not. We can then build up a graph in which each node corresponds to a variable in $\mathbf{X}$ and an arc joins two nodes if the variables represented by those nodes are not independent of each other. The resulting graph is known as a Bayesian network[1] [18], and the graphical structure provides a convenient computational framework in which to calculate the probabilities of interest to the agent. In general, the agent will have some set of variables whose values it can observe, and once these observations have been taken, will want to calculate the probabilities of the various values of some other set of variables.

[1] The notion of independence captured in the arcs of a Bayesian network is somewhat more complex than that described here, but the difference is not relevant for the purposes of this article. For full details, see [18].

Figure 1 is an example of a fragment of a Bayesian network for diagnosing faults in cars. It represents the fact that the age of the battery (represented by the node $battery\ old$ has a probabilistic influence on how good the battery is, and that this in turn has an influence on whether the battery is operational ($battery\ ok$), the latter being affected also by whether the alternator is working and, as a result, whether the battery is recharged when the car moves. The operational state of the battery affects whether the radio and lights will work. In this network it is expected that the observations that can be carried out are those relating to the lights and the radio (and possibly the age of the battery), and that the result of these observations can be propagated through the network to establish the probability of the alternator being okay and the battery being good. In this case these latter variables are the ones which we are interested in since they relate to fixing the car.

Typically the variables an agent will be interested in are those that relate to its goals. For instance, the agent may be interested in choosing an action that will allow it to achieve a goal, and might therefore be interested in choosing that action which has the greatest chance of succeeding in achieving that goal. When the agent has many goals it could achieve, this strategy could be extended to make the agent choose to achieve the goal which has the greatest chance of being achieved, and to do this by applying the action which gives this greatest chance.

However, building an agent which follows this strategy is somewhat short-sighted since the agent will not consider the value of the goals, and will therefore choose a goal which is easy to achieve, but worthless, over a goal which is hard to achieve but very valuable. To take account of this problem, decision theory makes use of the idea of *utility*. A utility is a value which is associated with a state of the world, and which represents the value that the agent places on that state of the world. Utilities provide a convenient means of encoding the preferences of an agent; as von Neumann and Morgenstern [15] showed, it is possible to define utility functions that faithfully encode preferences such that a state S_i is preferred to S_j, if and only if it has a higher utility for the agent.

Now, we can consider that our agent has a set of possible actions $\mathbf{A}$, each member A_i of which has a range of possible outcomes since the actions are not deterministic. The value of taking a particular action will depend upon what the state of the world is—it is of little value carrying a surfboard when taking a trip across the Sahara—and so in choosing which action to undertake, our agent will need to look at the value of $U(S_j)$ where S_j is the state it is in after the action. Doing this for each possible action, the agent can then choose the action which leads to the state it values most. We can certainly build an agent which works in this way, and it would unerringly chose to achieve the goal with the highest value as encoded by its utility function. However it would be just as flawed as the agent which only tried to achieve

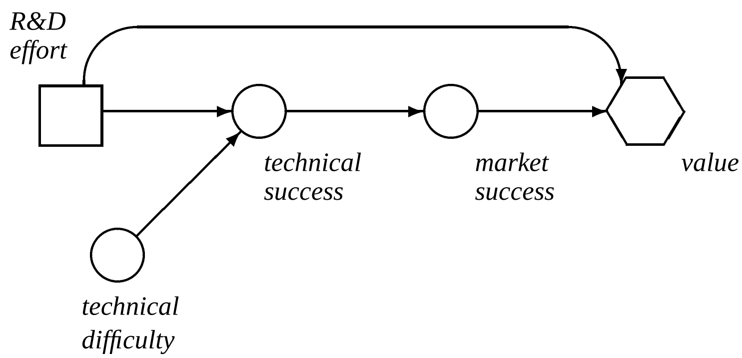

Figure 2. An example influence diagram

the most likely goal, trying to achieve the most valuable goal irrespective of the difficulty of that goal.

To build more sensible agents we combine probability and utility calculations for each action and calculate the *expected utility* of each. This amounts to calculating a weighted average of the utility of each outcome, where the weight is the probability of that outcome given the action being performed. Since each outcome is itself a state, we have:

$$EU(A_i) = \sum_{S_j \in \mathbf{S}} \Pr(S_j|A_i)U(S_j)$$

where $\mathbf{S}$ is the set of all states. The agent then selects action A^* where:

$$A^* = arg \max_{A_i \in \mathbf{A}} \sum_{S_j \in \mathbf{S}} \Pr(S_j|A_i)U(S_j)$$

Now, these states which are being considered here are just particular instantiatons of the set of state variables $\mathbf{X}$. Thus the probabilities in this calculation are just the probabilities of the X_i having particular values given the actions.

Harking back to the discussion of Bayesian networks above, we can think of the X_i as being structured as a graph, dropping the distinction between variables and the nodes in the graph which represent them. The A_i can be brought into the graph as well, as a different kind of node (square, perhaps, in contrast to the usual round ones relating to the X_i) linked to the X_i whose values they influence. We can also incorporate utilities. This time we only require a single node (a hexagon, to keep it distinct from the others), and this is linked to those X_i which affect its value. Such a graphical structure neatly captures all the dependencies in an expected utility calculation, and is known as an influence diagram [8].

Figure 2 is an example of a small influence diagram capturing a decision problem which a company has to make about its research and development budget. Since the budget is the thing the decision is being made about, it is represented by a square *decision node*. This is linked to the factors it directly effects, namely the technical success of the comapny's products and their overall profitability, that latter being captured by the hexagonal *value node*. The remaining nodes are *chance nodes* and represent the other factors which relate to the decision. These are just like nodes in a Bayesian network. Given a particular instantiation of the decision node, the relevant values can be propagated through the network, using an algorithm such as Shacter's graph reduction algorithm [27] to establish the expected utility of the decision.

Given that the basic mechanisms of decision theory fit so neatly into the context of intelligent agents, it is perhaps surprising that they have not been more widely employed in the field. However, agent systems which use decision theory seriously (that is adopting the notions of probability and utility) are rather scarce. One sub-area of decision theory is, however, becoming popular and that is the field of Markov decision processes (MDPs), discussed in detail in [2]. In essence an MDP is an iterative set of classical decision problems. Consider a state of the world as a node in a graph. Carrying out an action in that state will result in a transition to one of a number of states, each connected to the first state by an arc, with some probability, and incur some cost. After a series of transitions a goal state may be reached, and the sequence of actions executed to do this is known as a *policy*. Solving an MDP amounts to finding a minimal cost policy for moving from some initial state to a goal state.

MDPs capture many of the facets of real world problems, but unrealistically assume that whatever system is solving the MDP knows at every point what state it is in. This amounts to assuming that it is possible to measure some aspect of the world and from this measurement tell precisely what state the world is in. This is rarely the case; it is far more likely is that from the measurement something can be uncertainly inferred about the world. In such a situation, the states of an MDP are replaced by beliefs about those states, and we have a partially observable Markov decision process (POMDP). Because they can capture so many real situations, POMDPs are currently a hot topic in agent research, despite the fact that they are intractable for all but the smallest problems.

3. Game Theory in Multi-Agent Systems

Game theory is a branch of economics that studies interactions between self-interested agents. Like decision theory, with which it shares many concepts, game theory has its roots in the work of von Neumann and Morgenstern [15]. As its name suggests, the basic concepts of game theory arose from the study of games such as chess and checkers. However, it rapidly became clear that the techniques and results of game theory can equally be applied to *all* interactions that occur between self-interested agents.

The classic game theoretic question asked of any particular multi-agent encounter is: What is the best — most rational — thing an agent can do? In most multi-agent encounters, the overall outcome will depend critically on the choices made by all agents in the scenario. This implies that in order for an agent to make the choice that optimises its outcome, it must reason *strategically*. That is, it must take into account the decisions that other agent may make, and must assume that they will act so as to optimise their own outcome. Game theory gives us a way of formalising and analysing such concerns.

In the early days of multi-agent systems research, it was widely assumed that agents were *benevolent*: put simply, that agents could be assumed to share a common goal, and would therefore be happy to "help out" whenever asked. The focus was on *distributed problem solving* systems, in which groups of benevolent agents worked together to solve problems of common interest [21, p3]. There seemed to be an implicit assumption that this class of systems was the most common, and that scenarios in which agents are in competition were unusual at best, aberrations at worst. Over time, however, it has come to be recognised that in fact, benevolence is the exception; self-interest is the norm. The recognition of this fact appears to have been driven, at least in part, by the rapid growth of the Internet and the continuing trend towards ever more distributed systems in computer science generally.

In tandem with this increasing recognition that self-interested agents are the norm has been a steady growth of interest in the applications of game theory to multi-agent systems. Game theory entered the multi-agent systems literature largely through the work of Jeffrey Rosenschein and colleagues (see, e.g., [22, 23, 24]). In his 1985 PhD thesis [21], Rosenschein used game theoretic techniques to analyse a range of multi-agent interaction scenarios. For example, he showed how certain types of cooperation and deal making could take place without communication: both agents simply compute the best outcome and know that the party they are dealing with will do the same. Since agents can use game theoretic techniques to predict what others will do, this obviates the need for explicit communication — coordination arises because of the assumption of mutual rationality [23].

Perhaps the most compelling applications of game theory to multi-agent systems have been in the area of negotiation [12, 24, 26]. Put simply, negotiation is the process by which agents can reach agreement on matters of common interest. Negotiation and bargaining were studied in the game theory literature well before the emergence of multi-agent systems as a research discipline, and even before the advent of the first digital computer. However, computer science brings two important considerations to the game theoretic study of negotiation and bargaining:

1. Game theoretic studies of rational choice in multi-agent encounters typically assumed that agents were allowed to select the best strategy from the space of all possible strategies, by considering all possible interactions. It turns out that the "search space" of strategies and interactions that needs to be considered has exponential growth, which means that the problem of finding an optimal strategy is in general computationally intractable. In computer science, the study of such problems is the domain of computational complexity theory [17]. There is a significant literature devoted to the development of efficient (polynomial time) algorithms for apparently intractable problems, and the application of such techniques to the study of multi-agent encounters is a fruitful ongoing area of work.

2. The emergence of the Internet and World-Wide Web has provided an enormous commercial imperative to the further development of computational negotiation and bargaining techniques [16].

Given a particular negotiation scenario that will involve automated agents, game theoretic techniques can be applied to two key problems:

1. The design of an appropriate *protocol* that will govern the interactions between negotiation participants. The protocol defines the "rules of encounter" between agents [24]. Formally, a protocol can be understood as a function that, on the basis of prior negotiation history, defines what proposals are allowable by negotiation participants. It is possible to design protocols so that any particular negotiation history has certain desirable properties — this is *mechanism design*, and is discussed in more detail below.

2. The design of a particular *strategy* that individual agents can use while negotiating — an agent will aim to use a strategy that maximises its own individual welfare. A key difficulty here is that typically, the strategies that work best in theory tend to be computationally intractable, and are hence unusable by agents in practice.

As noted above, mechanism design involves the design of protocols for governing multi-agent interactions, such that these protocols have certain desirable properties. Possible properties include, for example [26, p204]:

– *Guaranteed success*.

  A protocol guarantees success if it ensures that, eventually, agreement is certain to be reached.

– *Maximising social welfare*.

  Intuitively, a protocol maximises social welfare if it ensures that any outcome maximises the sum of the utilities of negotiation participants. If the utility of an outcome for an agent was simply defined in terms of the amount of money that agent received in the outcome, then a protocol that maximised social welfare would maximise the *total* amount of money "paid out".

– *Pareto efficiency*.

  A negotiation outcome is said to be Pareto efficient if there is no other outcome that will make at least one agent better off without making at least one other agent worse off. Intuitively, if a negotiation outcome is not Pareto efficient, then there is another outcome that will make at least one agent happier while keeping everyone else at least as happy.

– *Individual rationality*.

  A protocol is said to be individually rational if following the protocol — "playing by the rules" — is in the best interests of negotiation participants. Individually rational protocols are essential because without them, there is no incentive for agents to engage in negotiations.

– *Stability*.

  A protocol is *stable* if it provides all agents with an incentive to behave in a particular way. The best-known kind of stability is *Nash equilibrium*: two strategies s and s' are said to be in Nash equilibrium if under the assumption that one agent is using s, the other can do no better than use s', and vice versa.

– *Simplicity*.

  A "simple" protocol is one that makes the appropriate strategy for a negotiation participant "obvious". That is, a protocol is simple if using it, a participant can easily (tractably) determine the optimal strategy.

– *Distribution*.

  A protocol should ideally be designed to ensure that there is no "single point of failure" (such as a single arbitrator), and ideally, so as to minimise communication between agents.

The fact that even quite simple negotiation protocols can be proven to have such desirable properties as these accounts in no small part for the success of game theoretic techniques for negotiation [12].

Despite these very obvious advantages, there are a number of problems associated with the use of game theory when applied to negotiation problems:

– Game theory assumes that it is possible to characterise an agent's preferences with respect to possible outcomes. *Humans*, however, find it extremely hard to consistently define their preferences over outcomes — in general, human preferences cannot be characterised even by a simple ordering over outcomes, let alone by numeric utilities [25, pp475–480]. In scenarios where preferences are obvious (such as the case of a person buying a particular CD and attempting to minimise costs), game theoretic techniques may work well. With more complex (multi-issue) preferences, it is much harder to use them.

– Most game theoretic negotiation techniques tend to assume the availability of unlimited computational resources to find an optimal solution — they have the characteristics of NP-hard problems. (A well known example is the problem of winner determination in combinatorial auctions.) In such cases, *approximations* of game theoretic solutions may be more appropriate.

Despite these problems, game theory is extremely compelling as a tool for automated negotiation. In cases where it is possible to characterise the preferences and possible strategies of negotiation participants, then game theory has much to offer.

4. The Papers

As discussed by Guttman *et al.* [7] and Crabtree [5], one of the niches in which autonomous agents are rapidly proving their worth is electronic commerce. Here agents help to “grease the wheels” that must turn in order that goods and services can be bought and sold across the Internet. One class of these wheel-greasing agents are shopbots, agents which search the Internet on behalf of consumers, comparing prices across dozens of web sites. Shopbots thus help to cut consumers’ costs, not just in the sense of allowing them to find the cheapest source of the good they require, but also in the more general sense of reducing the cost of obtaining optimal price and quality. Shopbots can also help to reduce the costs of suppliers, by reducing the cost of evaluating, updating and advertising prices, and thus have the potential to significantly affect the way markets operate. As a result, Kephart and Greenwald [11] have investigated the impact of shopbots in single commodity markets, modelling the behaviour of both buyers and sellers using game theoretic techniques, and the paper in this issue presents a summary of their results.

Shopbots can be considered as operating on the side of buyers in an electronic market helping them to find the best deals. The complementary kind of agent which operates on the side of sellers are what have been called “pricebots”, autonomous agents which fix the prices charged by a seller in order to secure the best price that sellers are prepared to pay. Tesauro and Kephart

[30] have investigated how such price-setting agents can be made adaptive, in particular how they can make use of Q-learning [32], an approach which factors in the long-term expected reward for a given action taken in a given state. They consider a number of different model economies, including one in which buyers are assumed to make use of shopbots, and found that the Q-learning approach leads, broadly speaking, to increased profits for sellers, in part because it reduces the effect of price wars (when sellers repreatedly undercut each other in an attenpt to capture a bigger share of the market).

In applications of game theory in multi-agent systems, agents are usually taken to maximise their expected utility. This approach, however, is not always practical—there are often bounds on computational resources which prevent the optimal solution being computed. As a result, there has been much interest in computing solutions under bounded rationality, that is approaches which aim to be rational in the sense of computing the solution with maximum expected utility, but which acknowledge bounds on their resources, and so relax one or more assumptions of the optimal approach. Stirling, Goodrich and Packard [29], consider an adaptation of Simon's idea of satisficing—that is searching for an optimal solution until the cost of continuing the search outweighs the improvement in solution that further work will bring. They do this by introducing the notion of praxeic utility, a measure which explicitly models the resources consumed, and allows these to be balanced against the desire to obtain the best solution.

The approach adopted by Stirling, Goodrich and Packard, while departing to some degree from classical decision theory, still makes use of the same basic mechanisms. In particular, the approach takes praxeic utilities as primitive although, as with utilities in classical decision theory, these will ultimately be grounded in some kind of preference order. In contrast, Lang, van der Torre and Weydert [13] start with an agent's goals, and consider how an agent might reason about its goals and use these to define its utility function. Their approach is logic-based, and thus an extension of recent work on qualitative decision theory [6].

The two papers just described deal with combinations of beliefs and utilities. In the case of Stirling, Goodrich and Packard, these beliefs are distributed over states of affairs (roughly speaking conjunctions of propositions), while in the work of Lang, van der Torre and Weydert, the beliefs are taken over individual propositions. Thus, in neither case, is there much structure to the items that beliefs are distributed over. Vane and Lehner [31], on the other hand, build belief distributions over much more complex objects—in fact they deal with beliefs over games, in the sense of game theory. In essence, their hypergame framework allows a agent in a game theoretic setting to hedge its bets about what its opponent is doing. It does this by identifying a set of possible games, representing the possible behaviours that its opponent might engage in, building a probability distribution over these games, and evaluating the best moves by the usual maximum expected utility algorithm. The result is an elegant formalism which is a strict generalisation of both game theory and decision theory, and which looks to be a useful tool for building future generations of mixed game theoretic and decision theoretic agents.

5. Further Reading

In this final section, we provide some pointers to further reading on the use of decision theory and game theory in the multi-agent systems literature.

As befits a subject which has been established for some time, there are a number of good textbooks on decision theory. Of these, perhaps the best are those by Lindley [14], Raiffa [20], and Smith [28]. The books by Lindley and Smith are both intended for a statistics audience, while Raiffa's intended audience is more a economics or business one, but all are written carefully enough to make them easy for computer scientists to understand. As mentioned above, the main area in which ideas from decision theory have been carried into artificial intelligence is that of Bayesian networks and influence diagrams. The standard reference on Bayesian networks remains Pearl's landmark volume [18], though Jensen's more recent contribution [10] provides a clearer introduction, and both Cowell *et al.* [4] and Castillo *et al.* [3] cover more ground. Sadly none of these authors cover influence diagrams in any detail, and the main reference for graphical decision models remains [8]. On the more specialised topic of Markov decision processes, there is at least one good monograph [19], and a number of articles which give much of the necessary detail. Of the latter, the paper by Boutilier, Dean and Hanks [2] is perhaps the most useful.

The game theory literature has grown steadily since the origins of the field — there are now a clutch of journals on the subject, and many textbooks. Unfortunately, for a non-economics audience, many of these textbooks are hard going. Among the more helpful is Binmore's *Fun and Games* [1] (which also includes a good summary of further reading and an amusing critique of other textbooks in the field). With respect to the multi-agent systems literature, most applications of game theory have been in the area of negotiation, and the starting point should undoubtedly be Rosenschein and Zlotkin's [24]. Kraus provides a summary of work on negotiation as of 1997 [12], and Sandholm's authoritative [26] provides an excellent summary of game theory for multi-agent interactions.

References

1. K. Binmore. *Fun and Games: A Text on Game Theory*. D. C. Heath and Company: Lexington, MA, 1992.

2. C. Boutilier, T. Dean, and S. Hanks. Decision-theoretic planning: structural assumptions and computational leverage. *Journal of Artificial Intelligence Research*, 11:1–94, 1999.
3. E. Castillo, J. M. Gutiérrez, and A. S. Hadi. *Expert Systems and Probabilistic Network Models*. Springer Verlag, Berlin, Germany, 1997.
4. R. G. Cowell, A. P. Dawid, S. L. Lauritzen, and D. J. Spiegelhalter. *Probabilistic Networks and Expert Systems*. Springer Verlag, Berlin, Germany, 1999.
5. B. Crabtree. What chance software agents? *The Knowledge Engineering Review*, 13(2):131–136, 1998.
6. J. Doyle and R. H. Thomason. Background to qualitative decision theory. *AI Magazine*, 20(2):55–68, 1999.
7. R. Guttman, A. G. Moukas, and P. Maes. Agent-mediated electronic commerce. *The Knowledge Engineering Review*, 13(2):147–159, 1998.
8. R. A. Howard and J. E. Matheson. Influence diagrams. In R. A. Howard and J. E. Matheson, editors, *Readings on the Principles and Applications of Decision Analysis*, pages 719–762. Strategic Decisions Group, Menlo Park, CA, 1984.
9. A. Hunter and S. Parsons, editors. *Symbolic and Quantitative Approaches to Reasoning and Uncertainty*. Springer Verlag, Berlin, Germany, 1999.
10. F. V. Jensen. *An Introduction to Bayesian Networks*. UCL Press, London, 1996.
11. J. O. Kephart and A. R. Greenwald. Shopbot economics. *International Journal of Autonomous Agents and Multi-Agent Systems*, (this issue), 2000.
12. S. Kraus. Negotiation and cooperation in multi-agent environments. *Artificial Intelligence*, 94(1-2):79–98, July 1997.
13. J. Lang, L van der Torre, and E. Weydert. Utilitarian desires. *International Journal of Autonomous Agents and Multi-Agent Systems*, (this issue), 2000.
14. D. V. Lindley. *Making Decisions*. John Wiley & Sons, Chichester, UK, 1975.
15. J. Von Neumann and O. Morgenstern. *Theory of Games and Economic Behaviour*. Princeton University Press, 1944.
16. P. Noriega and C. Sierra, editors. *Agent Mediated Electronic Commerce (LNAI Volume 1571)*. Springer-Verlag: Berlin, Germany, 1999.
17. C. H. Papadimitriou. *Computational Complexity*. Addison-Wesley: Reading, MA, 1994.
18. J. Pearl. *Probabilistic Reasoning in Intelligent Systems; Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA., 1988.
19. M. L. Puterman. *Markov Decision Processes*. John Wiley & Sons, New York, NY, 1994.
20. H. Raiffa. *Decision Analysis: Introductory Lectures on Choices under Uncertainty*. Addison Wesley, Reading, MA., 1968.
21. J. S. Rosenschein. *Rational Interaction: Cooperation Among Intelligent Agents*. PhD thesis, Computer Science Department, Stanford University, Stanford, CA 94305, 1985.
22. J. S. Rosenschein and M. R. Genesereth. Deals among rational agents. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence (IJCAI-85)*, pages 91–99, Los Angeles, CA, 1985.
23. J. S. Rosenschein, M. Ginsberg, and M. R. Genesereth. Cooperation without communication. In *Proceedings of the Fifth National Conference on Artificial Intelligence (AAAI-86)*, Philadelphia, PA, 1986.
24. J. S. Rosenschein and G. Zlotkin. *Rules of Encounter: Designing Conventions for Automated Negotiation among Computers*. The MIT Press: Cambridge, MA, 1994.
25. S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, 1995.
26. T. Sandholm. Distributed rational decision making. In G. Weiss, editor, *Multiagent Systems*, pages 201–258. The MIT Press: Cambridge, MA, 1999.
27. R. Shachter. Evaluating influence diagrams. *Operations Research*, 34:871–882, 1986.
28. J. Q. Smith. *Decision Analysis: A Bayesian Approach*. Springer Verlag, Berlin, 1999.

29. W. C. Stirling, M. A. Goodrich, and D. J. Packard. Satisficing equilibria: a non-classical theory of games and decisions. *International Journal of Autonomous Agents and Multi-Agent Systems*, (this issue), 2000.
30. G. Tesauro and J. O. Kephart. Pricing in agent economies using multi-agent q-learning. *International Journal of Autonomous Agents and Multi-Agent Systems*, (this issue), 2000.
31. R. Vane and P. Lehner. Using hypergames to increase planned payoff and reduce risk. *International Journal of Autonomous Agents and Multi-Agent Systems*, (this issue), 2000.
32. C. J. C. H. Watkins. *Learning from delayed rewards*. PhD thesis, Cambridge University, 1989.