A Constraint-Based Approach to Multi-Agent Planning

Nick Lacey, Henry Hexmoor
University of Arkansas
313 Engineering Hall
Fayetteville, Arkansas
{nlacey,hexmoor}@uark.edu

Abstract

Agents which exist in a multi-agent environment and wish to cooperate must individually form plans which incorporate the actions of their fellow agents. In this paper we present an agent architecture which permits high-level adaptive plans to be formed and revised, based on perceptions of the actions of other agents in the environment. This is achieved through a belief revision system based on constraint satisfaction in which high level beliefs concerning the relationships between values, norms, and intentions are represented and manipulated.

Introduction

One of the potential advantages of multi-agent systems is the ability to allow autonomous intentional agents to divide up a task between them. In order to allow such division of labour, each individual agent would have to possess at least the following abilities, as identified by Jennings, Sycara, and Wooldridge (Jennings, Sycara, & Wooldridge 1998):

- *Responsiveness* The agent would have to be able to perceive its environment, process its perceptions, and respond to them.
- *Pro-Activity* The agent would have to be able to exhibit goal-directed behaviour, and be able to achieve long-term objectives
- *Social Awareness* The agent has to be aware of the other agents in its environment, and of their individual abilities and limitations

There is, however, considerable difficulty in combining pro-activity with social awareness. This is because it is not enough for each individual agent to construct plans based on its own abilities. The fact that we are expecting our agents to cooperate means that agents have to form plans which concern the abilities and actions of *other agents*, as well as their own.

In order to address this issue we propose a framework for agent design which combines the following three components:

- Coherence-Based Constraint Satisfaction
- Intention-Level Planning
- The Representation of Social Notions

This paper is organised as follows. We begin by describing a coherence-based belief revision systems which has already been designed and implemented. We then describe an extension to this system which would permit an agent to reason pro-actively, as well as responsively.

Once this groundwork is in place, we describe how the representation of social notions, combined with intention-level planning and constraint satisfaction, can produce agents which are capable of producing and revising plans in multi-agent environments.

Belief Revision Using Coherence-Based Constraint Satisfaction

The work presented in this paper puts forward an approach to the problem of planning in multi-agent environments based on social notions, intentions, and constraint satisfaction. Our approach is based on the *coherence* approach to belief revision (Doyle 1992). Thagard and Millgram (Thagard & Millgram 1997) suggest an interesting approach to decision making based on coherence. They define decision making as inference to the best plan, and suggest that when people make decisions they do so by adopting *"…complex plans on the basis of a holistic assessment of various competing actions and goals"*

Like Thagard and Millgram, we suggest that decision making involves complex planning based on the assessment of competing holistic interpretations. However, whereas Thagard and Millgram suggest a connectionist approach to modeling this process, we have used a symbolic approach which allows us to explicitly model and represent different classes of belief, such as actions, perceptions, explanations, and intentions. Thus, we follow Jonker and Treur and the DESIRE system in that we use a high-level conceptual description of the agent's domain and problem information (Jonker & Treur 1998).

The approach presented in this paper builds on existing work concerning the comparison of different approaches to belief revision. Several agents incorporating different belief revision strategies were designed and implemented. The agent we shall be considering here was designed to be strongly holistic, and as such was named agent **SH**. The following is a very brief overview of the principal characteristics of the agent.[1]

Explanations

The primary method of inference is that of *Explanation-Based Backward Chaining*. The ultimate goal of the backward chaining process is to find the *Final Interpretation* for the current perception data. The value of this may be derived from several alternative explanations, each one representing an alternative high-level conception of the current state of the agent's environment.

The first explanation that the agent will attempt to backward chain is the *Default Explanation*. This represents the most coherent explanation of all information received by the agent prior to the present moment. If the agent is able to find the value of the *Default Explanation* without violating any constraints, then it has no need to investigate any of the alternative explanations, and can adopt the default explanation.

Alternatively, the agent may find that no value for default explanation can be found without violating constraints. This indicates the agent's internal model is out of step with its environment, so some adjustment is required. If this occurs, as many alternative explanations as necessary are explored. The alternative explanations are derived and then holistically evaluated. This approach mirrors Thagard and Millgram's concept of the holistic assessment of competing hypotheses, and is based on philosophical approaches to truth and justification based on coherence.[2]

Explanations are *Meta-Level* beliefs, meaning that they concern the relationships between other beliefs, rather than directly representing states of affairs in the agent's environment. Every explanation may incorporate other, lower-level explanations, as well as domain-dependent beliefs concerning facts and relations about the agent's environment. Alternative explanations will usually provide differing mechanisms to derive the same pieces of data. This means that if a piece of sensory data that is necessary for the default explanation is unavailable, the agent may be able to use its knowledge of its environment to derive the missing data from another source.

The precise nature of the explanations used by a given agent will depend upon the environment within which the agent is to operate. Coherence can be used to provide a measure of the "best" interpretation. This is done through the concepts of *integrity* and *centrality*. The concepts of integrity and centrality are used to measure the value of an entire knowledge base, and an individual belief, respectively.

If the ontology of the agent were to be visualised as a sphere, the central beliefs would be the meta-beliefs while the outermost beliefs would be state dependent assertions.[3] By associating each belief with the ontological level to which it belongs, the agent has access to a computationally inexpensive method of deriving the centrality of each belief.

This technique also addresses a common criticism of coherence based justificational structures, namely that they are unable to account for the differing epistemological importance attached to difference classes of belief. The spherical model provides a clear basis for holding that high-level core beliefs are more important to the agent's ontology that perceptual beliefs on the periphery, and hence are less likely to be revised.

Constraints

Constraints are used to alert the agent to the presence of an inconsistency. Their role can be seen as three-fold:

1. Constraints can be viewed as a shortcut, as they allow the agent to detect an inconsistency as early as possible in the knowledge base derivation process.
2. Constraints allow the high-level representation of states of affairs that cannot obtain in the agent's environment. If a particular constraint is violated under the current explanation, the agent can immediately infer that the explanation is flawed, and thus can invest its energies elsewhere.
3. Due to the high-level nature of constraints, it is possible to associate a particular recovery plan with every constraint. By following the recovery plan, the agent may be able to produce a new explanation which successfully resolves the current problem.

Constraints relevant to the agent's domain are provided by the agent designer. As such, constraints represent preprocessed knowledge provided to the agent. Pollock (Pollock 1998) notes that this technique obviates the agent from having to concern itself with all the intricacies of the domain, and risk encountering the frame problem. While we accept this point, it should be noted that the agent's design is flexible enough that it may be possible to allow it to modify its constraints or add new ones as a result of experience, and indeed this is an interesting area for future developments.

Planning At The Intention Level

The traditional BDI approach to intelligent agent design recognises the primary importance of *beliefs*, *intentions*, and *desires* when analysing agent design and behaviour (Wooldridge 2000). Intentions occupy a middle ground between desires and actions which allow the agent to focus on a manageable number of achievable short or long-term goals which will maximise, to the best of the agent's knowledge and abilities, its long-term utility. In other words, intentions allow an agent to be goal-driven, rather than event-driven (Schut & Wooldridge 2001).

In (Lacey, Hexmoor, & Beavers 2002) we describe the implementation and testing of an agent capable of intention-based planning. The agent was able to generate a set of intentions which would take it from the present state to the

[1] This work stems from an attempt to build agents representing alternative metaphysical conceptions of the world. The philosophical foundations of this research are not relevant here. For more details concerning the philosophical and design issues concerning these agents, see (Lacey 2000) and (Lacey & Lee 2001).

[2] For more information on coherence as an approach to truth and justification, see (Audi 1993) and (Kirkham 1995).

[3] This structure was inspired by Quine (Quine 1980).

goal state. Small variations in the environment were successfully dealt with by altering the action which was associated with an intention, *without* changing the intention itself. When the agent perceived significant changes in the environment, however, it was capable of reconsidering its intentions. This agent was based on an extended version of coherence-based belief revision, and as such was named agent **CX**.

Intentions are combined to form a rooted intention tree in which each node represents a choice between alternative intentions. The paths from the root intention to any of the leaves represent alternative paths through the intention space.

Intentions incorporate the following concepts:

Action Templates A data structure containing templates which describe how actions can be formed which will bring about the intention.

Preconditions States that must obtain in the world before the actions can be implemented.

Postconditions States that will obtain in the world once the actions have been implemented.

Constraints A list of constraints which are associated with the intention.

While the concepts of constraints and preconditions may seem similar, they serve different functions within the intention reconsideration process. The purpose of the preconditions and post-conditions is to guide the backward chaining process. They do this by allowing the backward chaining engine to match the preconditions of a required intention with the post-conditions of an intention which must precede it. In this manner executable intention paths can be formed.

We consider intentions to be the pro-active equivalent of explanations. Competing alternative explanations are used to guide the backward chaining process which eventually yields a maximally coherent interpretation of the agent's environment. Similarly, competing alternative intentions are used to guide the planning process, and will yield a set of intentions which are maximally coherent with the agent's existing beliefs.

Just as dealing with low-level sensor data alone makes perception difficult, so dealing with low-level actions alone makes planning difficult. By considering its actions at the intention level, the agent is better equipped to deal with dependencies and inconsistencies that may exist between its planned actions.

Responsiveness can be contrasted with pro-activity. A pro-active agent does not merely respond to its environment, but is able to exhibit goal-directed behaviour. As shown in Figure 1, adding intentions to this architecture renders the agent pro-active, as well as responsive.

This is not to suggest that the agent will have to generate a competing set of intentions on the basis of every interpretation of the environment. Rather, the default interpretation will always be carried forward from the previous interpretation. If nothing has changed, the agent can go ahead and use its default set of intentions. Even if the interpretation has changed, an agent may still be able to use the default intentions. Whether or not this is possible will depend on whether or not any of the constraints which link the interpretations and intentions have been violated. If they have not, then the default set of intentions is still valid. If constraints have been violated, then the intention reconsideration process must begin.

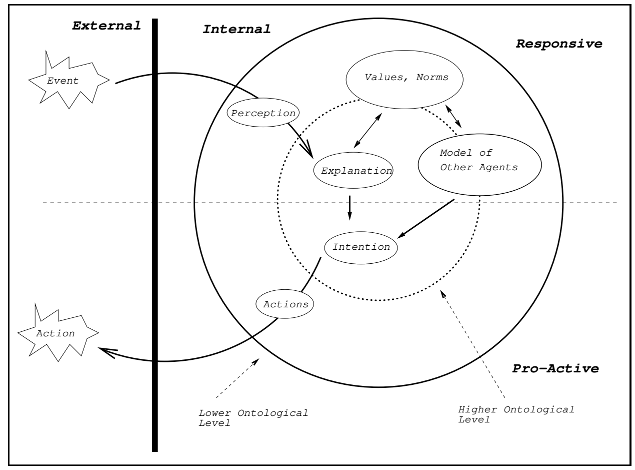

Figure 1: Adding Intentions to the System Yields Pro-Activity

An intention encapsulates the actions required to bring it about. It is assumed that the actions encapsulated within an intention are internally coherent. This assumption allows us to verify the coherence of the pro-active side of the agent only at the level of the agent's intentions.

Intentions are combined to form a rooted intention tree. The paths from the root intention to any of the leaves represent alternative paths through the intention space.

Intentions Give Meaning to Actions

Intentions combine to form coherent methods of achieving the desired goal. They are then translated into actions which are put into effect to achieve the desired intention. Note that while individual intentions are meaningful, individual actions have little meaning when taken in isolation from their related intentions. This reflects the distinction between human intentions and actions.

For example, consider my intention to type the word "for". At the intention level, I formulate the intention to type the word. At the action level, however, the actions required to carry out this intention are *all* of the form"Move hand horizontally to a certain position, then move fingers vertically."

The point is that very little meaning can be attached to actions in themselves. In order to determine the meanings of actions, we must refer to the intentions which the actions represent an attempt to bring about. This means that when reasoning about actions, we should be reasoning at the *intention* level, rather than the action level.

Intention constraints are used by the system to determine whether variations between intended and actual execution

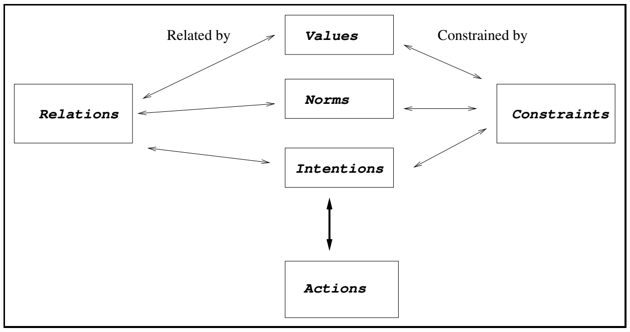

Figure 2: Relating Intentions, Actions, and Social Notions

have rendered the current set of intentions obsolete. If they have not, then intention reconsideration is not necessary. If they have, then the agent must reconsider its intentions.

Whether or not the intention constraints succeed, action constraints are applied prior to executing each action. Action constraints are designed to detect cases where the overall intention is still valid, but the original action which was originally associated with that constraint must be varied. This variation is entirely local to the current intention, and does not affect the rest of the system. In effect, this mechanism adds a degree of responsiveness to the system.

Planning for a Multi-Agent Environment

So far we have been considering the utility of representing intentions for an agent operating within a single agent environment. We now turn our attention toward the methods used to represent intentions and social notions in a multi-agent context.

Intentions in a Multi-Agent Environment

In (Hexmoor & Beavers 2002), Hexmoor and Beavers discuss extending the traditional BDI approach to agent design by also considering the concepts of values, obligations, and norms. They conclude that adding these new modalities increases the robustness and flexibility of the agents that are produced. In this paper we incorporate norms and values directly into the agent architecture, but we consider obligations to be implicitly represented.

We take values to be abstract beliefs that guide the behaviour of an agent, while norms are specific instances of behaviour. One or more norms may be associated with one or more values, and vice-versa. For example, consider an agent who holds the value that it is wrong to steal. Such an agent would also abide by norms of behaviour consistent with this value, such as not stealing and always ensuring that goods and services are paid for in full.

As shown in Figure 2, the relationships between values, norms, and intentions, are represented using relations and constraints. As in the single agent case described above, relations are used to encode the prescriptive relationships between the various entities, while constraints represent the inhibiting relationships.

In other words, relations are used to represent the fact that an agent which exhibits certain characteristics will also exhibit certain other characteristics. Constraints, on the other hand, are used to represent the fact that certain combinations of values, norms, and intentions are inconsitent.

Figure 2 also shows that values and norms are directly related to intentions, as opposed to actions. This is because, as described earlier, very little meaning can be attached to low-level actions, so relationships concerning the social make-up of agents must be defined at the intention level.

For example, consider an agent who takes goods from a store without paying for them. Whether or not this agent is violating the value that stealing is wrong depends entirely on what its intention was. If the agent was intending to steal the goods, then it has clearly violated a norm of behaviour that would be associated with the value that stealing is wrong. If, on the other hand, the agent was intending to take the goods home on approval, and was acting with the consent of the store owner, then no norms or values have been violated.

When combined with the explanation-based backward chaining mechanism described above, this framework allows the agent to construct default models of the social configuration of its fellow agents, and to generate plans on the basis of these models. The constraint violation detection mechanism will ensure that any agent behaviour which is inconsitent with the default model will be flagged as such, and will cause the model to be updated accordingly.

Planning Using the Intentions of Others

In (Wilkins & desJardins 2001), Wilkins and desJardins argue that planning systems should be capable of exploiting as much domain knowledge as possible. Our architecture addresses this requirement by providing a mechanism by which domain knowledge concerning the likely behaviour of other agents can be acquired, represented, revised, and used to construct intention-level plans.

We propose a mechanism by which agent **CX** constructs intention trees which incorporate the intentions of other agents. This in itself is not a new concept. In (Grosz, Hunsberger, & Kraus 1999), Grosz, Hunsberger, and Kraus distinguish *intentions-to* from *intentions-that*. An agent may hold an intention-that ϕ without *actually* knowing how to bring about ϕ, as long as it has a belief that another agent does know how to bring about ϕ. Intentions-to, on the other hand, refer to actions that the agent is capable of performing itself.

Unlike Grosz, Hunsberger, and Kraus, we do not distinguish between those intentions which will be carried out by the agent constructing the plan, and those that will be brought about by other agents. This is because, as mentioned above, agent **CX** constructs its plans *at the intention level*. As long as the agent keeps track of which intentions are due to be performed by which agent, it is perfectly acceptable to combine the intentions relevant to multiple agents within the same intention tree.

Figure 3 shows a simple multi-agent intention tree from a blocksworld domain. There are two agents in the environment. Their task is for one of the agents to exit from a room, but the exit is blocked by a block. The agent constructing the intention tree (agent **A**) is less well equipped to push the block than the other agent (agent **B**) is. In this simplified

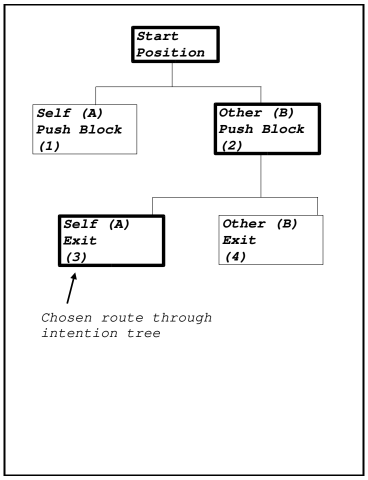

Figure 3: A Simple Multi-Agent Intention Tree

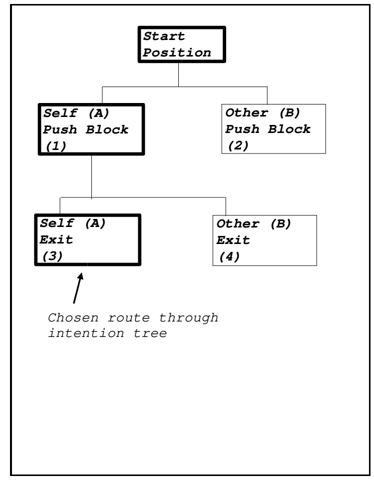

Figure 4: The Revised Intention Tree of Agent **A**

example there are only two possible intentions, `push` and `exit`. The intentions in bold are the ones that are actually selected.

The intention tree is constructed as follows. From the start position, the only intention which can be directly applied is the `push` intention. This is because the exit is not clear, and the exit being clear is a necessary pre-condition for the `exit` intention. The `push` intention is available to both agents, so separate nodes are created in which *self* or *other* are pushing the block. These nodes are labeled (1) and (2) in the figure.

Once the first layer of the tree is complete, it is checked for consistency. As both agents are able to push the block, the first layer of the tree is consistent. The next step is to evaluate which of the two available paths offers the highest utility. The utility of each path is determined by holistically evaluating the utility of the expected result of bringing about each intention. As agent **B** is better equipped to push the block than agent **A**, intention (2) receives a higher utility score than intention (1), and so intention (2) is selected.

The next layer of the tree is grown from intention (2). Now that the block is clear, the pre-condition of the `exit` intention is satisfied, so `exit` intentions are created for agents **A** and **B**. As there is no inconsistency with agent **A** exiting from the room now that the exit is clear, intention (3) is selected.

It is no coincidence that, in the example shown in Figure 3, intentions for agent **A** are added before intentions for agent **B**. In this example, as long as agent **B** adds intentions for agent **A** to its own intention tree before adding its own intentions, it will generate an identical tree to that generated by agent **A**. This mechanism allows adaptive multi-agent co-operation on complex tasks *without* the need for inter-agent communication.

The order in which agents add their intentions to the tree is determined by the system designer, and as long as it is the same for all agents, we can guarantee that all agents will independently generate identical intention trees. This is due to the way that the coherence-based backward chaining mechanism operates: in every case, the default path will be chosen, unless any constraints are violated.

Taking Values and Norms Into Account

We now consider how intention trees can be affected by beliefs concerning the values and norms of other agents. With social agents, once one forms a belief concerning the intentions of other agents, it can examine the relevance of those intentions to its intention tree.

Intentions are related to values and norms by relations and constraints, as illustrated in Figure 2. If agent **A** detects that agent **B** is executing an intention which is incompatible with the values and norms agent **A** previously ascribed to agent **B**, then agent **A** must re-evaluate the norms and values it believes are held by agent **B**.

Favouring minimal change, agent **A** examines the impact of agent **B**'s norms and values on its intention tree. The effects of the revised norms and values will be limited to the parts of the intention tree which are due to be executed by agent **B**. With respect to the goal of agent **A**, these effects will be either:

1. neutral, or
2. beneficial, or
3. detrimental

If a change in norms and values is neutral, then it does not affect agent **A**'s model of the intentions of agent **B**, and so can be ignored. For the change in the perceived norms and values of agent **B** to be beneficial to agent **A** implies that agent **A** believes that agent **B** will now play a more active role in the task at hand.

Conversely, it is also possible that a change in the norms and values ascribed to agent **B** by agent **A** are inconsistent with the intentions that agent **A** had assigned to agent **B**. In this case, agent **A** must re-generate its intentions, taking the revised norms and values of agent **B** into account.

For example, suppose that after observing agent **B**, agent **A** revised its opinion of the norms and values held by agent

B such that it became inconsistent for agent **B** to hold intention (2). These new conditions mean that even though agent **B** is better equipped to push the block, agent **A** now believes that agent **B** will not in fact push the block. This in turn means that agent **A** revises its intention tree to that shown in Figure 4.

Acknowledgements

This work is supported by AFOSR grant F49620-00-1-0302.

Conclusions

In this paper we have described how the representation of norms and values within an agent's ontology, combined with a coherence-based constraint satisfaction system, allows an agent to construct high level default models of the social configuration of other agents.

As long as the intentions derived from the agent's perceptions do not contradict the model, the agent will be able to construct intention-level plans which incorporate the intentions and actions of other agents. Furthermore, as long as all agents in the systems share the same ontology, all agents in the system will construct *identical* plans, and hence be able to co-operate in complex tasks without communication.

If, on the other hand, an agent observes another agent carrying out an intention which contradicts its existing model of the other agent's social configuration, it will revise its beliefs concerning the values and norms of the other agent. If these revised beliefs are inconsistent with the portions of its existing intention tree which are to be executed by the agent in question, then those parts of the intention tree will be revised as appropriate.

References

Audi, R. 1993. *The Structure of Justification*. Cambridge, UK: Cambridge University Press.

Doyle, J. 1992. Reason maintenance and belief revision. Foundations vs Coherence theories. In Gärdenfors, P., ed., *Belief Revision*, volume 29 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge: Cambridge University Press. 26–51.

Grosz, B.; Hunsberger, L.; and Kraus, S. 1999. Planning and acting together. *AI Magazine* 20(4):23–34.

Hexmoor, H., and Beavers, G. 2002. In search of simple and responsible agents. In *Proceedings of the GSFC Workshop on Radical Agents*.

Jennings, N. R.; Sycara, K.; and Wooldridge, M. 1998. A roadmap of agent research and development. *International Journal of Autonomous Agents and Multi-Agent Systems* 1(1):7–38.

Jonker, C. M., and Treur, J. 1998. Agent-based simulation of reactive, pro-active and social animal behaviour. In *Proceedings of the 11th International Conference on Industrial and Engineering Applications of AI and Expert Systems*, Lecture Notes in AI. IEA/AIE'98.

Kirkham, R. L. 1995. *Theories of Truth: A Critical Introduction*. Cambridge, Mass: The MIT Press.

Lacey, N., and Lee, M. 2001. The implications of philosophical foundations for knowledge representation and learning in agents. In Kudenko, D., and Alonso, E., eds., *Proceedings of the AISB 01 Symposium on Adaptive Agents and Multi-Agent Systems*, 13–24. York, UK: The Society for the Study of Artificial Intelligence and the Simulation of Behaviour.

Lacey, N.; Hexmoor, H.; and Beavers, G. 2002. Planning at the intention level. In *15th International FLAIRS Conference*. Pensacola Beach, Florida: FLAIRS. (To appear).

Lacey, N. 2000. *Investigating the Relevance and Application of Epistemological and Metaphysical Theories to Agent Knowledge Bases*. Ph.D. Dissertation, University of Wales, Aberystwyth.

Pollock, J. 1998. Planning Agents. In Rao, and Wooldridge., eds., *Foundations of Rational Agency*. Kluwer.

Quine, W. V. O. 1980. Two dogmas of empiricism. In *From a Logical Point of View : 9 logico-philosophical essays*. Cambridge, Mass: Harvard University Press. chapter 2, 20–46.

Schut, M., and Wooldridge, M. 2001. Principles of intention reconsideration. In *AGENTS'01*. Montreal, Canada: ACM.

Thagard, P., and Millgram, E. 1997. Inference to the best plan: A coherence theory of decision. In Rom, A., and Leake, D., eds., *Goal-driven learning*. Cambridge, MA: MIT Press. 439–454.

Wilkins, D. E., and desJardins, M. 2001. A call for knowledge-based planning. *AI Magazine* 22(1).

Wooldridge, M. 2000. *Reasoning About Rational Agents*. Cambridge, MA: MIT Press.