SRI International

EXPERIMENTAL ROBOT PSYCHOLOGY

Technical Note 363

November 5, 1985

By: Kurt G. Konolige
Artificial Intelligence Center
Computer Science and Technology Division

**APPROVED FOR PUBLIC RELEASE:
DISTRIBUTION UNLIMITED**

This research was made possible, in part, by a gift from the System Development Foundation. It was also supported, in part, by Grant N00014-80-C-0296 from the Office of Naval Research.

SRI International®

333 Ravenswood Ave. • Menlo Park, CA 94025
(415) 326-6200 • TWX: 910-373-2046 • Telex: 334-486

Abstract

In this paper I argue that an *intentional methodology* is appropriate in the design of robot agents in cooperative planning domains — at least in those domains that are sufficiently open-ended to require extensive reasoning about the environment (including other agents). That is, we should take seriously the notion that an agent's cognitive state expresses beliefs about the world, desires or goals to change the world, and intentions or plans that are likely to achieve these goals. In cooperative situations, reasoning about these cognitive structures is important for communication and problem-solving. How can we construct such models of agent cognition? Here I propose an approach that I call it *experimental robot psychology* because it involves formalizing and reasoning about the design of existing robot agents. It shows promise of yielding an efficient and general means of reasoning about cognitive states.

Contents

List of Figures

I. Introduction

Recently I had the experience of helping a friend paint the interior of a house. I was given the task of painting the walls in one room. As is normally the case, the ceiling had been painted first (if the ceiling were painted *after* the walls, there would be a substantial risk of sprinkling the walls with ceiling paint). Having had experience with this sort of thing before, I was particularly careful when painting the junction of the walls and ceiling, to make sure that I did not get any wall paint on the ceiling.

Exercising care in painting the junction cost me extra time and effort in a task I wanted to do as quickly and easily as possible. Had the circumstances been slightly different — if the ceiling had been unpainted, if my friend had wanted the walls painted very quickly, or if I had wanted to work fast because of an important appointment — I would not have been as careful. As a rational agent, I was influenced in carrying out the task by a variety of considerations: my beliefs about the state of the physical world, and the intentions and desires of my workmate. To have done otherwise would have been less than rational; one could justifiably accuse me of "blindly following orders," without taking into account circumstances that any reasonable person would consider in modifying his performance of a cooperative task. Such literal behavior is irrational because it results in a loss of efficiency in achieving an overall goal — the ceiling might have to be painted over again.

My purpose in employing this example is to suggest that an *intentional model* is an appropriate one to use in building cooperative robot planning systems. By intentional I mean that the cognitive state of the robot is constructed from a set of *beliefs* about the world and a set of *desires* or *goals* that it attempts to realize by forming *plans*. This model is both normative and descriptive: a robot would consitute an intentional system, and, in

addition, reason about other robots (or people, if they were involved) as if they also were intentional systems. By having robot planners reason about the beliefs and desires of other agents, we can hope to have robots achieve the type of rational cooperative behavior that people exhibit in such tasks as wall-painting.

The idea of endowing robots with an intentional character is certainly not a new one in artificial intelligence (AI); it has its roots in McCarthy's Advice Taker program [9]. More recently, Dennett [3] has advanced the notion that it is often useful from a descriptive standpoint to consider complicated programs (such as chess-playing machines) as intentional systems, even if they were not designed to represent beliefs or desires explicitly. I am not going to argue furtber for what might be called a *normative intentional methodology*;[1] instead of that, I will describe the characteristics of cooperative planning domains for which the full-blown intentional approach is appropriate. In these domains, it is essential that agents be able to reason effectively about the cognitive state of other agents. I will then suggest a useful approach for accomplishing this task is one I have called *experimental robot psychology*. This method involves analyzing the design of a robot agent's cognitive processes, axiomatizing them, and developing inference rules that can be "plugged back" into the agent.

[1]Until very recently there has been a notable lack of competing methodologies in the AI field. Rosenschein and Pereira [15] describe an alternative approach based on a theory of an automaton interacting with its environment.

II. Cooperative Planning in Open-Ended Domains

Problem domains for which a normative intentional approach is suitable have the following characteristics:

- There is a natural partitioning of the environment, either spatially or functionally. An example of the former is a surveillance task in which several areas must be patrolled; of the latter, assembly tasks in which major subcomponents can be assembled in parallel.
- Each process has a dynamic view of the environment: other processes may be performing tasks, and there may be chains of events that are the result of ongoing natural processes.
- Each process has partial or imperfect knowledge of the environment.
- The cost of communication among processes may be high, or the communication channels may be slow, so that there is a significant delay in sending large amounts of information.

I will call a problem domain that satisfies these criteria an *open-ended, cooperative domain*. The *cooperative* aspect comes from the partitioned nature of the solution: it suggests an approach in which processes in a network cooperate with one another to achieve certain goals, communicating to coordinate tasks and exchange information. By *open-ended*, I refer to the inherent complexity of the problem, which does not admit of enumerative or algorithmic solutions. I will elaborate on both these concepts.

Since the cost of communication is high, each process must be capable of reacting intelligently on its own to changes in the environment. Furthermore, because of the uncertainty

inherent in information gathered from sensory apparatus, and because of limitations on the functional capabilities of the processes, each process must have a well-developed model of its environment (including the presence of other cooperating agents), along with the ability to reason about actions and events in quite complex ways. Such autonomous intelligent processes will be called *agents*.

Relaxing any of the above criteria allows a simpler approach to the problem domain. The most obvious is the cost of communication. If processes are linked by reliable, high-speed communication channels, it makes more sense to centralize planning and coordination for all the processes than to provide each one with an autonomous ability to reason about its interaction with other processes. The collection of processes can be viewed as a large, coordinated machine. An example of this type of coordination is the operating system on mainframe computers: an executive program regulates the activity of both system and user programs, while a fast main memory is shared for interprocess communication.

Relaxing the criterion of imperfect information leads to a different simplification. In this case, communication costs can still be high, so a solution using distributed processors is efficient. However, because perfect information about the problem is available, planning can be conducted from a central location. Various subtasks are then distributed to the individual processes, along with the requisite information for their solution (*e.g.*, protocols for communication if one processor's subtask must be coordinated with others). This paradigm has been called *distributing the solution* by Davis [2]. Since a solution can be planned in a centralized manner, there is no need for individual processes to model or reason about the intentions of other processes.

Finally, the environment (which includes the processes themselves) must be complicated enough to require reasoning about intentionality. An example of a device for which an intentional model is *not* necessary is a thermostat. Although McCarthy [10] would allow thermostats to be described as having beliefs and goals, he admits that they can be understood without attributing such qualities to them. In the first place, the design of thermostats (at least the simple ones that run home heaters) does not include any explicit

representation of beliefs or goals as we are accustomed to finding them in AI programs. The bimetal spring may be an *indicator* of temperature, but it does not have the functionality of a belief: for example, no further facts about the world are ever deduced from it by the thermostat. It follows that, if the design of the thermostat is simple enough, I can predict its behavior without reference to beliefs or goals — I just note that, at certain temperatures, it will switch on the heater, while at another it will switch it off.

These examples are all by way of negation; the problem domains are simple enough (not *open-ended*) so that a set of processes could function effectively without incorporating a full-blown intentional structure. Typically, successful AI systems are limited to displaying narrow expertise in specialized fields: expert systems for diagnosing specific medical domains, programs for chess-playing, and so on. These systems do not exhibit "common-sense" intelligence; they have no ability to reason about what I want from them, except perhaps in a very limited fashion — for example, a "verbosity switch" or some similar feature might control the amount of output.

The type of reasoning I have in mind for intentional, cooperative agents is of a very different nature. I will try to give a broad outline of the necessary capabilities by dividing them into five general categories:

- Reasoning about the environment on the basis of what is known, including reasoning about the beliefs, goals, and plans of other agents.
- Communicating to exchange information about the environment and about intentions to act.
- Reasoning about the effects and interactions of future actions and events.
- Forming cooperative plans on the basis of all the above information.
- Monitoring and synchronizing the execution of individual plans.

This list is not meant to suggest a strict sequential division of the reasoning task. In practice, plan formation and execution are going on all the time, and execution of one plan

(e.g., a plan to communicate a request for assistance) may be necessary for the formation of another. Similarly, communication and reasoning about the environment can occur not only at the time of plan formation, but also during execution (for example, for synchronization of activities or recovery from errors).

The various components of an intentional system fit into this framework in the following way. An agent's *beliefs* represent the environment as the agent views it. In current AI technology, these beliefs are a set of sentences in some internal language, often a variant of the predicate calculus; some authors call the set of beliefs a *knowledge base* [8]. An agent reasons about the world by making inferences based on its beliefs. As new information comes in from its sensors, the agent tries to keep its beliefs consistent with the observed state of the world.

An agent also has a possibly ordered set of *desires* representing current goals of the agent, which it attempts to realize by performing actions. The agent must have some beliefs regarding its own capabilities for action; by reasoning about the state of the world and the way in which acting is likely to change it, an agent attempts to derive a set of *intentions* or *plans* for achieving some or all of its goals.

A special class of sensors and effectors provide the input/output channels used for communication with other agents. Communication is singled out as an action of special importance because of the role it plays in exchanging information among agents as they form and execute cooperative plans. It is significant that agents must reason extensively about the content of a communication before information can be exchanged effectively. For example, if agent A tells agent B that p is true, it does not necessarily follow that B believes this to be the case; if B has better information about p, he may know p to be false, and will therefore not accept A's utterance at face value. This is in direct contrast to communication in a "distributed-solution" domain, where messages have a precisely defined effect. In an open-ended domain, effective communication is made possible by agents' knowledge of one anothers' cognitive states.

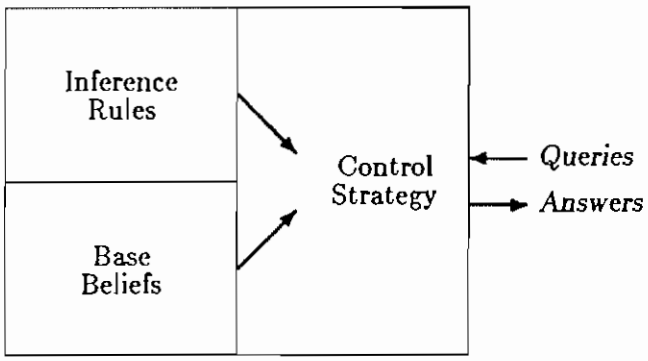
III. Experimental Robot Psychology

In the normative intentional methodology, an agent's cognitive state is constructed by dividing it into components: beliefs, goals, and plans. As I have already indicated, many of the reasoning tasks to be performed by agents depend upon knowledge of other agents' cognitive states. Representing and effectively reasoning about cognitive states are thus important topics in multiagent planning. Let me suggest an approach that I have employed successfully in the area of belief.

Consider a typical robot planning agent of the sort that was popular in the early 1970s, such as STRIPS [4]. Its domain is the *blocks world*, a simple abstract space of multi-colored blocks in which it was the only agent. The beliefs of this agent are given as a finite set of sentences in the first-order predicate calculus. STRIPS has a simple inference mechanism for deriving consequences of its beliefs; for example, from $ON(A, B)$ and $ON(x, y) \supset \neg CLEAR(y)$ it can deduce $\neg CLEAR(B)$. STRIPS also has a mechanism for building plans to achieve its goals; I will return to this in a moment, but for now I want to concentrate on the concept of belief.

The part of STRIPS that is dedicated to belief can be represented by the diagram of Figure III.1. There is a set of sentences, the *base sentences*, that are STRIPS' beliefs about the world. Inference rules can be applied by a control strategy to derive consequences of the base set. The whole box I call a *belief subsystem* to identify it is a component of the agent's cognitive state. A belief subsystem interacts with other components by means of queries and responses. For example, the planning process may need to know whether a predicate P is true of the world, so it issues a query about P. The control strategy receives the query and checks if P or $\neg P$ is one of the base beliefs. If it is, an answer can be returned

Figure III.1: A Belief Subsystem

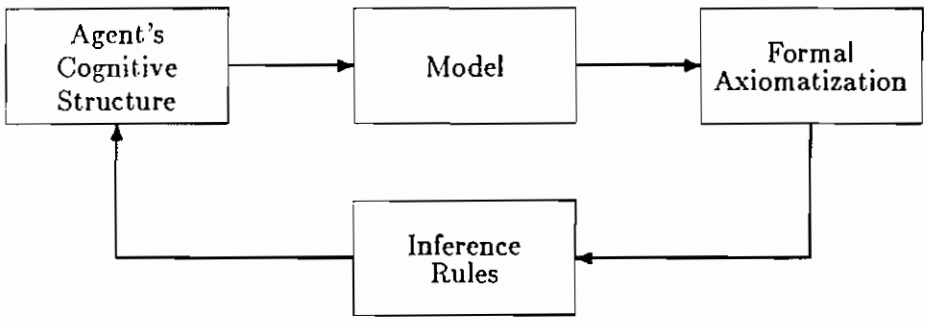


immediately. If not, the control strategy might try to derive P or $\neg P$, using the base beliefs and the inference rules. At some point, it either succeeds in a derivation or gives up and answers that P is not believed.

Belief subsystems, for the most part, are admirably suited to description and axiomatization in a formal system. Let L be the language of the belief subsystem (the base sentences and all derived beliefs are expressions in L); I will call L the *object language* because it is the object of the description. For the description language itself, I use another language — ML, the *metalanguage*. ML has terms that refer to expressions of L (which is what makes it a *meta*language), and a distinguished predicate *Bel*, such that $Bel(a, \lceil p \rceil)$ means the L-expression "p" is one of agent a's beliefs. The inference rules and control strategy of a belief subsystem can be described by writing suitable axioms in ML (I did this originally in 1980 [6], by assuming that the inference rules and control strategy formed a complete first-order proof system).

The point I want to make here is that it is often possible to develop cognitive models as a basis for reasoning about cognitive states by examining the internal design of robot agents. This is precisely what I did in arriving at the belief subsystem model. The formal axiomatization of the model then provides a means of reasoning about cognitive states. Of course, in the interest of efficiency and technical feasibility it is often necessary to abstract

Figure III.2: The procees of experimental robot psychology



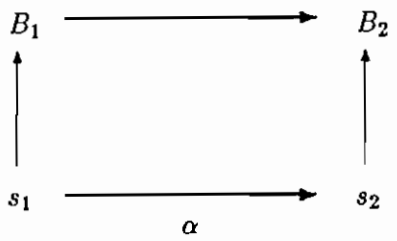
just the important properties of a model and make simplifying assumptions. For example, this led me to the *deduction model* of belief, in which the control strategy part of the belief subsystem is assumed to take on a particularly simple form [7].

Now suppose that this strategy has been carried through for some portion of the cognitive state, say for belief. I actually have in hand a formal means of reasoning about belief, based with fair accuracy on the design of the agent's belief subsystem. If I want the agent to reason about other agents' beliefs (and reflect introspectively upon its own), it is a natural step to simply include the belief model just developed as part of the agent's reasoning abilities. That is, the agent will view other agents as having belief subsystems similar to its own. I call this the *recursive property* of belief [7].

The whole process I have just described, that is, of developing a reasoning mechanism for belief that is useful to agents, can be outlined as in Figure III.2. The first part of the process — deriving a model by analyzing cognitive structure — I call *experimental robot psychology*. In many cases, as in the deduction model of belief, it can be a useful methodology for constructing models of robot cognition that can then be "plugged back" into the agent to serve as a means of having the agent reason about cognitive states.

The method of "plugging back" an axiomatization, that is, integrating it with the inferential mechanism already present in an agent, is not an easy one. As long experience in AI has shown, letting a general-purpose automatic theorem prover loose on a set of axioms is almost a guarantee of computational inefficiency. Many theorems are proved, but usually

Figure III.3: Beliefs and action



not the ones we consider important for reasoning about the intended domain. So there is a process of refinement of the axioms and inference rules that operate on them to achieve what McCarthy and Hayes (1969) have called *heuristic adequacy*: the ability to derive those consequences of the axioms that are necessary for commonsense reasoning. That is why I have drawn an additional box in the diagram labeled *inference rules* — these are the special inference techniques that must be developed to make reasoning about the model efficient.

Fortunately, the very manner in which experimental robot psychology proceeds makes available a method for achieving heuristic adequacy. Let me illustrate this by returning to the STRIPS example and examining the role of the belief subsystem in the planning process. STRIPS starts out with a set of sentences containing its beliefs about an initial situation, plus a goal sentence that it is supposed to make true by performing actions. The effect of an action is to change the state of the world, including the state of STRIPS' beliefs. I will diagram this in Figure III.3. In this diagram, STRIPS has a belief set B_1 in the initial situation s_1. The relation between the beliefs and the situation, indicated by the double arrow, is that the beliefs are intended to be true of the situation. If STRIPS performs action α, the state of the world will change to situation s_2; I have indicated this by drawing a line between s_1 and s_2, indexed by the action α. STRIP's beliefs in the new situation, B_2, should now reflect the changed state of the world, so that the double arrow again indicates that B_2 is true of s_2.

A theoretically justifiable way for an agent to plan would be to reason as follows: "Suppose that my beliefs are B_1; then the world looks like s_1. Under the influence of my

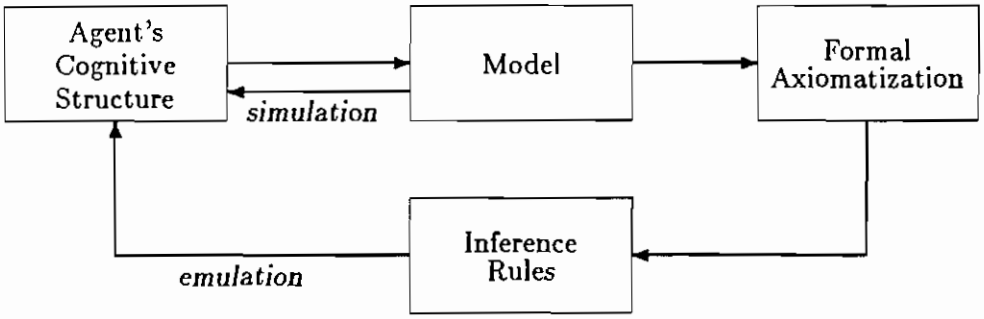
performing action α, the world would look like s_2, in which case my beliefs should be B_2." The advantage of this kind of reasoning is its flexibility; it accounts for the general relationship of belief and action, and so can be called upon to sanction many types of inferences — for example the concept of a *test*: an action performed to determine if some property was true of s_1 by examining its outcome in s_2. This is the type of theory advanced by Moore [13] in his thesis on the interaction of knowledge and action.

In STRIPS, on the other hand, actions are described by precondition-effect pairs that operate on the belief sets: if the preconditions of α are satisfied by B_1, then the action may be performed, changing B_1 into B_2 in the manner indicated by the effects. STRIPS bypasses (or perhaps, *compresses*, depending on your point of view) all of the general reasoning about action and belief — from B_1 to s_1 to s_2 to B_2 — by employing a syntactic transformation of belief sets. I have drawn a direct arrow from B_1 to B_2 in the diagram to indicate this.

To form a plan, one strategy STRIPS uses is to start with its beliefs in the initial situation, then to check whether the preconditions of any action are satisfied. If so, it applies the action by changing the belief set into a new one, based on the effect part of the action. STRIPS now has a belief set that applies to the new situation; it can use inference rules to deduce consequences of the new beliefs, just as it did in the initial situation. It is important to realize that, when STRIPS employs an action description in this manner, it is transforming a set of beliefs in one situation into *the set of beliefs that should obtain in the new situation resulting from the action.* For example, if STRIPS possesses the belief that the door is closed, and it successfully performs the action of opening the door, it should then believe that the door is open; STRIPS can arrive at this belief in the new situation either by direct observation of the door or by reasoning about the effects of the door-opening action that just took place. This use of action descriptions by STRIPS to form plans is, in essence, a *simulation* of its own actual behavior.

One consequence of this approach is that the generality of reasoning about the effect of actions on situations is nullified: STRIPS has no way of reasoning about tests, for example. What is gained is efficiency in the reasoning process. In the more general theory, axioms are

Figure III.4: Simulation by semantic attachment



written that describe the effects of actions, as well as the properties of belief subsystems (as I did previously with the *Bel* predicate), and then theorems are proved about the particular case involving s_1, B_1, and α. As I have argued above, this *emulation* strategy does not by itself generate a practical automatic reasoning system. But the STRIPS approach does. Its representation of action allows it to calculate the belief set B_2. With B_2 in hand, so to speak, it can then *simulate* the processing its belief subsystem would actually carry out in the new situation. So, in effect, STRIPS reasons about its beliefs in a future situation s_2 by acting as if it had beliefs appropriate to s_2, and letting its internal cognitive processes (such as belief derivation) operate in the normal manner. Efficiency is achieved if the internal cognitive processes are themselves efficient, which must be true in any case for any agent that interacts in a real-time environment.

There is a natural way to view STRIPS' simulative strategy in terms of the diagram outlining experimental robot psychology. Instead of proceeding from the model to a formal axiomatization and then, finally, to inference rules, the model itself is "plugged back" and used to reason about belief. As I have remarked, efficiency is achieved at the cost of generality, for the model can be used only if complete knowledge of a situation is available (STRIPS' representation of action demands that the belief set describe each situation completely, or at least, the part of it that is relevant to performing the action). So now the diagram looks like Figure III.4. Fortunately, there is a way of reconciling the efficiency of simulation with the generality of emulation: namely, to incorporate simulation as one

of the inference rules, applicable when complete information is available. Readers who are familiar with the work of Weyhrauch [16] will recognize the method of *semantic attachment*: proving facts about a model by running a computational version of it. To make this method truly useful, however, it must be generalized to work for at least some simple cases of partial information. For example, an agent could not use simulation to reason about the statement "There is someone whom John believes to be a spy," because John's belief is incompletely specified. This is where the idea of *partial models* becomes important. Partial models are models whose parts may remain unspecified. It is possible to use partial models for simulation when some information is missing.[1] In the deduction model of belief, I make extensive use of partial models in arriving at efficient proof methods.

[1]The working title of this paper was *Partial Models for Robot Cognition*; originally I intended to expand at some length on the topic of partial models, but instead realized that the methodology of experimental robot psychology would require explanation. So I will reserve the topic of partial models for a future paper.

IV. Conclusion

In this paper I have argued briefly for a *normative intentional methodology* in the design of robot agents in cooperative planning domains — at least in those domains that are sufficiently open-ended to require extensive reasoning about the environment (including other agents). Since this entails reasoning about the cognitive state of agents, I have proposed an approach to modeling cognitive states that is based on the already available design of current robot agents; I call this approach *experimental robot psychology*. When combined with the technique of *semantic attachment*, it shows promise of yielding an efficient and general means of reasoning about cognitive states.

Let me conclude with a few remarks about what experimental robot psychology will *not* achieve on its own in the cooperative planning domain, unless combined with significant theoretical underpinnings on other fronts. In developing the deduction model of belief, I already had available the design of belief subsystems, as realized in robot planning systems such as STRIPS. And the study of belief systems or knowledge bases continues to be an area in which much AI research is concentrated, as attested to by recent developments in introspective models [7,8,14] and default and nonmonotonic reasoning [1]. In other areas, theory-building efforts have either not yet begun or barely gotten underway, despite the fact that they are sorely needed for progress in the cooperative planning domain. Among these are a theory of action for concurrent and multiple-agent environments (although Georgeff [5] has made a start here), theories that relate conflicting goals to the formation of intention, and the general relation of intention to action (but see McDermott [12] for an interesting approach and discussion of issues). Also neglected are theories of complicated interactions that can arise in the real world, such as the revision of false belief under new information,

or the replanning process that must occur if some phase of a plan's execution fails. In all these areas, it is impossible to pursue the methodology of experimental robot psychology because the relevant cognitive structure does not exist.

Bibliography

[1] — Special Issue on Nonmonotonic Reasoning. *Artificial Intelligence 13:1-2*, 1980.

[2] Davis, R. and R. G. Smith, "Negotiation as a Metaphor for Distributed Problem Solving." Artificial Intelligence Laboratory Memo No. 624, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1981.

[3] Dennett, D. C., "Intentional Systems." *The Journal of Philosophy 28*, 1971, pp. 87–106.

[4] Fikes, R. E. and N. J. Nilsson, "STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving." *Artificial Intelligence 2:3-4*, 1971.

[5] Georgeff, M. P., "A Theory of Action for Multiagent Planning." In Proceedings of the Fourth National Conference on Artificial Intelligence, University of Texas at Austin, 1984.

[6] Konolige, K., "A first-order formalization of knowledge and action for a multiagent planning system." Artificial Intelligence Center Tech Note 232, SRI International, Menlo Park, California, 1980.

[7] Konolige, K., "A Deduction Model of Belief and its Logics." Doctoral dissertation, Stanford University, Stanford, California, 1984.

[8] Levesque, H. J., "A Formal Treatment of Incomplete Knowledge Bases." FLAIR Technical Report No. 614, Fairchild Laboratories, Palo Alto, California, 1982.

[9] McCarthy, J., "Programs with Common Sense." In *Semantic Information Processing*, M. Minsky (editor), MIT Press, Cambridge, Massachusetts, 1968.

[10] McCarthy, J., "Ascribing Mental Qualities to Machines." In *Philosophical Perspectives in Artificial Intelligence*, M. Ringle (editor), Humanities Press, 1978.

[11] McCarthy, J. and Hayes, P. J., "Some philosophical problems from the standpoint of Artificial Intelligence." In *Machine Intelligence 4*, B. Meltzer and D. Michie editors, Edinburgh University Press, Edinburgh, Scotland, 1969, pp. 120–147.

[12] McDermott, D., "Reasoning about Plans." In *Formal Theories of the Commonsense World*, J. Hobbs and R. C. Moore (editors), Ablex Publishing Corporation, Norwood, New Jersey, 1984.

[13] Moore, R. C., "Reasoning About Knowledge and Action." Artificial Intelligence Center Technical Note 191, SRI International, Menlo Park, California, 1980.

[14] Moore, R. C., "Semantical Considerations on Nonmonotonic Logic." Artificial Intelligence Center Technical Note 284, SRI International, Menlo Park, California, 1983.

[15] Rosenschein, S. and F. Pereira, "The Flow of Information in Physical Systems: An Alternative to the 'Representational' Paradigm for AI." Talk delivered at the Workshop on Reasoning about Cooperative Agents and Concurrent Processes, August 22–24, Monterey, California, 1984.

[16] Weyhrauch, R., "Prolegomena to a Theory of Mechanized Formal Reasoning." *Artificial Intelligence* **13**, no. 1–2, 1980.