On the Epistemic Foundations of Agent Theories

Ho Ngoc Duc

Institute of Informatics
University of Leipzig
Augustusplatz 10-11
D-04009 Leipzig, Germany

Phone: +49-341-9732206
duc@informatik.uni-leipzig.de

Abstract. We argue that none of the existing epistemic logics can adequately serve the needs of agent theories. We suggest a new concept of knowledge which generalizes both implicit and explicit knowledge and argue that this is the notion we need to formalize agents in Distributed Artificial Intelligence. A logic of the new concept is developed which is formally and practically adequate in the following sense: first, it does not suffer from any kind of logical omniscience. Second, it can account for the intuition that agents *are* rational, though not hyper-rational. Third, it is expressive enough. The advantages of the new logic over other formalisms is demonstrated by showing that none of the existing systems can fulfill all these requirements simultaneously.

1 Introduction

In recent years a number of approaches have been proposed in (Distributed) Artificial Intelligence to specify rational agents in terms of mental qualities like knowledge, belief, want, goal, commitment, and intention. (See [15] for an overview of some recent agent theories.) There is no clear consensus in the DAI community about precisely which combination of mental attitudes is best suited to characterizing agents, yet it seems to be an agreement that belief (or knowledge[1]) should be taken as one of the basic notions of the agent theory (see, e.g., [2], [11], [12], [13], [14], [15].)

Since agents need to act upon what they *actually* know, and not what they merely *possibly* know, agent theories must be based on logics that can capture what agents actually know. Although DAI researchers are aware that the modal approach to epistemic logic does not capture actual knowledge properly, they still use modal systems most frequently for modeling knowledge in their agent theories — for the lack of suitable alternatives. We shall argue that the use of modal epistemic logic is only justified in some very restricted domains, but for the more knowledge-intensive applications, other logics of knowledge are needed.

[1] For the purposes of the paper the distinction between knowledge and belief is irrelevant. We use the two terms interchangeably.

In the sequel I shall present a novel approach to epistemic logic which overcomes the weaknesses of the existing formalisms. The main idea is to combine epistemic logic with a complexity analysis: we consider how long an agent will need to compute the answer to a certain query. We shall show that our approach offers an intuitive solution to the logical omniscience problem while preserving the intuition that agents are rational. In the next section we shall review briefly the epistemic concepts which have been discussed in the literature and argue that none of them is suitable for formalizing the informational aspect of intelligent agents. Then we shall present the main intuitions underlying our novel approach, introduce a new concept of knowledge which generalizes both implicit and explicit knowledge, and develop formal theories of this new concept.

2 Implicit knowledge, explicit knowledge, and agent action

Originally, epistemic logic was developed to formalize the concept of knowledge as it is used in normal discourse, to describe what agents (actually) know. It was soon realized that this concept cannot be described by any "interesting" logic: it does not seems to obey any (nontrivial) logical law. From the information that an agent knows certain sentences one cannot infer what else he knows. It cannot be reliably assumed that agents know even the most elementary logical consequences of what they knows. To make epistemic logic possible, idealizations must be made concerning the reasoning capacities of the agents, and modal systems have been proposed to describe such idealized agents. However, the idealizations made by modal epistemic logic are too strong for any realistic agent: it is assumed that agents know all logical consequences of what they know, including all logical truths. Such perfectly rational, logically omniscient agents are nonexistent. Thus, to save modal logic as logic of knowledge, a new interpretation of epistemic logic has been proposed: the concept of implicit knowledge is invented, and modal epistemic logic is now interpreted as as describing this concept. That is, epistemic logic is not taken as describing what an agent *actually* knows, but only what is implicitly represented in his information state, i.e., what logically follows from his actual knowledge. What an agent actually knows is called his explicit knowledge.

From the viewpoint of agent theories, actual (explicit) knowledge is clearly more important than implicit knowledge: it is the former kind of knowledge that agents can act upon, but not the latter. The mere implicit knowledge that some path connecting all towns in a region is the shortest one is useless for a traveling salesman who seeks to maximize his profit — he must make this implicit knowledge explicit in order to choose what path to travel. The mere implicit knowledge that a certain strategy leads to victory is useless for a chess player who must make the next move within a short time. An information agent whose knowledge is represented as a knowledge base must normally make complex and time-consuming inferences before he can answer a query.

Because of the importance of explicit knowledge for agents' action, the search for logics of explicit knowledge still continues, and a number of systems have been proposed for that purpose. (For overviews and discussions see, e.g., [4] and [7, Chapter 9].) Although the concrete solutions are very different, the main strategy is the same in most cases: the reasoning capacities of agents are restricted by some *ad hoc* postulates. Most approaches following this strategy can in fact avoid logical omniscience to some extent, and logics of explicit knowledge can be constructed, which, like Eberle's system ([5]), can even "[provide] for total ignoramusses (ones who knows nothing), complete idiots (ones who cannot draw even the most elementary inferences), and ultimate fools (ones who believe nothing but contradictions) ..."[2].

Why are modal epistemic logics still widely used in agent theories despite the facts that implicit knowledge is useless when agents need to act *and* logics of explicit knowledge are readily available? The answer is simple: because we want to model *rational, intelligent* agents, and not "complete idiots". Logics of explicit knowledge avoid logical omniscience, but they cannot offer anything what can account for the rationality of agents. Surely agents are not perfectly rational, yet they *are* rational. Facing the dilemma between "perfectly rational agents" and "complete idiots", agent theorists understandably opt for the former and use logics of implicit knowledge for modeling their agents, hoping that such logics can describe "almost correctly" what agents actually know.

The assumption underlying the use of modal epistemic logics may be justified in some simple domains ("small worlds", "toy examples"), where the reasoning tasks involved are quite simple, where the decision process is not very complex, or when the time available is unlimited. In such simple domains, it can be assumed that whenever an agent needs some (implicitly available) information, he can perform the necessary inferences to have the information explicitly. However, such an assumption is not justified in more complex applications. Agents normally have to act under tight time constraints, their decisions what actions to be performed depend heavily on their actual knowledge, and the reasoning needed for making correct choices can be very complex and time-consuming. Take the aforementioned examples: calculating the shortest tour linking all towns in a region, computing the winning strategy in chess, and inferring the answer to a query from a given database are all very hard problems. It is obvious that modal epistemic logics and other logics of implicit knowledge cannot describe correctly what agents actually know in such applications. To describe agents realistically in knowledge-intensive applications, we simply need other logics of knowledge.

What properties should a logic of knowledge have if it is to be useful for describing realistic, implementable agents? The first obvious requirement is that it

[2] Other logics of explicit knowledge could be developed, e.g., within the "impossible worlds" framework ([10]) or by using an "awareness filter" ([6]). Strictly speaking, Levesque's logic of "explicit" belief ([9]) and Konolige's deduction model ([8]) must be viewed as logics of implicit belief, because what is believed by an agent in these models is *not* immediately available to the agent, but must be inferred using some (classical or relevant, complete or incomplete) deduction mechanism.

must not suffer from the LOP. However, solving the LOP does not automatically make a logic suitable for reasoning about knowledge. There are other requirements that the logic must fulfill. It is important that the logic can do justice to the intuition that agents are rational: although the agents do not automatically know all consequences of their knowledge, they are in principle able to do so. Because of this rationality the agents are able to act upon their knowledge: they can answer questions based on their knowledge, they can plan their actions in advance, they can predict what other agents can and will do, and so on. If a logic cannot account for the agents' rationality, then there is hardly any justification at all to call it a logic of knowledge. Another important requirement is that the logic be expressive enough to formalize "interesting" situations. This condition must remain somewhat vague, because different applications will require different expressive powers of the logic. However, we should keep in mind that the complexity of a logic generally increases with its expressive power, so we must try to find a good trade-off between expressiveness and simplicity.

3 Knowledge, reasoning, and time

The existing systems of epistemic logic try to deal with either implicit knowledge or explicit knowledge. The former concept is not very helpful when agents must act, while the latter is not governed by any non-trivial logic. To develop useful alternatives to existing epistemic logics we suggest to consider another notion of knowledge. The main intuition is the following. An agent's action depends not only on what he knows *now*, but also on what he can infer within some specific amount of time (intuitively, the time within which a decision must be made — a classical example being the time available to make the next move in chess.) An agent may not know a sentence *now*, but he may possess a procedure to 'prove' that sentence within a certain amount of time, where the amount of time needed depends on the complexity of the sentence, the agent's reasoning power, etc. If an agent knows that another agent must act under some time constraint, he may infer what the second agent can or cannot know within this constraint and predict his action accordingly. Therefore, it is worth considering what the agents can know within $1, 2, 3, \ldots$ time units, and not just what they currently know, i.e., what they know within 0 unit of time.

We want to represent not only *what* agents know or can know, but also *when* they are expected to know what they can know. The first question is answered by specifying the logic used by agents in their reasoning, and the second by a complexity analysis. What time structure do we need in modeling that kind of knowledge? Temporal logics have been dealing with linear and branching, point-based and interval-based, qualitative and quantitative time structures. Our obvious choice is a point-based, linear structure with a metric defined on it, because temporal constraints are usually given in quantitative terms and over a linear time line. For simplicity we assume time to be isomorph to the natural numbers (with the usual ordering and metric.)

The language we consider extends the usual language of the propositional calculus by n two-place knowledge operators $K_1, \ldots, K_n$, each for one agent, such that $K_i^x \alpha$ is a formula whenever x is a natural number and α is a formula. The formula $K_i^x \alpha$ can be read "agent i knows α within x units of time" and is interpreted: "if agent i chooses to 'derive' α from his current knowledge, then after at most x time units he will succeed", or alternatively, "if asked about α, i is able to derive reliably the answer 'yes' within x units of time". That is, we require not only that i has at least one procedure to 'prove' α, but also that i be able to choose the correct procedure leading to α under the given time constraint, namely, to arrive at the conclusion α after at most x time units[3]. The word 'prove' (or 'derive') should not be interpreted too narrowly as 'deductive proof': the procedure to gain the knowledge of α may be any acceptable method, e.g., sensing the environment, looking up in a standard reference book, or asking some expert.

Formally, our language is defined as follows:

Definition 1. Let $\mathbb{N}$ be the set of natural numbers and Var be a set of number variables. Let $Agent$ be a finite set of agents and At be a countably infinite set of atomic formulae.

1. The set of temporal terms is the least set $Term$ such that $\mathbb{N} \subseteq Term$, $Var \subseteq Term$, and $t_1, t_2 \in Term$ implies $t_1 + t_2 \in Term$
2. The set of formulae is the least set Fml such that $At \subseteq Fml$, $\{\alpha, \beta\} \subseteq Fml$ implies $\{\neg\alpha, \alpha \wedge \beta\} \subseteq Fml$, $x \in Var$ and $\alpha \in Fml$ imply $\forall x \alpha \in Fml$, and $i \in Agent$, $t \in Term$, $\alpha \in Fml$ imply $K_i^t \alpha \in Fml$.

The other propositional connectives $(\vee, \rightarrow)$ and the existential quantifiers $(\exists)$ are defined as usual.

Our notion of knowledge could be called "algorithmic knowledge": knowledge is tied up with an algorithm to establish it. But the term should be used with care: "algorithmic knowledge" has been used with different meanings elsewhere ([1], [7]). In our framework, "i knows α explicitly" can be defined as $K_i^0 \alpha$. Implicit knowledge can be defined as what follows logically from explicit knowledge (relative to some suitable logic.) Another useful concept of implicit knowledge can be defined as what can be reliably established by the agent: "i knows α implicitly" means $\exists x K_i^x \alpha$. The rationality of agents is expressed through two capacities: first, the ability to draw logical consequences from what is already known, and second, the ability to compute the complexities of certain reasoning problems in order to infer when something can be known. Note that these too capacities are implementable. It turns out that we can develop quite rich theories of the notion of knowledge we have introduced.

[3] This reading does not imply that agent i *will* know α at time $t_{now} + n$, where t_{now} is the current time. If the agent is not asked to provide the information α, then she has no reason to waste her resources in order to find a useless answer: The aspect of goal-directedness is implicit in our concept of knowledge.

To develop logics of algorithmic knowledge we try to establish logical relationships among the formulae of the language we have defined. This is done by developing our earlier framework ([3], [4]) a step further. In [3] and [4] we have suggested a new approach to epistemic logic which overcomes the drawbacks of existing approaches. The idea is to consider the evolution of one's knowledge over time: at one moment an agent may or may not know (explicitly) a certain consequence of his knowledge; however, he can perform some reasoning steps to know it at some moment in the future. We have argued that the traditional approaches fail to capture the concept of actual knowledge correctly because they do not take the cost of inferring new information into account: they assume that whenever an agent knows all premises of a valid inference rule then she automatically knows the conclusion. We have shown that axioms for epistemic logics must have the form: "if the agent knows all premises of a valid inference rule, and if she performs the correct inference step, then she will know the conclusion". Following this idea we have developed logics that can solve all forms of the logical omniscience problem and at the same time can account for the intuition that agents are rational beings. However, the systems presented (and indicated) in [3] and [4] have too little expressive power: they are not able to describe situations where introspective reasoning or reasoning about the reasoning of other agents is required. Moreover, they are based on a non-metrical branching time structure, which makes it very difficult to deal with time constrains. These problems are avoided in the present paper.

Our logics of knowledge will be built up step by step from some basis logic. We shall take classical logic as our basis logic. As we assume the natural numbers as our time structure, we shall also assume some laws of number theory. For our purposes it suffices to assume Presburger arithmetic (i.e., additive number theory.) Our epistemic systems will be obtained by adding (proper) epistemic laws to this basis. Now let us see how such laws may look like.

Suppose that an agent i knows α within x units of time, i.e., he needs x time units to infer α if needed. Then it is plausible to assume that he is able to do it when even more time is available. So we can take as axiom all ground instances of the formula $K_i^x\alpha \to K_i^y\alpha$, where $x < y$. Note that this axiom *does not* say that knowledge is persistent in the sense that once established it will be available henceforth. In this aspect our present approach makes a more realistic assumption than the persistence axiom in [4].

Now let us assume that an agent i knows explicitly $\alpha_1, \ldots, \alpha_n$ and that $\alpha_1 \wedge \ldots \wedge \alpha_n \to \beta$ is valid (i.e., β follows logically from the premises $\alpha_1, \ldots, \alpha_n$). What can be said about agent i's information state? Of course we cannot assume that i knows β automatically, even if β can be deduced from $\alpha_1, \ldots, \alpha_n$ algorithmically: perhaps i just does not care about β and does not even try to prove it. If he wants to know β then he has to perform some inferences, which can be very hard and time-consuming. His reasoning takes some time, so we can only say reliably that he will know β after some time if he chooses to derive it: $\exists x K_i^x \beta$. So, if $\alpha_1 \wedge \ldots \wedge \alpha_n \to \beta$ is a theorem then we can assume that $K_i^0\alpha_1 \wedge \ldots \wedge K_i^0\alpha_n \to \exists x K_i^x\beta$ is valid.

In the preceding paragraph we have considered the case when the premises $\alpha_1, \ldots, \alpha_n$ are explicitly available to the agent, so that he can use them immediately to derive the conclusion β. Consider now the more general case where the premises $\alpha_1, \ldots, \alpha_n$ are not immediately available but need to be inferred separately, i.e., $K_i^{x_1}\alpha_1, \ldots, K_i^{x_n}\alpha_n$ for some $x_1, \ldots, x_n$. Because $x_j < x_1 + \ldots + x_n$ for all $j = 1, \ldots, n$ we may assume that i can derive every premise within $x_1 + \ldots + x_n$ units of time. Once the premises are available they can be used to infer the conclusion β. Thus, it is plausible to adopt the principle $\exists x_1 K_i^{x_1}\alpha_1, \ldots, \exists x_n K_i^{x_n}\alpha_n \to \exists y K_i^y \beta$, provided that $\alpha_1 \wedge \ldots \wedge \alpha_n \to \beta$ is valid.

As a special case of this general principle we have the axiom $\exists x K_i^x \alpha \wedge \exists y K_i^y(\alpha \to \beta) \to \exists z K_i^z \beta$, which say that agent i can use modus ponens in his reasoning: if he can derive both α and $\alpha \to \beta$ then he is also able to derive β. Another special case allows us to infer $\exists x K_i^x \alpha$ from α. This inference rule says that all logical truths can in principle be known, though they are not known immediately as required by modal epistemic logic. We shall use this inference rule and the mentioned axiom schema to axiomatize our basic logic.

Definition 2. The logic **LAK0** (Logic of Algorithmic Knowledge) consists of all propositional tautologies (in the language Fml), the theory of Presburger arithmetic (with respect to the numerical part of the language), and the following axioms and rules of inference:

KA1. $\exists x K_i^x \alpha \wedge \exists y K_i^y(\alpha \to \beta) \to \exists z K_i^z \beta$
KA2. $K_i^x \alpha \to K_i^y \alpha$, for all pairs x, y such that $x < y$.
KR. From α infer $\exists x K_i^x \alpha$
MP. From α and $\alpha \to \beta$ infer β

The notions of a proof, of theoremhood etc. are defined as usual.

It is easy to see that **LAK0** is consistent and that it solves all variants of the logical omniscience problem. To see this it suffices to observe that the set $\{\neg K_i^0 \alpha | \alpha \in Fml\}$ is consistent with **LAK0**, i.e., **LAK0** can describe agents who (at some of their information states) know nothing explicitly. (However, they always know something implicitly.) Likewise, it is easy to see that what an agent explicitly knows (i.e., what she knows in 0 unit of time) needs not be closed under logical consequences or even under any logical law, e.g., $K_i^0 \alpha \wedge K_i^0(\alpha \to \beta) \wedge \neg K_i^0 \beta$ is perfectly **LAK0**-consistent. On the other hand, agents described by our logic are rational: they can draw all logical consequences of their knowledge if the necessary resources are available, as the following lemma shows.

Lemma 3. *The following rule of inference is valid:*

$$\frac{\alpha \to \beta}{\exists x K_i^x \alpha \to \exists y K_i^y \beta}$$

In particular, from $\alpha \to \beta$ one can infer $K_i^0 \alpha \to \exists x K_i^x \beta$.

Proof. Suppose that $\alpha \to \beta$ is a theorem. By (**KR**) we can infer $\exists z K_i^z(\alpha \to \beta)$. The formula $\exists z K_i^z(\alpha \to \beta) \to (\exists x K_i^x \alpha \to \exists y K_i^y \beta)$ is equivalent to (**KA1**) and is therefore a theorem of **LAK0**. Applying modus ponens we get the desired result.

4 Knowledge and complexity

Now we have developed **LAK0**, a basic logic of algorithmic knowledge which can account for rational, but not hyper-rational agents. The set of **LAK0**-theorems is recursively enumerable and so can be generated algorithmically. If β follows from α and an agent i knows α then he can employ a theorem prover for **LAK0** to deduce β, so he will know β after some time.

But how long will the agent need to infer a formula which follows logically from his knowledge base? Recall that our aim is to represent not only *what* agents know or can know, but also *when* they are expected to know what they can know. Our analyses up to now can only answer the first question. To answer the the second question, a complexity analysis is needed. The underlying idea is simple: if an agent i receives a query of length l and the complexity of computing answers to queries of that class is a function f, then after at most $c_i f(l)$ units of time he is expected to have the answer, where c_i is a number that measures the computation speed of i.

It is still an open question what is the exact complexity of the decision problem for **LAK0**. It is well-known that Presburger arithmetic is decidable, but we still do not know whether or not the system **LAK0** is decidable. Therefore a complexity analysis for the whole system **LAK0** seems to be impossible. However, there are problem classes which can be expressed in **LAK0** and whose complexities are known. Hence we can analyze such problems and estimate the amount of time an agent would need to infer some piece of information. With the help of complexity theory we can obtain epistemic principles for specific problem classes. Adding those principles to the basic system **LAK0** will yield more powerful logics of algorithmic knowledge. Let us consider some examples.

Assume that currently an agent i's explicit knowledge (his knowledge base) is a set $\gamma = \{\alpha_1, \ldots, \alpha_n\}$ of sentences. Let β be a propositional consequence of γ. We cannot assume that i automatically knows β, however simple the derivation of β from γ may be. To know β, i has to perform some inferences, which take some time. The amount of time needed to answer the query β can be estimated with the help of complexity theory. Let $m = \|\alpha_1\| + \ldots + \|\alpha_n\|$ be the size of γ and $l = \|\beta\|$ be the size of β. To determine if β is in fact a consequence of γ, our agent may check if every truth-value assignment which falsifies β also falsifies γ. There are at most 2^l such assignments, and the time needed to check if an assignment satisfies γ is polynomial to the size of γ. Hence, i can reliably know β in some time proportional to $P(m)2^l$ for some polynomial P. Let c_i be a number that measures the computation speed of i. Then the time the agent i needs to infer β is $t = c_i P(m)2^l$, hence $K_i^0\alpha_1 \wedge \ldots K_i^0\alpha_n \to K_i^t\beta$ is valid.

Interestingly, the previous analysis can be used by an agent within the system in order to reason about other agents, provided that he has a built-in mechanism to calculate the complexity of reasoning problems. An agent k can reason about another agent i exactly like we did to expect i to know β within $t = c_i P(m) 2^l$ time units. But to estimate the time i would need to derive β, k does not have to actually derive it. He has only to calculate the complexity of β, which can be accomplished in a short time. So if $\alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \beta$ is a propositional tautology then we may infer $K_k^0 K_i^0 (K_i^0 \alpha_1 \wedge \ldots K_i^0 \alpha_n) \rightarrow K_k^{t_k} K_i^t \beta$, where t_k is some fairly small number (which depends on k's capacities).

What about self-introspection? Positive introspection is relatively easy, so for any formula α we may adopt the axiom $\forall x (K_i^x \alpha \rightarrow K_i^{x+1} K_i^x \alpha)$, which says that after proving α, agent i may introspect his knowledge and discovers that he knows α. For negative introspection we cannot expect to have any axiom of that generality, but we may still have some rules for special cases.

It is sometimes important to know not only what an agent knows, but also what he does not know within a certain time limit. For instance, when we use public-key cryptography to encrypt a message, we want to be sure that someone without the secret key will not be able to know its content within reasonable time — although he can in principle infer it from the public key. The expectation that our message cannot be quickly decrypted is based on the complexity of the reasoning required: we use lower complexity bounds to estimate the least amount of time that an agent would need to infer some sentence, and so to infer what he cannot reliably know within some given limit of time.

5 Conclusions

We have argued that existing agent theories are developed on inadequate epistemic foundations. We have then proposed a new epistemic concept, the concept of algorithmic knowledge, which generalizes both notions of knowledge considered in the literature: that of explicit and that of implicit knowledge. Logics of our concept of algorithmic knowledge have been developed following a clear methodology. The main idea is to combine epistemic logic with a complexity analysis: we consider how long an agent will need to compute the answer to a certain query. It is shown that our approach can account adequately for our intuitions about knowledge and that it solves the problems associated with other approaches. In contrast to logics of implicit knowledge such as modal systems or the deduction model, our logics can describe realistic, entirely non-omniscient agents. But unlike logics of explicit knowledge, our systems are capable of describing *rational* agents, who can use their reasoning capacities to infer new information from what they know. Moreover, our systems have a greater expressive power than most existing logics. For example, time constraints can be expressed easily in our framework.

Currently there still exists a wide gap between agent theories and agents existing in practice. Our work is an attempt to bridge this gap. We are trying to develop theories of mental concepts that make much more realistic assumptions about agents than other theories. Our work is guided by the principle that the capacities attributed to agents must be implementable. Much remains to be done to develop our framework further. But we firmly believe that our framework is a very useful one, which can be used to represent the kind of knowledge needed by agent theories better than any other existing logic of knowledge.

References

1. K. Binmore and H. S. Shin. Algorithmic knowledge and game theory. In C. Bicchieri and M.-L. Dalla-Chiara, editors, *Knowledge, belief, and strategic interaction*, chapter 9, pages 141–154. Cambridge University Press, Cambridge, 1992.
2. P. R. Cohen and H. J. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42:213–261, 1990.
3. H. N. Duc. Logical omniscience vs. logical ignorance. On a dilemma of epistemic logic. In C. P. Pereira and N. Mamede, editors, *Progress in Artificial Intelligence. Proceedings of EPIA'95*, volume 990 of *Lecture Notes in Artificial Intelligence*, pages 237–248, Heidelberg, 1995. Springer Verlag.
4. H. N. Duc. Reasoning about rational, but not logically omniscient agents. *Journal of Logic and Computation*, 7, 1997. Available online at http://arp.anu.edu.au/arp/duc/papers.html.
5. R. Eberle. A logic of believing, knowing and inferring. *Synthese*, 26:356–382, 1974.
6. R. Fagin and J. Y. Halpern. Belief, awareness and limited reasoning. *Artificial Intelligence*, 34:39–76, 1988.
7. R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning about Knowledge*. MIT Press, Cambridge, Mass., 1995.
8. K. Konolige. *A Deduction Model of Belief*. Pitman Publishing, London, 1986.
9. H. Levesque. A logic of implicit and explicit belief. In *Proceedings AAAI-84*, pages 198–202, Austin, TX, 1984.
10. V. Rantala. Impossible worlds semantics and logical omniscience. *Acta Philosophica Fennica*, 35:18–24, 1982.
11. A. S. Rao and M. P. Georgeff. Modeling rational agents within a BDI-architecture. In R. Fikes and E. Sandewall, editors, *Proceedings of Knowledge Representation and Reasoning (KR&R-91)*, pages 473–484, San Mateo, CA., 1991. Morgan Kaufmann Publishers.
12. Y. Shoham. Agent-oriented programming. *Artificial Intelligence*, 60(1):51–92, 1993.
13. M. P. Singh. *Multiagent Systems: A Theoretical Framework for Intentions, Know-How, and Communications*, volume 799 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, Heidelberg, 1994.
14. M. Wooldridge. Practical reasoning with procedural knowledge: A logic of BDI agents with know-how. In D. Gabbay and H-J. Ohlbach, editors, *Practical Reasoning. Proceedings of FAPR'96*, volume 1085 of *Lecture Notes in Artificial Intelligence*, Heidelberg, 1996. Springer Verlag. Available online at http://www.doc.mmu.ac.uk/STAFF/mike/papers.html.
15. M. Wooldridge and N. R. Jennings. Intelligent agents: Theory and practice. *Knowledge Engineering Review*, 10(2):115–152, 1995. Available online at http://www.doc.mmu.ac.uk/STAFF/mike/ker95/ker95-html.html.