Agent Theories, Architectures, and Languages: A Survey

Michael J. Wooldridge
Dept. of Computing
Manchester Metropolitan University
Chester Street, Manchester M1 5GD
United Kingdom
EMAIL M.Wooldridge@doc.mmu.ac.uk
TEL (+44 61) 247 1531
FAX (+44 61) 247 1483

Nicholas R. Jennings
Dept. of Electronic Engineering
Queen Mary & Westfield College
Mile End Road, London E1 4NS
United Kingdom
EMAIL N.R.Jennings@qmw.ac.uk
TEL (+44 71) 975 5349
FAX (+44 81) 981 0259

Abstract

The concept of an *agent* has recently become important in Artificial Intelligence (AI), and its relatively youthful subfield, Distributed AI (DAI). Our aim in this paper is to point the reader at what we perceive to be the most important theoretical and practical issues associated with the design and construction of intelligent agents. For convenience, we divide the area into three themes (though as the reader will see, these divisions are at times somewhat arbitrary). *Agent theory* is concerned with the question of what an agent is, and the use of mathematical formalisms for representing and reasoning about the properties of agents. *Agent architectures* can be thought of as software engineering models of agents; researchers in this area are primarily concerned with the problem of constructing software or hardware systems that will satisfy the properties specified by agent theorists. Finally, *agent languages* are software systems for programming and experimenting with agents; these languages typically embody principles proposed by theorists. The paper is *not* intended to serve as a tutorial introduction to all the issues mentioned; we hope instead simply to identify the key issues, and point to work that elaborates on them. The paper closes with a detailed bibliography, and some bibliographical remarks.

1 Introduction

One way of defining AI is by saying that it is the subfield of computer science which aims to construct *agents* that exhibit aspects of intelligent behaviour. One view, which would nowadays be regarded as extreme by many AI researchers, is that these agents will recreate intelligent human behaviour in all respects; a perhaps more widely held view is that even if human intelligence is out of the question, (at least for the time being), it would nevertheless be useful to be able to build agents that can exhibit *some aspects* of intelligent human behaviour. The notion of an *agent* is thus central to AI. It is perhaps surprising, therefore, that until the mid to late 1980s, researchers from mainstream AI gave relatively little consideration to the issues surrounding agent synthesis. Since then, however, there has been a marked flowering of interest in the subject, and the concept of an 'agent' has been adopted by a variety of subdisciplines of AI and mainstream computer science. One now hears of 'agents' in software engineering, data communications and concurrent systems research, as well as robotics, AI, and distributed AI. A recent article in a British national daily paper made the following prediction:

> 'Agent-based computing (ABC) is likely to be the next significant breakthrough in software development'[1].

[1] 'Back to school for a brand new ABC'. *The Guardian*, March 12th, 1992, page 28. See [55] for a (somewhat inaccurate) overview of 'agent based computing' from the popular science press.

A whole programming paradigm has even been christened 'agent-oriented programming' [121]. Our aim in this paper is to survey what we perceive to be the most important issues in the design and construction of agents, from the standpoint of (D)AI. For convenience, we identify three key issues, and structure our survey around these (cf. [117, p1]):

Agent theories: What exactly are agents? What properties should they have, and how are we to formally represent and reason about these properties?

Agent architectures: How are we to construct agents that satisfy the properties we expect of them? What software and/or hardware structures are appropriate?

Agent languages: How are we to program agents? What are the right primitives for this task? How are we to effectively compile or execute agent programs?

We begin, in the following section, with the issue of agent theories, and a consideration of the question of how to define agency; in section 3, we discuss architectures, and in section 4, we discuss languages for programming agents. Some concluding remarks appear in section 5.

2 Agent Theories

As we observed above, there are many different usages of the term *agent* in AI and computer science, and each yet each of these usages appeals to a subtly different notion of agency. An obvious point of departure for our study is therefore a consideration of the question: *what is an agent?*

A dictionary defines an agent as: 'one who, or that which, exerts power or produces an effect'[2]. While this definition is not terribly helpful, it does at least indicate that *action* is somehow involved, and indeed it does seem at first sight that the notion of action is inextricably bound to that of agency:

> 'Agents do things, they *act*: that is why they are called agents'. [118]

A tacit assumption is that agents take an active role, originating actions that affect their environment, rather than passively allowing their environment to affect them. Two terms often used to describe agentive action are *autonomy* and *rationality*. Autonomy generally means that an agent operates without direct human (or other) intervention or guidance. Rationality is not so easily tied down, but is often used in the pseudo-game-theoretic sense of an agent maximizing its performance with respect to some 'valuation function' (see [48, pp49–54] for a discussion of rationality and agency).

Unfortunately, autonomous rational action, so defined, is a weak criterion for agenthood, as it admits a very wide class of objects as agents. For example, it is perfectly consistent to describe a transistor — essentially the simplest form of electronic switch — as an autonomous rational agent by this definition.

Perhaps more troubling for an action-based analysis of agency is that the very notion of action is a slippery one. For example, almost any action can be described in a number of different ways, each seemingly valid. A classic example, due to the philosopher Searle, is that of Gavrilo Princip in 1914: did he pull a trigger, fire a gun, kill Archduke Ferdinand, or start World War I? Each of these seem to be equally valid descriptions of the same action or event. Trying to describe actions in terms of causal links does not help, as it introduces a seemingly infinite regress. For example, in waving to a friend, I lift my arm, which was caused by muscles contracting, which was caused by some neurons firing, which was caused by… and so on. There is no easy way of halting this regress without appealing to a notion of primitive action, which is philosophically suspect[3].

[2] *The Concise Oxford Dictionary of Current English (7th edn)*, Oxford University Press, 1988

[3] See [4] for a classic AI attempt to deal with the notion of action, and [118] for an analysis of the relationship between action and agency.

An action-based analysis of agency does not look like it is going to work. What other properties of agency might one consider? Shoham has suggested that the term 'agent' in AI is often used to denote 'high-level' systems, that employ symbolic representations, and perhaps enjoy some 'cognitive-like' function, (such as explicit logical reasoning) [121]. This 'high-level' condition excludes systems such as transistors and thermostats, the neuron-like entities of connectionism, and the objects of object-oriented programming. It implies that agents possess significant computational resources (though these resources will, of course, be finite). However, the 'high-level' property is a contentious one: a number of researchers vigorously argue that 'high-level' agents are not the best way to go about AI. The chief protagonist in this debate is Brooks, who has built a number of robotic agents which are certainly not 'high-level' by Shoham's definition, and yet are able to perform tasks that are impressive by AI standards (see the discussion in section 4). So a 'high-level' condition does not seem to be useful for classifying agents, as it discriminates against systems that do not employ explicit cognitive-like functions.

Perhaps the most widely held view is that an agent is an entity 'which appears to be the subject of beliefs, desires, etc.' [117, p1]. The philosopher Dennett has coined the term *intentional system* to denote such systems.

2.1 Agents as Intentional Systems

When explaining human activity, it is often useful to make statements such as the following:

> Janine took her umbrella because she *believed* it was going to rain.
> Michael worked hard because he *wanted* to possess a PhD.

These statements makes use of a *folk psychology*, by which human behaviour is predicted and explained through the attribution of *attitudes*, such as believing and wanting (as in the above examples), hoping, fearing, and so on. This folk psychology is well established: most people reading the above statements would say they found their meaning entirely clear, and would not give them a second glance.

The attitudes employed in such folk psychological descriptions are called the *intentional* notions. The philosopher Daniel Dennett has coined the term *intentional system* to describe entities 'whose behaviour can be predicted by the method of attributing belief, desires and rational acumen' [35, p49]. Dennett identifies different 'grades' of intentional system:

> 'A *first-order* intentional system has beliefs and desires (etc.) but no beliefs and desires *about* beliefs and desires. [...] A *second-order* intentional system is more sophisticated; it has beliefs and desires (and no doubt other intentional states) about beliefs and desires (and other intentional states) — both those of others and its own'. [35, p243]

One can carry on this hierarchy of intentionality as far as required.

An obvious question is whether it is legitimate or useful to attribute beliefs, desires, and so on, to artificial agents. Isn't this just anthropomorphism? McCarthy, among others, has argued that there are occasions when the *intentional stance* is appropriate:

> 'To ascribe *beliefs*, *free will*, *intentions*, *consciousness*, *abilities*, or *wants* to a machine is <u>legitimate</u> when such an ascription expresses the same information about the machine that it expresses about a person. It is <u>useful</u> when the ascription helps us understand the structure of the machine, its past or future behaviour, or how to repair or improve it. It is perhaps never <u>logically required</u> even for humans, but expressing reasonably briefly what is actually known about the state of the machine in a particular situation may require mental qualities or qualities isomorphic to them. Theories of belief, knowledge and wanting can be constructed for machines in a simpler setting than for humans, and later applied to humans. Ascription of mental qualities is <u>most straightforward</u> for machines of known structure such as thermostats and computer operating systems, but is <u>most useful</u> when applied to entities whose structure is incompletely known'. [93], (quoted in [121])

What objects can be described by the intentional stance? As it turns out, more or less anything can. In his doctoral thesis, Seel showed that even very simple, automata-like objects can be consistently ascribed intentional descriptions [117]; similar work by Rosenschein and Kaelbling, (albeit with a different motivation), arrived at a similar conclusion [111]. For example, consider a light switch:

> 'It is perfectly coherent to treat a light switch as a (very cooperative) agent with the capability of transmitting current at will, who invariably transmits current when it believes that we want it transmitted and not otherwise; flicking the switch is simply our way of communicating our desires'. [121, p6]

And yet most adults would find such a description absurd — perhaps even infantile. Why is this? The answer seems to be that while the intentional stance description is perfectly consistent with the observed behaviour of a light switch, and is internally consistent,

> '… it does not *buy us anything*, since we essentially understand the mechanism sufficiently to have a simpler, mechanistic description of its behaviour'. [121, p6]

Put crudely, the more we know about a system, the less we need to rely on animistic, intentional explanations of its behaviour. However, with very complex systems, even if a complete, accurate picture of the system's architecture and working *is* available, a mechanistic, *design stance* explanation of its behaviour may not be practicable. Consider a computer. Although we might have a complete technical description of a computer available, it is hardly practicable to appeal to such a description when explaining why a menu appears when we click a mouse on an icon. In such situations, it may be more appropriate to adopt an intentional stance description, if that description is consistent, and simpler than the alternatives. The intentional notions are thus *abstraction tools*, which provide us with a convenient and familiar way of describing, explaining, and predicting the behaviour of complex systems.

Being an intentional system seems to be a *necessary* condition for agenthood, but is it a *sufficient* condition? In his recent Master's thesis, Shardlow trawled through the literature of cognitive science and its component disciplines in an attempt to find a unifying concept that underlies the notion of agenthood. He was forced to the following conclusion:

> 'Perhaps there is something more to an agent than its capacity for beliefs and desires, but whatever that thing is, it admits no unified account within cognitive science'. [118]

So, an agent is a system that is most conveniently described by the intentional stance; one whose simplest consistent description requires the intentional stance. Before proceeding, it is worth considering exactly which attitudes are appropriate for representing agents. For the purposes of this survey, the two most important categories are *information attitudes* and *pro-attitudes*:

$$
\begin{array}{rl}
\text{information attitudes} & \begin{cases} \text{belief} \\ \text{knowledge} \end{cases} \\
\text{pro-attitudes} & \begin{cases} \text{desire} \\ \text{intention} \\ \text{obligation} \\ \text{commitment} \\ \text{choice} \\ \ldots \end{cases}
\end{array}
$$

Thus information attitudes are related to the information that an agent has about the world it occupies, whereas pro-attitudes are those that in some way guide the agent's actions. Precisely which *combination* of attitudes is most appropriate to characterise an agent is, as we shall see later, an issue of some debate. However, it seems reasonable to suggest that an agent must be represented in terms of at least one information attitude, and at least one pro-attitude. Note that pro- and information attitudes are closely linked, as a rational agent will make make choices and form intentions, etc., on the basis of the information it has about the world. Much work in agent theory is concerned with sorting out exactly what the relationship between the different attitudes is.

The next step is to investigate methods for representing and reasoning about these intentional notions.

2.2 Representing Intentional Notions

Suppose one wishes to reason about intentional notions in a logical framework. Consider the following statement (after [50, pp210–211]):

Janine believes Cronos is the father of Zeus. (1)

A naive attempt to translate (1) into first-order logic might result in the following:

$$Bel(Janine, Father(Zeus, Cronos)) \tag{2}$$

Unfortunately, this naive translation does not work, for at least two reasons. The first is syntactic: the second argument to the *Bel* predicate is a *formula* of first-order logic, and is not, therefore, a term. So (2) is not a well-formed formula of classical first-order logic. The second problem is semantic, and is more serious. The constants *Zeus* and *Jupiter*, by any reasonable interpretation, denote the same individual: the supreme deity of the classical world. It is therefore acceptable to write, in first-order logic:

$$(Zeus = Jupiter). \tag{3}$$

Given (2) and (3), the standard rules of first-order logic would allow the derivation of the following:

$$Bel(Janine, Father(Jupiter, Cronos)) \tag{4}$$

But intuition rejects this derivation as invalid: believing that the father of Zeus is Cronos is *not* the same as believing that the father of Jupiter is Cronos. So what is the problem? Why does first-order logic fail here? The problem is that the intentional notions — such as belief and desire — are *referentially opaque*, in that they set up *opaque contexts*, in which the standard substitution rules of first-order logic do not apply. In classical (propositional or first-order) logic, the denotation, or semantic value, of an expression is dependent solely on the denotations of its sub-expressions. For example, the denotation of the propositional logic formula $p \wedge q$ is a function of the truth-values of p and q. The operators of classical logic are thus said to be *truth functional*. In contrast, intentional notions such as belief are *not* truth functional. It is surely not the case that the truth value of the sentence:

Janine believes p (5)

is dependent solely on the truth-value of p[4]. So substituting equivalents into opaque contexts is not going to preserve meaning. This is what is meant by referential opacity. The existence of referentially opaque contexts has been known since the time of Frege. He suggested a distinction between *sense* and *reference*. In ordinary formulae, the 'reference' of a term/formula (i.e., its denotation) is needed, whereas in opaque contexts, the 'sense' of a formula is needed (see also [117, p3]).

Clearly, classical logics are not suitable in their standard form for reasoning about intentional notions: alternative formalisms are required.

The field of formal methods for reasoning about intentional notions is widely reckoned to have begun with the publication, in 1962, of Hintikka's book *Knowledge and Belief* [71]. At that time, the

[4]Note, however, that the sentence (5) is itself a proposition, in that its denotation is the value true or false.

subject was of interest to comparatively few researchers in logic and the philosophy of mind. Since then, however, it has become an important research area in its own right, with contributions from researchers in AI, formal philosophy, linguistics and economics. Despite the diversity of interests and applications, the number of basic techniques in use is quite small. Recall, from the discussion above, that there are two problems to be addressed in developing a logical formalism for intentional notions: a syntactic one, and a semantic one. It follows that any formalism can be characterized in terms of two independent attributes: its *language of formulation*, and *semantic model* [80, p83].

There are two fundamental approaches to the syntactic problem. The first is to use a *modal* language, which contains non-truth-functional *modal operators*, which are applied to formulae. An alternative approach involves the use of a *meta-language*: a many-sorted first-order language containing terms that denote formulae of some other *object-language*. Intentional notions can be represented using a meta-language predicate, and given whatever axiomatization is deemed appropriate. Both of these approaches have their advantages and disadvantages, and will be discussed in the sequel.

As with the syntactic problem, there are two basic approaches to the semantic problem. The first, best known, and probably most widely used approach is to adopt a *possible worlds* semantics, where an agent's beliefs, knowledge, goals, etc. are characterized as a set of so-called *possible worlds*, with an *accessibility relation* holding between them. Possible worlds semantics have an associated *correspondence theory* which makes them an attractive mathematical tool to work with [26]. However, they also have many associated difficulties, notably the well-known *logical omniscience* problem, which implies that agents are perfect reasoners. A number of variations on the possible-worlds theme have been proposed, in an attempt to retain the correspondence theory, but without logical omniscience. The commonest alternative to the possible worlds model for belief is to use a *sentential*, or *interpreted symbolic structures* approach. In this scheme, beliefs are viewed as symbolic formulae explicitly represented in a data structure associated with an agent. An agent then believes φ if φ is present in its belief data structure. Despite its simplicity, the sentential model works well under certain circumstances [80].

In the subsections that follow, we discuss various approaches in some more detail. We begin with a close look at the basic possible worlds model for logics of knowledge (*epistemic* logics) and logics of belief (*doxastic* logics).

Possible Worlds Semantics

The possible worlds model for logics of knowledge and belief was originally proposed by Hintikka [71], and is now most commonly formulated in a normal modal logic using the techniques developed by Kripke [84][5]. Hintikka's insight was to see that an agent's beliefs could be characterized in terms of a set of *possible worlds*, in the following way. Consider an agent playing a card game such as poker[6]. In this game, the more one knows about the cards possessed by one's opponents, the better one is able to play. And yet complete knowledge of an opponent's cards is generally impossible, (if one excludes cheating). The ability to play poker well thus depends, at least in part, on the ability to deduce what cards are held by an opponent, given the limited information available. Now suppose our agent possessed the ace of spades. Assuming the agent's sensory equipment was functioning normally, it would be rational of her to believe that she possessed this card. Now suppose she were to try to deduce what cards were held by her opponents. This could be done by first calculating all the various different ways that the cards in the pack could possibly have been distributed among the various players. (This is not being proposed as an actual card playing strategy, but for illustration!) For argument's sake, suppose that each possible configuration is described on a separate piece of paper. Once the process was complete, our agent can then begin to systematically eliminate from this large pile of paper all those configurations which are *not possible, given what she knows*. For example, any configuration in which she did not possess the

[5] In Hintikka's original work, he used a technique based on 'model sets', which is equivalent to Kripke's formalism, though less elegant. See [72, pp351–352] for a comparison and discussion of the two techniques.

[6] This example was adapted from [64].

ace of spades could be rejected immediately as impossible. Call each piece of paper remaining after this process a *world*. Each world represents one state of affairs considered possible, given what she knows. Hintikka coined the term *epistemic alternatives* to describe the worlds possible given one's beliefs. Something true in *all* our agent's epistemic alternatives could be said to be believed by the agent. For example, it will be true in all our agent's epistemic alternatives that she has the ace of spades.

On a first reading, this seems a peculiarly roundabout way of characterizing belief, but it has two advantages. First, it remains neutral on the subject of the cognitive structure of agents. It certainly doesn't posit any internalized collection of possible worlds. It is just a convenient way of characterizing belief. Second, the mathematical theory associated with the formalization of possible worlds is extremely appealing (see below).

The next step is to show how possible worlds may be incorporated into the semantic framework of a logic. Epistemic logics are usually formulated as *normal modal logics* using the semantics developed by Kripke [84]. Before moving on to explicitly epistemic logics, we consider a simple normal modal logic. This logic is essentially classical propositional logic, extended by the addition of two operators: '$\Box$' (necessarily), and '$\Diamond$' (possibly). Let $Prop = \{p, q, \ldots\}$ be a countable set of *atomic propositions*. Then the syntax of the logic is defined by the following rules: (i) if $p \in Prop$ then p is a formula; (ii) if φ, ψ are formulae, then so are $\neg\varphi$ and $\varphi \vee \psi$; and (iii) if φ is a formula then so are $\Box\varphi$ and $\Diamond\varphi$.

The operators '$\neg$' (not) and '$\vee$' (or) have their standard meanings. The remaining connectives of classical propositional logic can be defined as abbreviations in the usual way. The formula $\Box\varphi$ is read: 'necessarily φ', and the formula $\Diamond\varphi$ is read: 'possibly φ'. Now to the semantics of the language.

Normal modal logics are concerned with truth at worlds; models for such logics therefore contain a set of worlds, W, and a binary relation, R, on W, saying which worlds are considered possible relative to other worlds. Additionally, a valuation function π is required, saying what propositions are true at each world. Formally, a model is a triple $\langle W, R, \pi\rangle$, where W is a non-empty set of worlds, $R \subseteq W \times W$, and $\pi : W \rightarrow \mathsf{powerset}\ Prop$ is a valuation function, which says for each world $w \in W$ which atomic propositions are true in w. An alternative, equivalent technique would have been to define π as $\pi : W \times Prop \rightarrow \{T, F\}$.

The semantics of the language are given via the satisfaction relation, '$\models$', which holds between pairs of the form $\langle M, w\rangle$, (where M is a model, and w is a reference world), and formulae of the language. The semantic rules defining this relation are given below.

$$
\begin{array}{lcll}
\langle M, w\rangle & \models & p & \text{where } p \in Prop, \text{ iff } p \in \pi(w) \\
\langle M, w\rangle & \models & \neg\varphi & \text{iff } \langle M, w\rangle \not\models \varphi \\
\langle M, w\rangle & \models & \varphi \vee \psi & \text{iff } \langle M, w\rangle \models \varphi \text{ or } \langle M, w\rangle \models \psi \\
\langle M, w\rangle & \models & \Box\varphi & \text{iff } \forall w' \in W, \text{ if } (w, w') \in R \text{ then } \langle M, w'\rangle \models \varphi \\
\langle M, w\rangle & \models & \Diamond\varphi & \text{iff } \exists w' \in W, (w, w') \in R \text{ and } \langle M, w'\rangle \models \varphi
\end{array}
$$

The definition of satisfaction for atomic propositions thus captures the idea of truth in the 'current' world, (which appears on the left of '$\models$'). The semantic rules for '$\neg$' and '$\vee$' are standard. The rule for '$\Box$' captures the idea of truth in all accessible worlds, and the rule for '$\Diamond$' captures the idea of truth in at least one possible world. Note that the two modal operators are *duals* of each other, in the sense that the universal and existential quantifiers of first-order logic are duals:

$$\Box\varphi \Leftrightarrow \neg\Diamond\neg\varphi \qquad \Diamond\varphi \Leftrightarrow \neg\Box\neg\varphi.$$

It would thus have been possible to take either one as primitive, and introduce the other as a derived operator.

Correspondence Theory

To understand the extraordinary properties of this simple logic, it is first necessary to introduce *validity* and *satisfiability*. A formula is *satisfiable* if it is satisfied for some model/world pair, and *unsatisfiable* otherwise. A formula is *true in a model* if it is satisfied for every world in the model, and *valid in a class of models* if it true in every model in the class. Finally, a formula is valid *simpliciter* if it is true in the class of all models. If φ is valid, we write $\models \varphi$.

The two basic properties of this logic are as follows. First, the following axiom schema is valid.

$$\models \Box(\varphi \Rightarrow \psi) \Rightarrow (\Box\varphi \Rightarrow \Box\psi)$$

This axiom is called K, in honour of Kripke. The second property is as follows.

$$\text{If } \models \varphi \text{ then } \models \Box\varphi$$

Proofs of these properties are left as an exercise for the reader. Now, since K is valid, it will be a theorem of any complete axiomatization of normal modal logic. Similarly, the second property will appear as a rule of inference in any axiomatization of normal modal logic; it is generally called the *necessitation* rule. These two properties turn out to be the most problematic features of normal modal logics when they are used as logics of knowledge/belief (this point will be examined later).

The most intriguing properties of normal modal logics follow from the properties of the accessibility relation, R, in models. To illustrate these properties, consider the following axiom schema.

$$\Box\varphi \Rightarrow \varphi$$

It turns out that this axiom is *characteristic* of the class of models with a *reflexive* accessibility relation. (By characteristic, we mean that it is true in all and only those models in the class.) There are a host of axioms which correspond to certain properties of R: the study of the way that properties of R correspond to axioms is called *correspondence theory*. For our present purposes, we identify just four axioms: the axiom called T, (which corresponds to a reflexive accessibility relation); D (serial accessibility relation); 4 (transitive accessibility relation); and 5 (euclidean accessibility relation):

$$\begin{array}{ll} \text{T} & \Box\varphi \Rightarrow \varphi \\ \text{D} & \Box\varphi \Rightarrow \Diamond\varphi \\ 4 & \Box\varphi \Rightarrow \Box\Box\varphi \\ 5 & \Diamond\varphi \Rightarrow \Box\Diamond\varphi. \end{array}$$

The results of correspondence theory make it straightforward to derive completeness results for a range of simple normal modal logics. These results provide a useful point of comparison for normal modal logics, and account in a large part for the popularity of this style of semantics. A *system of logic* can be thought of as a set of formulae valid in some class of models; a member of the set is called a *theorem* of the logic (if φ is a theorem, this is usually denoted by $\vdash \varphi$). The notation $\mathrm{K}\Sigma_1 \ldots \Sigma_n$ is often used to denote the smallest normal modal logic containing axioms $\Sigma_1, \ldots, \Sigma_n$ (recall that any normal modal logic will contain the K axiom [58, p25]).

For the axioms T, D, 4, and 5, it would seem that there ought to be sixteen distinct systems of logic (since $2^4 = 16$). However, some of these systems turn out to be equivalent (in that they contain the same theorems), and as a result there are only eleven distinct systems: K, K4, K5, KD, KT (= KDT), K45, KD5, KD4, KT4 (=KDT4), KD45, and KT5 (= KT45, KDT5, KDT45); see [80, p99], and [26, p132]. Because some modal systems are so widely used, they have been given names:

KT	is known as	T	KT4	is known as	S4
KD45	is known as	weak-S5	KT5	is known as	S5.

*

Normal Modal Logics of Knowledge and Belief

To use the logic developed above as an epistemic logic, the formula $\Box\varphi$ is read as: 'it is known that φ'. The worlds in the model are interpreted as epistemic alternatives, the accessibility relation defines what the alternatives are from any given world. The logic deals with the knowledge of a single agent. To deal with multi-agent knowledge, one adds to a model structure an indexed set of accessibility relations, one for each agent. A model is then a structure $\langle W, R_1, \ldots, R_n, \pi \rangle$ where R_i is the knowledge accessibility relation of agent i. The simple language defined above is extended by replacing the single modal operator '$\Box$' by an indexed set of unary modal operators $\{K_i\}$, where $i \in \{1, \ldots, n\}$. The formula $K_i\varphi$ is read: 'i knows that φ'. The semantic rule for '$\Box$' is replaced by the following rule:

$$\langle M, w \rangle \models K_i\varphi \quad \text{iff } \forall w' \in W, \text{ if } (w, w') \in R_i \text{ then } \langle M, w' \rangle \models \varphi$$

Each operator K_i thus has exactly the same properties as '$\Box$'. Corresponding to each of the modal systems Σ, above, a corresponding system Σ_n is defined, for the multi-agent logic. Thus K_n is the smallest multi-agent epistemic logic and $S5_n$ is the largest.

The next step is to consider how well normal modal logic serves as a logic of knowledge/belief. Consider first the necessitation rule and axiom K, since any normal modal system is committed to these. The necessitation rule tells us that an agent knows all valid formulae. Amongst other things, this means an agent knows all propositional tautologies. Since there are an infinite number of these, an agent will have an infinite number of items of knowledge: immediately, one is faced with a counter-intuitive property of the knowledge operator.

Now consider the axiom K, which says that an agent's knowledge is closed under implication. Suppose φ is a logical consequence of the set $\Phi = \{\varphi_1, \ldots, \varphi_n\}$, then in every world where all of Φ are true, φ must also be true, and hence the formula $\varphi_1 \wedge \cdots \wedge \varphi_n \Rightarrow \varphi$ must be valid. By necessitation, this formula will also be believed. Since an agent's beliefs are closed under implication, whenever it believes each of Φ, it must also believe φ. Hence an agent's knowledge is closed under logical consequence. This also seems counter intuitive. For example, suppose, like every good logician, our agent knows Peano's axioms. Now Fermat's last theorem follows from Peano's axioms — but it took the combined efforts of some of the best minds over the past century to prove it. Yet if our agent's beliefs are closed under logical consequence, then our agent must know it. So consequential closure, implied by necessitation and the K axiom, seems an overstrong property for resource bounded reasoners.

These two problems — that of knowing all valid formulae, and that of knowledge/belief being closed under logical consequence — together constitute the famous *logical omniscience* problem. It has been widely argued that this problem makes the possible worlds model unsuitable for representing resource bounded believers — and any real system is resource bounded.

Axioms for Knowledge and Belief

We now consider the appropriateness of the axioms D_n, T_n, 4_n, and 5_n for logics of knowledge/belief. The axiom D_n says that an agent's beliefs are non-contradictory; it can be re-written in the following form:

$$K_i\varphi \Rightarrow \neg K_i \neg\varphi$$

which is read: 'if i knows φ, then i doesn't know $\neg\varphi$'. This axiom seems a reasonable property of knowledge/belief.

The axiom T_n is often called the *knowledge* axiom, since it says that what is known is true. It is usually accepted as the axiom that distinguishes knowledge from belief: it seems reasonable that one could believe something that is false, but one would hesitate to say that one could *know* something false.

Knowledge is thus often defined as true belief: i knows φ if i believes φ and φ is true. So defined, knowledge satisfies T_n.

Axiom 4_n is called the *positive introspection axiom*. Introspection is the process of examining one's own beliefs, and is discussed in detail in [80, Chapter 5]. The positive introspection axiom says that an agent knows what it knows. Similarly, axiom 5_n is the *negative introspection axiom*, which says that an agent is aware of what it doesn't know. Positive and negative introspection together imply an agent has perfect knowledge about what it does and doesn't know (cf. [80, Equation (5.11), p79]). Whether or not the two types of introspection are appropriate properties for knowledge/belief is the subject of some debate. However, it is generally accepted that positive introspection is a less demanding property than negative introspection, and is thus a more reasonable property for resource bounded reasoners.

Given the comments above, the modal system $S5_n$ is often chosen as a logic of (idealised) *knowledge*, and weak-$S5_n$ is often chosen as a logic of (idealised) *belief*.

Alternatives to the Possible Worlds Model

As a result of the difficulties with logical omniscience, many researchers have attempted to develop alternative formalisms for representing belief. Some of these are attempts to adapt the basic possible worlds model; others represent significant departures from it. In the subsections that follow, we examine some of these attempts.

Levesque — belief and awareness

In a 1984 paper, Levesque proposed a solution to the logical omniscience problem that involves making a distinction between *explicit* and *implicit* belief [87]. Crudely, the idea is that an agent has a relatively small set of explicit beliefs, and a very much larger (infinite) set of implicit beliefs, which include the logical consequences of the explicit beliefs. To formalise this idea, Levesque developed a logic with two operators; one each for implicit and explicit belief. The semantics of the explicit belief operator were given in terms of a weakened possible worlds semantics, by borrowing some ideas from situation semantics [10, 37]. The semantics of the implicit belief operator were given in terms of a standard possible worlds approach. A number of objections have been raised to Levesque's model [109, p135]: first, it does not allow quantification — this drawback has been rectified by Lakemeyer [85]; second, it does not seem to allow for nested beliefs; third, the notion of a situation, which underlies Levesque's logic is, if anything, more mysterious than the notion of a world in possible worlds; and fourth, under certain circumstances, Levesque's proposal still makes unrealistic predictions about agent's reasoning capabilities.

In an effort to recover from this last negative result, Fagin and Halpern have developed a 'logic of general awareness', based on a similar idea to Levesque's but with a very much simpler semantics [40]. However, this proposal has itself been criticised by some [81].

Konolige — the deduction model

A more radical approach to modelling resource bounded believers was proposed by Konolige [80]. His *deduction model of belief* is, in essence, a direct attempt to model the 'beliefs' of symbolic AI systems. Konolige observed that a typical knowledge-based system has two key components: a database of symbolically represented 'beliefs', (which may take the form of rules, frames, semantic nets, or, more generally, formulae in some logical language), and some logically incomplete inference mechanism. Konolige modelled such systems in terms of *deduction structures*. A deduction structure is a pair $d = (\Delta, \rho)$, where Δ is a base set of formula in some logical language, and ρ is a set of inference rules, (which may be logically incomplete), representing the agent's reasoning mechanism. To simplify the formalism, Konolige assumed that an agent would apply its inference rules wherever possible, in order to generate the *deductive closure* of its base beliefs under its deduction rules. We model deductive closure in a function *close*:

$$close((\Delta, \rho)) \stackrel{\text{def}}{=} \{\varphi \mid \Delta \vdash_\rho \varphi\}$$

where $\Delta \vdash_\rho \varphi$ means that φ can be proved from Δ using only the rules in ρ. A belief logic can then be defined, with the semantics to a modal belief connective $[i]$, where i is an agent, given in terms of the deduction structure d_i modelling i's belief system:

$$[i]\varphi \qquad \text{iff} \qquad \varphi \in close(d_i).$$

Konolige went on to examine the properties of the deduction model at some length, and developed a variety of proof methods for his logics, including resolution and tableau systems [49]. The deduction model is undoubtedly simple; some might even argue that it is naive. However, as a direct model of the belief systems of AI agents, it has much to commend it.

Meta-languages and syntactic modalities

A meta-language is one in which it is possible to represent the properties of another language. A first-order meta-language is a first-order logic, with the standard predicates, quantifiers, terms, and so on, whose domain contains formulae of some other language, called the *object* language. Using a meta-language, it is possible to represent a relationship between a meta-language term denoting an agent, and an object language term denoting some formula. For example, the meta-language formula $Bel(Janine, \ulcorner Father(Zeus, Cronos)\urcorner)$ might be used to represent the example (1) the we saw earlier. The quote marks, $\ulcorner \ldots \urcorner$, are used to indicate that their contents are a meta-language term denoting the corresponding object-language formula.

Unfortunately, meta-language formalisms have their own package of problems, not the least of which is that they tend to fall prey to inconsistency [95, 132]. However, there have been some fairly successful meta-language formalisms, including those by Konolige [79], Haas [61], Morgenstern [97], and Davies [32]. Some results on retrieving consistency appeared in the late 1980s [101, 102, 36, 133].

2.3 Towards a Theory of Agency

All of the formalisms considered so far have focussed on just one aspect of intelligent agency: either knowledge or belief. However, it is to be expected that any realistic *agent theory* will be represented in a much richer logical framework. First, neither agents nor the world they inhabit are static. In addition to the information and pro-attitudes we mentioned earlier, an agent logic must therefore be capable of representing the time-varying aspects of agents and their world. Moreover, although we suggested earlier that action was a somewhat slippery concept, we shall ultimately expect our agents to *do* things; some representation of action is therefore desirable.

A complete agent theory, expressed in a logic with these properties, must show how these attributes are related. For example, it will need to explain how an agent's information and pro-attitudes are related; how an agent's cognitive state changes over time; how the environment affects an agent's cognitive state; and how an agent's information and pro-attitudes lead it to perform actions. Giving a good account of these relationships is perhaps the most significant problem faced by agent theorists.

Such an all-embracing agent theory is some time off, and yet significant steps have been taken towards it. In the following subsections, we briefly review some of this work.

Moore — knowledge and action

Moore was in many ways a pioneer of the use of logics for capturing aspects of agency [96]. His main concern was the study of *knowledge pre-conditions for actions* — the question of what an agent needs to know in order to be able to perform some action. He formalised a model of *ability* in a logic containing a modality for knowledge, and a dynamic logic-like apparatus for modelling action (cf. [67]). This formalism allowed for the possibility of an agent having incomplete information about how to achieve some goal, and performing actions in order to find out how to achieve it. Critiques of the formalism (and attempts to improve on it) may be found in [97, 86].

Cohen & Levesque — intention

Probably the best-known and most influential contribution to the area of agent theory is due to Cohen and Levesque [28]. Their formalism was originally used to develop a theory of intention (as in 'I intend to…'), which the authors required as a pre-requisite for a theory of speech acts [29]. However, the logic has subsequently proved to be so useful for reasoning about agents that it has been used in an analysis of conflict and cooperation in multi-agent dialogue [48, 47], as well as several studies in the theoretical foundations of cooperative problem solving [88, 73, 21, 22]. Here, we shall review its use in developing a theory of intention.

When building intelligent agents — particularly agents that must interact with humans — it is important that a *rational balance* is achieved between the beliefs and goals of the agents:

> 'For example, the following are desirable properties of intention: An autonomous agent should act on its intentions, not in spite of them; adopt intentions it believes are feasible and forgo those believed to be infeasible; keep (or commit to) intentions, but not forever; discharge those intentions believed to have been satisfied; alter intentions when relevant beliefs change; and adopt subsidiary intentions during plan formation'. [28, p214]

Following Bratman, [14, 15], Cohen and Levesque identify seven properties that must be satisfied by a reasonable theory of intention:

1. Intentions pose problems for agents, who need to determine ways of achieving them.
2. Intentions provide a 'filter' for adopting other intentions, which must not conflict.
3. Agents track the success of their intentions, and are inclined to try again if their attempts fail.
4. Agents believe their intentions are possible.
5. Agents do not believe they will not bring about their intentions.
6. Under certain circumstances, agents believe they will bring about their intentions.
7. Agents need not intend all the expected side effects of their intentions.

Given these criteria, Cohen and Levesque adopt a two-tiered approach to the problem of formalizing a theory of intention. First, they construct a *logic of rational agency*, 'being careful to sort out the relationships among the basic modal operators' [28, p221]. On top of this framework, they introduce a number of derived constructs, which constitute a 'partial theory of rational action' [28, p221]; intention is one of these constructs. Syntactically, the logic is a many-sorted, quantified, multi-modal logic with equality, containing four primary modalities:

$(\mathsf{BEL}\ x\ \varphi)$	Agent x believes φ	$(\mathsf{GOAL}\ x\ \varphi)$	Agent x has goal of φ
$(\mathsf{HAPPENS}\ \alpha)$	Action α will happen next	$(\mathsf{DONE}\ \alpha)$	Action α has just happened

The semantics of BEL and GOAL are given via possible worlds, in the usual way: each agent is assigned a belief accessibility relation, and a goal accessibility relation. The belief accessibility relation is euclidean, transitive, and serial, giving a belief logic of KD45. The goal relation is serial, giving a conative logic KD. It is assumed that each agent's goal relation is a subset of its belief relation, implying that an agent will not have a goal of something it believes will not happen. Worlds in the formalism are a discrete sequence of events, stretching infinitely into past and future.

The two basic temporal operators, HAPPENS and DONE, are augmented by some operators for describing the structure of event sequences, in the style of dynamic logic [67]. The two most important of these constructors are ';' and '?':

$$\begin{array}{rl} \alpha;\alpha' & \text{denotes } \alpha \text{ followed by } \alpha' \\ \alpha? & \text{denotes a `test action' } \alpha \end{array}$$

The standard future time operators of temporal logic, '$\Box$' (always), and '$\Diamond$' (sometime) can be defined as abbreviations, along with a 'strict' sometime operator, LATER:

$$\Diamond\alpha \stackrel{\text{def}}{=} \exists x \cdot (\mathsf{HAPPENS}\ x; \alpha?) \quad \Box\alpha \stackrel{\text{def}}{=} \neg\Diamond\neg\alpha \quad (\mathsf{LATER}\ p) \stackrel{\text{def}}{=} \neg p \wedge \Diamond p$$

A temporal precedence operator, (BEFORE p q) can also be derived, and holds if p holds before q. An important assumption is that *all* goals are eventually dropped: $\Diamond\neg(\mathsf{GOAL}\ x\ (\mathsf{LATER}\ p))$.

The first major derived construct is a *persistent* goal.

$$(\mathsf{P\text{-}GOAL}\ x\ p) \stackrel{\text{def}}{=} \begin{array}{l} (\mathsf{GOAL}\ x\ (\mathsf{LATER}\ p)) \wedge (\mathsf{BEL}\ x\ \neg p) \quad \wedge \\ \left[\begin{array}{l} \mathsf{BEFORE} \\ \quad ((\mathsf{BEL}\ x\ p) \vee (\mathsf{BEL}\ x\ \Box\neg p)) \\ \quad \neg(\mathsf{GOAL}\ x\ (\mathsf{LATER}\ p)) \end{array}\right] \end{array}$$

So, an agent has a persistent goal of p if:

1. It has a goal that p eventually becomes true, and believes that p is not currently true.

2. Before it drops the goal, one of the following conditions must hold: (i) the agent believes the goal has been satisfied; or (ii) the agent believes the goal will never be satisfied.

It is a small step from persistent goals to a first definition of intention, as in 'intending to act'. Note that 'intending that something becomes true' is similar, but requires a slightly different definition; see [28].

$$(\mathsf{INTEND}\ x\ \alpha) \stackrel{\text{def}}{=} (\mathsf{P\text{-}GOAL}\ x\ [\mathsf{DONE}\ x\ (\mathsf{BEL}\ x\ (\mathsf{HAPPENS}\ \alpha))?; \alpha])$$

Cohen and Levesque go on to show how such a definition meets many of Bratman's criteria for a theory of intention (outlined above). A critique of Cohen and Levesque's theory of intention may be found in [126]; space restrictions prevent a discussion here.

Rao & Georgeff — belief, desire, intention architectures

As we observed earlier, there is no clear consensus in either the AI or philosophy communities about precisely which combination of information and pro-attitudes are best suited to characterising rational agents. In the work of Cohen and Levesque, described above, just two basic attitudes were used: beliefs and goals. Further attitudes, such as intention, were defined in terms of these. In related work, Rao and Georgeff have developed a logical framework for agent theory based on three primitive modalities: beliefs, desires, and intentions [105, 104, 107]. Their formalism is based on a branching model of time, (cf. [39]), in which belief-, desire- and intention-accessible worlds are themselves branching time structures. They are particularly concerned with the notion of *realism* — the question of how an agent's beliefs about the future affect its desires and intentions. In other work, they also consider the potential for adding (social) plans to their formalism [106, 77].

Singh

A quite different approach to modelling agents was taken by Singh, who has developed an interesting family of logics for representing intentions, beliefs, knowledge, know-how, and communication in a branching-time framework [123, 124, 127, 125]. The model of intentions and beliefs is based on Asher-Kamp Discourse Representation Theory. Singh's formalism is extremely rich, and considerable effort has been devoted to establishing its properties. However, its complexity prevents a detailed discussion here.

Werner

In an extensive sequence of papers, Werner has laid the foundations of a general model of agency, which draws upon work in economics, game theory, situated automata theory, situation semantics, and philosophy [136, 137, 138, 139]. At the time of writing, however, the properties of this model have not been investigated in depth.

Wooldridge — modelling multi-agent systems

For his 1992 doctoral thesis, Wooldridge developed a family of logics for representing the properties of multi-agent systems [143, 145]. Unlike the approaches cited above, Wooldridge's aim was not to develop a general framework for agent theory. Rather, he hoped to construct formalisms that might be used in the specification and verification of realistic multi-agent systems. To this end, he developed a simple, and in some sense general, model of multi-agent systems, and showed how the histories traced out in the execution of such a system could be used as the semantic foundation for a family of both linear and branching time temporal belief logics. He then gave examples of how these logics could be used in the specification and verification of moderately realistic protocols for cooperative action.

2.4 Further Reading

For a detailed discussion of intentionality and the intentional stance, see [34, 35]. A number of papers on AI treatments of agency may be found in [5]. For an introduction to modal logic, see [26]; a slightly older, though more wide ranging introduction, may be found in [72]. As for the use of modal logics to model belief, see [65], which includes complexity results and proof procedures. Related work on modelling knowledge has been done by the distributed systems community, who give the worlds in possible worlds semantics a precise interpretation; for an introduction and further references, see [64, 41]. Overviews of formalisms for modelling belief and knowledge may be found in [63, 80, 108, 143]. A variant on the possible worlds framework, called the *recursive modelling method*, is described in [57]; a deep theory of belief may be found in [89]. *Situation semantics*, developed in the early 1980s and recently the subject of renewed interest, represent a fundamentally new approach to modelling the world and cognitive systems [10, 37]. However, situation semantics are not (yet) in the mainstream of (D)AI, and it is not obvious what impact the paradigm will ultimately have.

Logics which integrate time with mental states are discussed in [83, 66, 146]; the last of these presents a tableau-based proof method for a temporal belief logic. Two other important references for temporal aspects are [119, 120]. Thomas has developed some logics for representing agent theories as part of her framework for agent programming languages; see [131, 130] and section 4. For an introduction to the temporal logics and related topics, see [58, 38]. A non-formal discussion of intention may be found in [14], or more briefly [15]. Further work on modelling intention may be found in [60, 114, 59, 82]. Related work, focussing less on single-agent attitudes, and more on social aspects, is [74, 144, 147].

3 Agent Architectures

Until now, this article has been concerned with agent theory — the construction of formalisms for reasoning about agents, and the properties of agents expressed in such formalisms. Our aim in this section is to shift the emphasis from theory to practice. We consider the issues surrounding the construction of computer systems that satisfy the properties specified by agent theorists. We begin by looking at the symbolic AI paradigm, and the assumptions that underpin it.

3.1 Classical Approaches: Deliberative Architectures

The foundation upon which the symbolic AI paradigm rests is the *physical-symbol system hypothesis*, formulated by Newell and Simon [99][7]. A physical symbol system is defined to be a physically realizable set of physical entities (symbols) that can be combined to form structures, and which is capable of running processes which operate on those symbols according to symbolically coded sets of instructions. The physical-symbol system hypothesis then says that such a system is capable of general intelligent action.

It is a short step from the notion of a physical symbol system to McCarthy's dream of a *sentential processing automaton*, or *deliberate agent* (the term 'deliberate agent' was introduced by Genesereth, [50, pp325–327], but is here used in a slightly more general sense). A deliberate agent is one which contains an explicitly represented, symbolic model of the world, and in which decisions (for example about what actions to perform) are made via logical (or at least pseudo-logical) reasoning:

> 'Supporters of classical AI have, in general, accepted the physical symbol system hypothesis … [C]omplacent acceptance of this hypothesis, or some variant of it, led researchers to believe that the appropriate way to design an agent capable of finding its way round and acting in the physical world would be to equip it with some formal, logic-based representation of that world and get it to *do a bit of theorem proving*'. [118, §3.2]

If one aims to build such an agent, then there are at least two important problems to be solved:

1. The transduction problem: that of translating the real world into an accurate, adequate symbolic description, in time for that description to be useful.

2. The representation/reasoning problem: that of how to symbolically represent information about complex real-world entities and processes, and how to get agents to reason with this information in time for the results to be useful.

The former problem has led to work on vision, speech understanding, learning, etc. The latter has led to work on knowledge representation, automated reasoning, automated planning, etc. Despite the immense volume of work that the problems have generated, most researchers would accept that neither problem is anywhere near solved. Even seemingly trivial problems, such as commonsense reasoning, have turned out to be extremely difficult. It is because of these problems that some researchers have looked to alternative techniques; such alternatives are discussed in section 3.2. First, however, we consider efforts made within the symbolic AI community to construct agents.

Planning agents

Since the early 1970s, the AI planning community has been closely concerned with the design of artificial agents; in fact, it seems reasonable to claim that most innovations in agent design have come from this community. Planning is essentially automatic programming: the design of a detailed course of action which, when executed, will result in the achievement of some desired goal. Within the symbolic AI

[7] See [118] for a detailed discussion of the way that this hypothesis has affected thinking in symbolic AI.

community, it has long been assumed that some form of AI planning system will be a central component of any artificial agent. Perhaps the best know early planning system was STRIPS [44]. This system takes a symbolic descripion of both the world and a desired goal state, and a set of action descriptions, which characterise the pre- and post-conditions associated with various actions. It then attempts to find a sequence of actions that will achieve the goal, by using a simple means-ends analysis, which essentially involves matching the post-conditions of actions against the desired goal. The STRIPS planning algorithm was very simple, and proved to be ineffective on problems of even moderate complexity. Much effort was subsequently devoted to developing more effective automatic planning techniques. Two major innovations were *hierarchical* and *non-linear* planning [113, 112]. However, in the mid 1980s, Chapman established some theoretical results which indicate that even such refined techniques will ultimately turn out to be unusable in any time-constrained system [24]. These results have had a profound influence on subsequent AI planning research; perhaps more than any other results, they have caused some researchers to question the whole symbolic AI paradigm, and have thus led to the work on alternative approaches that we discuss in section 3.2.

In spite of these difficulties, various attempts have been made to construct agents whose primary component is a planner. For example: the Integrated Planning, Execution and Monitoring (IPEM) system is based on a sophisticated non-linear planner [7]; Wood's AUTODRIVE system has planning agents operating in a highly dynamic environment (a traffic simulation) [142].

Bratman, Israel & Pollack — IRMA

In section 2, we saw that some researchers have considered frameworks for agent theory based on beliefs, desires, and intentions [105]. Some researchers have also developed agent architectures based on these attitudes. One example is the *Intelligent Resource-bounded Machine Architecture* (IRMA) [16]. This architecture has four key symbolic data structures: a plan library, and explicit representations of beliefs, desires, and intentions. Additionally, the architecture has: a reasoner, for reasoning about the world; a means-ends analyser, for determining which plans might be used to achieve the agent's intentions; an *opportunity analyser*, which monitors the environment in order to determine further options for the agent; a *filtering process*; and a *deliberation process*. The filtering process is responsible for determining the subset of the agent's potential courses of action that have the property of being consistent with the agent's current intentions. A final choice between options is made by the deliberation process. The IRMA architecture has been evaluated in an experimental scenario known as the *Tileworld* [103].

Vere & Bickmore — Homer

An interesting experiment in the design of intelligent agents was conducted by Vere and Bickmore [134]. They argued that the enabling technologies for intelligent agents are sufficiently developed to be able to construct a prototype autonomous agent, with linguistic ability, planning and acting capabilities, and so on. They developed such an agent, and christened it Homer. This agent is a simulated robot submarine, which exists in a two-dimensional 'Seaworld', about which it has only partial knowledge. Homer takes instructions from a user in a limited subset of English with about an 800 word vocabulary; instructions can contain moderately sophisticated temporal references. Homer can plan how to achieve its instructions, (which typically relate to collecting and moving items around the Seaworld), and can then execute its plans, modifying them as required during execution. The agent has a limited *episodic memory*, and using this, is able to answer questions about its past experiences.

3.2 Alternative Approaches: Reactive Architectures

As we observed above, there are many unsolved (some would say intractable) problems associated with symbolic AI. These problems have lead some researchers to question the viability of the whole paradigm, and to the development of what are generally know as *reactive* architectures. For our purposes, we shall define a reactive architecture to be one which does not include any kind of central symbolic world model, and does not use complex symbolic reasoning.

Brooks — behaviour languages

Probably the most vocal critic of the symbolic AI notion of agency has been Rodney Brooks, a researcher at MIT who apparently became frustrated by AI approaches to building control mechanisms for autonomous mobile robots. In a 1985 paper, he outlined an alternative architecture for building agents, the so called *subsumption architecture* [17]. The analysis of alternative approaches begins with Brooks' work.

In recent papers, [20, 19, 18], Brooks has propounded three key theses:

1. Intelligent behaviour can be generated *without* explicit representations of the kind that symbolic AI proposes.

2. Intelligent behaviour can be generated *without* explicit abstract reasoning of the kind that symbolic AI proposes.

3. Intelligence is an *emergent* property of certain complex systems.

Brooks identifies two key ideas that have informed his research:

1. Situatedness and embodiment: 'Real' intelligence is situated in the world, not in disembodied systems such as theorem provers or expert systems.

2. Intelligence and emergence: 'Intelligent' behaviour arises as a result of an agent's interaction with its environment. Also, intelligence is 'in the eye of the beholder'; it is not an innate, isolated property.

If Brooks was just a Dreyfus-style critic of AI, his ideas might not have gained much currency. However, to demonstrate the validity of his claims, he has built a number of robots, based on the *subsumption architecture*. A subsumption architecture is a hierarchy of task-accomplishing *behaviours*. Each behaviour 'competes' with the others to exercise control over the robot. Lower layers represent more primitive kinds of behaviour, (such as avoiding obstacles), and have precedence over layers further up the hierarchy. It should be stressed that the resulting systems are, in terms of the amount of computation they need to do, *extremely* simple, with no explicit reasoning, or even pattern matching, of the kind found in symbolic AI systems. But despite this simplicity, Brooks has demonstrated the robots doing tasks that would be impressive if they were accomplished by symbolic AI systems. Similar work has been reported by Steels, who described simulations of 'Mars explorer' systems, containing a large number of subsumption-architecture agents, that can achieve near-optimal performance in certain tasks [129].

Agre & Chapman — Pengi

At about the same time as Brooks was describing his first results with the subsumption architecture, Chapman was completing his Master's thesis, in which he reported the theoretical difficulties with planning described above, and was coming to similar conclusions about the inadequacies of the symbolic AI model himself. Together with his co-worker Agre, he began to explore alternatives to the AI planning paradigm [25].

Agre observed that most everyday activity is 'routine' in the sense that it requires little — if any — new abstract reasoning. Most tasks, once learned, can be accomplished in a routine way, with little variation. Agre proposed that an efficient agent architecture could be based on the idea of 'running arguments'. Crudely, the idea is that as most decisions are 'routine', they can be encoded into a low-level structure (such as a digital circuit), which only needs periodic updating, perhaps to handle new kinds of problems. His approach was illustrated with the celebrated Pengi system [3]. Pengi is a simulated video game, with the central character controlled using a scheme such as that outlined above.

Rosenschein & Kaelbling — situated automata

Another sophisticated approach is that of Rosenschein and Kaelbling [110, 111, 76]. In their *situated automata* paradigm, an agent is specified in terms of a logic of knowledge. This specification is then compiled down to a low-level digital machine, which satisfies the intentional specification. The technique depends upon the possibility of giving the worlds in possible worlds semantics a concrete interpretation in terms of the states of an automaton:

> '[An agent] … x is said to carry the information that p in world state s, written $s \models K(x, p)$, if for all world states in which x has the same value as it does in s, the proposition p is true.' [76, p36]

The authors have developed several software tools to assist in the construction of agents: the Ruler program is used to specify the perception component of an agent; the Gapps program is used to specify the action component. Both of these languages are implemented over a third, LISP-like language, called Rex, which is used to specify simple digital machines. The situated automata paradigm has attracted much interest. However, at the time of writing, the theoretical limitations of the approach are not well understood.

Connah & Wavish — ABLE

A group of researchers at Philips research labs in the UK have developed an *Agent Behaviour Language*, (ABLE), in which agents are programmed in terms of simple, rule-like *licences* [31, 135]. Licences may include some representation of time (though the language is not based on any kind of temporal logic): they loosely resemble behaviours in the subsumption architecture (see above). ABLE can be compiled down to a simple digital machine, realised in the 'C' programming language. The idea is similar to situated automata, though there appears to be no equivalent theoretical foundation. The result of the compilation process is a very fast implementation, which has been reportedly used to control an Compact Disk-Interactive (CD-I) application.

3.3 Hybrid Architectures

Many researchers have suggested that neither a completely deliberative nor completely reactive approach is suitable for building agents. They have argued the case for *hybrid* systems, which attempt to marry classical and alternative approaches. In this section, we review these approaches.

Georgeff & Lansky — PRS

One of the best known agent architectures is the *Procedural Reasoning System* (PRS), developed by Georgeff and Lansky [54]. Like IRMA, (see above), the PRS is a belief-desire-intention architecture, which includes a plan library, as well as explicit symbolic representations of beliefs, desires, and intentions. Beliefs are facts, either about the external world or the system's internal state, and are expressed in classical first-order logic. Desires are represented as *system behaviours* (rather than as static representations of goal states). A PRS plan library contains a set of partially-elaborated plans, called *knowledge areas* (KAs), each of which is associated with an *invocation condition*. This condition determines when the KA is to be *activated*. KAs may be activated in a goal-driven or data driven fashion; KAs may also be *reactive*, allowing the PRS to respond rapidly to changes in its environment. The set of currently active KAs in a system represent its *intentions*. These various data structures are manipulated by a *system interpreter*, which is responsible for updating beliefs, invoking KAs, and executing actions. The PRS has been evaluated in a simulation of maintenance procedures for the space shuttle, as well as other domains [52].

Ferguson — TouringMachines

For his 1992 Doctoral thesis, Ferguson developed the TouringMachines hybrid agent architecture [43, 42][8]. The architecture consists of *perception* and *action* subsystems, which interface directly with the agent's environment, and three *control layers*, embedded in a *control framework*, which mediates between the layers. Each layer is an independent, activity-producing, concurrently executing process.

The *reactive layer* generates potential courses of action in response to events that happen too quickly for the other layers to deal with. It is implemented as a set of situation-action rules, in the style of Brooks' subsumption architecture (see above).

The *planning layer* constructs plans and selects actions to execute in order to achieve the agent's goals. This layer consists of two components: a planner, and a *focus of attention mechanism*. The planner integrates plan generation and execution, and uses a library of partially elaborated plans, together with a topological world map, in order to construct plans that will accomplish the agent's main goal. The purpose of the focus of attention mechanism is to limit the amount of information that the planner must deal with, and so improve its efficiency. It does this by filtering out irrelevant information from the environment.

The *modelling layer* contains symbolic representations of the cognitive state of other entities in the agent's environment. These models are manipulated in order to identify and resolve *goal conflicts* — situations where an agent can no longer achieve its goals, as a result of unexpected interference.

The three layers are able to communicate with each other (via message passing), and are embedded in a control framework. The purpose of this framework is to mediate between the layers, and in particular, to deal with conflicting action proposals from the different layers. The control framework does this by using *control rules*.

3.4 Further Reading

Most introductory textbooks on AI discuss the physical symbol system hypothesis; a good recent example of such a text is [56]. There are many objections to the symbolic AI paradigm, in addition to those we have outlined above. Again, introductory textbooks provide the stock criticisms and replies.

There is a wealth of material on planning and planning agents. See [51] for an overview of the state of the art in planning (as it was in 1987), [5] for a thorough collection of papers on planning, (many of the papers cited above are included), and [140] for a detailed description of SIPE, a sophisticated planning system used in a real-world application (the control of a brewery!) Another important collection of planning papers is [53]. The books by Dean & Wellman and Allen *et al.* contain much useful related material [33, 6]. There is now a regular international conference on planning; the proceedings of the first were published as [68].

The collection of papers edited by Maes [90] contains many interesting papers on alternatives to the symbolic AI paradigm. Kaelbling [75] presents a clear discussion of the issues associated with developing resource-bounded rational agents, and proposes an agent architecture somewhat similar to that developed by Brooks. A proposal by Nilsson for *teleo reactive programs* — goal directed programs that nevertheless respond to their environment — is described in [100]. The proposal draws heavily on the situated automata paradigm; other work based on this paradigm is described in [121, 78]. Schoppers has

[8]It is worth noting that Ferguson's thesis gives an excellent overview of the problems and issues associated with building rational, resource-bounded agents. Moreover, the description given of the TouringMachines architecture is itself extremely clear. We recommend it as a point of departure for further reading.

proposed compiling plans in advance, using traditional planning techniques, in order to develop *universal plans*, which are essentially decision trees that can be used to efficiently determine an appropriate action in any situation [115]. Other proposals for 'reactive planners' are *reactive action packages* [45] and *competence modules* [91].

A hybrid architecture specifically developed for multi-agent applications is described in [62].

4 Agent Languages

By an *agent language*, we mean a system that allows one to program hardware or software computer systems in terms of some of the concepts developed by agent theorists. At the very least, we expect such a language include some structure corresponding to an agent. However, we would also expect to see some other attributes of agency (beliefs, goals, or other mentalistic notions) used to program agents. As the reader can see, the distinction between an agent language and architecture is somewhat artificial; many of the architectures mentioned above could be counted as languages by this definition.

Much of the current interest in agent languages is a result of Shoham's proposal for *agent-oriented programming*. We begin our consideration of languages with a survey of Shoham's work.

Shoham — agent-oriented programming

Yoav Shoham has proposed a 'new programming paradigm, based on a societal view of computation' [121, p4],[122]. The key idea which informs this *agent-oriented programming* (AOP) paradigm is that of directly programming agents in terms of the mentalistic, intentional notions that agent theorists have developed to represent the properties of agents. The motivation behind such a proposal is that, as we observed in section 2, humans use the intentional stance as an *abstraction* mechanism for representing the properties of complex systems. In the same way that we use the intentional stance to describe humans, it might be useful to use the intentional stance to program machines.

Shoham proposes that a fully developed AOP system will have three components:

- a logical system for defining the mental state of agents;
- an interpreted programming language for programming agents;
- an 'agentification' process, for compiling agent programs into low-level executable systems.

At the time of writing, Shoham has only published results on the first two components. (In [121, p12] he wrote that 'the third is still somewhat mysterious to me', though later in the paper he indicated that he was thinking of something along the lines of Rosenschein & Kaelbling's situated automata paradigm [111].) Shoham's first attempt at an AOP language was the AGENT0 system. The logical component of this system is a quantified multi-modal logic, allowing direct reference to time. No semantics are given, but the logic appears to be based on [131]. The logic contains three modalities: belief, commitment and ability. The following is an acceptable formula of the logic, illustrating it's key properties:

$$CAN_a^5\ open(door)^8 \Rightarrow B_b^5\ CAN_a^5\ open(door)^8.$$

This formula is read: 'if at time 5 agent a can ensure that the door is open at time 8, then at time 5 agent b believes that at time 5 agent a can ensure that the door is open at time 8'.

Corresponding to the logic is the AGENT0 programming language. In this language, an agent is specified in terms of a set of capabilities (things the agent can do), a set of initial beliefs and commitments, and a set of *commitment rules*. The key component, which determines how the agent acts, is the commitment rule set. Each commitment rule contains a *message condition*, a *mental condition*, and an action. In order to determine whether such a rule fires, the message condition is matched against the messages the agent has received; the mental condition is matched against the beliefs of the agent. If the rule fires, then the agent becomes committed to the action. Actions may be *private*, corresponding to an internally executed subroutine, or *communicative*, i.e., sending messages. Messages are constrained to be one of three types: 'requests' or 'unrequests' to perform or refrain from actions, and 'inform' messages, which pass on information — Shoham indicates that he took his inspiration for these message types from speech act theory [116, 30]. Request and unrequest messages typically result in the agent's commitments being modified; inform messages result in a change to the agent's beliefs.

AGENT0 was only ever intended as a prototype, to illustrate the principles of AOP. A more refined implementation was developed by Thomas, for her 1993 doctoral thesis [130]. Her Planning Communicating Agents (PLACA) language was intended to address one severe drawback to AGENT0: the inability of agents to plan, and communicate requests for action via high-level goals. Agents in PLACA are programmed in much the same way as in AGENT0, in terms of *mental change* rules. The logical component of PLACA is similar to AGENT0's, but includes operators for planning to do actions and achieve goals. The semantics of the logic and its properties are examined in detail. However, PLACA is not at the 'production' stage; it is an experimental language.

Fisher — Concurrent METATEM

One drawback with both AGENT0 and PLACA is that the relationship between the logic and interpreted programming language is only loosely defined: in neither case can the programming language be said to truly *execute* the associated logic. The Concurrent METATEM language developed by Fisher can make a stronger claim in this respect [46]. A Concurrent METATEM system contains a number of concurrently executing agents, each of which is able to communicate with its peers via asynchronous broadcast message passing. Each agent is programmed by giving it a temporal logic specification of the behaviour that it is intended the agent should exhibit. An agent's specification is executed directly to generate its behaviour. Execution of the agent program corresponds to iteratively building a logical model for the temporal agent specification. It is possible to prove that the procedure used to execute an agent specification is correct, in that if it is possible to satisfy the specification, then the agent will do so [9].

The logical semantics of Concurrent METATEM are closely related to the semantics of temporal logic itself. This means that, amongst other things, the specification and verification of Concurrent METATEM system is a viable proposition. At the time of writing, only prototype implementations of the language are available; full implementations are expected soon.

4.1 Further Reading

There are many other languages which, while they may not be agent languages in the sense we have described, are nevertheless of interest. *Concurrent object* languages are of considerable interest in software engineering. The notion of a self-contained concurrently executing object, with some internal state that is not directly accessible to the outside world, responding to messages from other such objects, is very close to the concept of an agent as we have defined it. The only significant difference is that our agents are defined in terms of beliefs, goals, and so on. The earliest concurrent object framework was Hewitt's Actor model [70, 1]; another well-known example is the ABCL system [148]. A recent collection of papers on concurrent object systems is [2].

Other languages of interest include Oz [69] and IC PROLOG II [27]. The latter, as its name suggests, is an extension of PROLOG, which includes multiple-threads, high-level communication primitives, and some object-oriented features.

5 Concluding Remarks

In this paper, we hope to have at least mentioned the major research issues and developments associated with the synthesis of artificial agents from the point of view of AI. In this final section, we point the reader at some current applications of agent technology in AI and computer science generally.

Distributed AI

As we observed in section 1, there has been a marked flowering of interest in agent technology since the mid-1980s. This interest is in part due to the renewed interest in Distributed AI. Although DAI encompasses most of the issues we have discussed in this paper, it should be stressed that the classical emphasis in DAI has been on *macro* phenomona (the *social* level), rather than the *micro* phenomena (the *agent* level) that we have been concerned with in this paper. DAI thus looks at such issues as how a group of agents can be made to cooperate in order to efficiently solve problems, and how the activities of such a group can be efficiently coordinated. DAI researchers have applied agent technology in a variety of areas. Example applications include power systems management [141], air traffic control [128], and intelligent document retrieval [98]. The classic reference to DAI is [13].

Agents in CSCW

The possible applications of agent technology in computer supported cooperative work (CSCW) are currently the subject of much interest. CSCW is informally defined by Baecker to be 'computer assisted coordinated activity such as problem solving and communication carried out by a group of collaborating individuals' [8, p1]. The primary emphasis of CSCW is on the development of (hardware and) software tools to support collaborative human work — the term *groupware* has been coined to describe such tools. Various authors have proposed the use of agent technology in groupware. For example, in his *participant systems* proposal, Chang suggests systems in which humans collaborate with not only other humans, but also with artificial agents [23]. McGregor has imagined *prescient agents* — intelligent administrative assistants, that predict our actions, and carry out routine or repetetive administrative procedures on our behalf [94]. For example, imagine a group of computer agents that scanned email and prioritised it, junking irrelevant or duplicated mail items[9]; or imagine a group of agents that scanned USENET news for you, bringing to your attention conference announcements of interest, perhaps even using FTP or World Wide Web to cooperate with other agents and obtain papers that look relevant; imagine how useful (or perhaps annoying) it would be to have an agent remind you of a conference deadline in time to finish off a half-written paper. While these applications are (mostly) some time off, their basic principles are, at the time of writing, the subject of serious academic and industrial research. We refer the interested reader to the collection of papers edited by Baecker for more details [8].

Agents in Virtual Environments

The potential for marrying agent technology with virtual environments is being investigated in the Oz project[10] [11]. The aim of the project is to develop '... artistically interesting, highly interactive, simulated worlds... to give users the experience of living in (not merely watching) dramatically rich worlds that include moderately competent, emotional agents' [12, p1]. A pre-requisite is the development of *broad* agents — systems that include 'a broad set of capabilities, including goal-directed reactive behaviour, emotional state and behaviour, and some natural language abilities' [12, p1].

[9] Such a system has been prototyped [92].

[10] Not to be confused with the OZ programming language [69].

Towards Agent-Based Open Systems

While the reviews presented above are by no means exhaustive, they at least indicate that computer scientists and AI researchers with a range of interests, applications, and backgrounds are taking agent technology seriously. As computer systems become ever more open and interconnected, we may expect that this technology will become increasingly common.

Acknowledgements

Much of this paper was adapted from the first author's 1992 PhD thesis [143], and as such this work was supported by the UK Science and Engineering Research Council (SERC).

References

[1] G. Agha. *ACTORS: A Model of Concurrent Computation in Distributed Systems*. The MIT Press, 1986.

[2] G. Agha, P. Wegner, and A. Yonezawa, editors. *Research Directions in Concurrent Object-Oriented Programming*. The MIT Press, 1993.

[3] P. Agre and D. Chapman. PENGI: An implementation of a theory of activity. In *Proceedings of the Sixth National Conference on Artificial Intelligence (AAAI-87)*, pages 268–272, Seattle, WA, 1987.

[4] J. F. Allen. Towards a general theory of action and time. *Artificial Intelligence*, 23(2):123–154, 1984.

[5] J. F. Allen, J. Hendler, and A. Tate, editors. *Readings in Planning*. Morgan Kaufmann Publishers, Inc., 1990.

[6] J. F. Allen, H. Kautz, R. Pelavin, and J. Tenenberg. *Reasoning About Plans*. Morgan Kaufmann Publishers, Inc., 1991.

[7] J. Ambros-Ingerson and S. Steel. Integrating planning, execution and monitoring. In *Proceedings of the Seventh National Conference on Artificial Intelligence (AAAI-88)*, St. Paul, MN, 1988.

[8] R. M. Baecker, editor. *Readings in Groupware and Computer-Supported Cooperative Work*. Morgan Kaufmann Publishers, Inc., 1993.

[9] H. Barringer, M. Fisher, D. Gabbay, G. Gough, and R. Owens. METATEM: A framework for programming in temporal logic. In *REX Workshop on Stepwise Refinement of Distributed Systems: Models, Formalisms, Correctness (LNCS Volume 430)*, pages 94–129. Springer-Verlag, June 1989.

[10] J. Barwise and J. Perry. *Situations and Attitudes*. The MIT Press, 1983.

[11] J. Bates. Virtual reality, art, and entertainment. *PRESENCE: Teleoperators and Virtual Environments*, 1(1):133–138, 1992.

[12] J. Bates, A. Bryan Loyall, and W. Scott Reilly. Integrating reactivity, goals, and emotion in a broad agent. Technical Report CMU–CS–92–142, School of Computer Science, Carnegie-Mellon University, Pittsburgh, PA, May 1992.

[13] A. H. Bond and L. Gasser, editors. *Readings in Distributed Artificial Intelligence*. Morgan Kaufmann Publishers, Inc., 1988.

[14] M. E. Bratman. *Intentions, Plans, and Practical Reason*. Harvard University Press, 1987.

[15] M. E. Bratman. What is intention? In P. R. Cohen, J. L. Morgan, and M. E. Pollack, editors, *Intentions in Communication*, pages 15–32. The MIT Press, 1990.

[16] M. E. Bratman, D. J. Israel, and M. E. Pollack. Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4:349–355, 1988.

[17] R. Brooks. A robust layered control system for a mobile robot. Technical Report AI Memo 864, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1985.

[18] R. Brooks. Intelligence without reason. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91)*, Sydney, Australia, 1991.

[19] R. Brooks. Intelligence without representation. *Artificial Intelligence*, 47, 1991.

[20] R. A. Brooks. Elephants don't play chess. In P. Maes, editor, *Designing Autonomous Agents*, pages 3–15. The MIT Press, 1990.

[21] C. Castelfranchi. Social power. In Y. Demazeau and J.-P. Müller, editors, *Decentralized AI — Proceedings of the First European Workshop on Modelling Autonomous Agents in Multi-Agent Worlds (MAAMAW-89)*, pages 49–62. Elsevier/North Holland, 1990.

[22] C. Castelfranchi, M. Miceli, and A. Cesta. Dependence relations among autonomous agents. In E. Werner and Y. Demazeau, editors, *Decentralized AI 3 — Proceedings of the Third European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-91)*, pages 215–231. Elsevier/North Holland, 1992.

[23] E. Chang. Participant systems. In M. Huhns, editor, *Distributed Artificial Intelligence*, pages 311–340. Pitman/Morgan Kaufmann, 1987.

[24] D. Chapman. Planning for conjunctive goals. *Artificial Intelligence*, 32, 1987.

[25] D. Chapman and P. Agre. Abstract reasoning as emergent from concrete activity. In M. P. Georgeff and A. L. Lansky, editors, *Proceedings of the 1986 Workshop on Reasoning About Actions and Plans*. Morgan Kaufmann Publishers, Inc., 1986.

[26] B. Chellas. *Modal Logic: An Introduction*. Cambridge University Press, 1980.

[27] D. Chu. I.C. PROLOG II: A language for implementing multi-agent systems. In S. M. Deen, editor, *Proceedings of the 1992 Workshop on Cooperating Knowledge Based Systems (CKBS-92)*, pages 61–74. DAKE Centre, University of Keele, UK, 1993.

[28] P. R. Cohen and H. J. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42:213–261, 1990.

[29] P. R. Cohen and H. J. Levesque. Rational interaction as the basis for communication. In P. R. Cohen, J. Morgan, and M. E. Pollack, editors, *Intentions in Communication*, pages 221–256. The MIT Press, 1990.

[30] P. R. Cohen and C. R. Perrault. Elements of a plan based theory of speech acts. *Cognitive Science*, 3, 1979.

[31] D. Connah and P. Wavish. An experiment in cooperation. In Y. Demazeau and J.-P. Müller, editors, *Decentralized AI — Proceedings of the First European Workshop on Modelling Autonomous Agents in Multi-Agent Worlds (MAAMAW-89)*, pages 197–214. Elsevier/North Holland, 1990.

[32] N. J. Davies. *Truth, Modality, and Action*. PhD thesis, Department of Computer Science, University of Essex, Colchester, UK, March 1993.

[33] T. L. Dean and M. P. Wellman. *Planning and Control*. Morgan Kaufmann Publishers, Inc., 1991.

[34] D. C. Dennett. *Brainstorms*. The MIT Press, 1978.

[35] D. C. Dennett. *The Intentional Stance*. The MIT Press, 1987.

[36] J. des Rivieres and H. J. Levesque. The consistency of syntactical treatments of knowledge. In J. Y. Halpern, editor, *Proceedings of the 1986 Conference on Theoretical Aspects of Reasoning About Knowledge*, pages 115–130. Morgan Kaufmann Publishers, Inc., 1986.

[37] K. Devlin. *Logic and Information*. Cambridge University Press, 1991.

[38] E. A. Emerson. Temporal and modal logic. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, pages 996–1072. Elsevier, 1990.

[39] E. A. Emerson and J. Y. Halpern. 'Sometimes' and 'not never' revisited: on branching time versus linear time temporal logic. *Journal of the ACM*, 33(1):151–178, 1986.

[40] R. Fagin and J. Y. Halpern. Belief, awareness, and limited reasoning. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence (IJCAI-85)*, Los Angeles, CA, 1985.

[41] R. Fagin, J. Y. Halpern, and M. Y. Vardi. What can machines know? on the properties of knowledge in distributed systems. *Journal of the ACM*, 39(2):328–376, 1992.

[42] I. A. Ferguson. *TouringMachines: An Architecture for Dynamic, Rational, Mobile Agents*. PhD thesis, Clare Hall, University of Cambridge, UK, November 1992. (Also available as Technical Report No. 273, University of Cambridge Computer Laboratory).

[43] I. A. Ferguson. Towards an architecture for adaptive, rational, mobile agents. In E. Werner and Y. Demazeau, editors, *Decentralized AI 3 — Proceedings of the Third European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-91)*, pages 249–262. Elsevier/North Holland, 1992.

[44] R. E. Fikes and N. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 5(2), 1971.

[45] J. A. Firby. An investigation into reactive planning in complex domains. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence (IJCAI-87)*, pages 202–206, Milan, Italy, 1987.

[46] M. Fisher. A survey of Concurrent METATEM — the language and its applications. In D. Gabbay and H.-J. Ohlbach, editors, *Proceedings of the First International Conference on Temporal Logic (ICTL-94)*. Springer-Verlag, July 1994. (To appear).

[47] J. R. Galliers. A strategic framework for multi-agent cooperative dialogue. In *Proceedings of the Eighth European Conference on Artificial Intelligence (ECAI-88)*. Pitman, 1988.

[48] J. R. Galliers. *A Theoretical Framework for Computer Models of Cooperative Dialogue, Acknowledging Multi-Agent Conflict*. PhD thesis, Open University, UK, 1988.

[49] C. Geissler and K. Konolige. A resolution method for quantified modal logics of knowledge and belief. In J. Y. Halpern, editor, *Proceedings of the 1986 Conference on Theoretical Aspects of Reasoning About Knowledge*, pages 309–324. Morgan Kaufmann Publishers, Inc., 1986.

[50] M. R. Genesereth and N. Nilsson. *Logical Foundations of Artificial Intelligence*. Morgan Kaufmann Publishers, Inc., 1987.

[51] M. P. Georgeff. Planning. *Annual Review of Computer Science*, 2, 1987.

[52] M. P. Georgeff and F. F. Ingrand. Decision-making in an embedded reasoning system. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI-89)*, Detroit, MI, 1989.

[53] M. P. Georgeff and A. L. Lansky, editors. *Proceedings of the 1986 Workshop on Reasoning About Actions and Plans*. Morgan Kaufmann Publishers, Inc., 1986.

[54] M. P. Georgeff and A. L. Lansky. Reactive reasoning and planning. In *Proceedings of the Sixth National Conference on Artificial Intelligence (AAAI-87)*, pages 677–682, Seattle, WA, 1987.

[55] E. Germain. Software's special agents. *New Scientist*, 142(1920):19–20, April 1994.

[56] M. Ginsberg. *Essentials of Artificial Intelligence*. Morgan Kaufmann Publishers, Inc., 1993.

[57] P. Gmytrasiewicz and E. H. Durfee. Elements of a utilitarian theory of knowledge and action. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI-93)*, pages 396–402, Chambéry, France, 1993.

[58] R. Goldblatt. *Logics of Time and Computation*. Centre for the Study of Language and Information — Lecture Notes Series, 1987. (Distributed by Chicago University Press).

[59] R. P. Goldman and R. R. Lang. Intentions in time. Technical Report TUTR 93–101, Tulane University, January 1991.

[60] B. J. Grosz and C. L. Sidner. Plans for discourse. In P. R. Cohen, J. Morgan, and M. E. Pollack, editors, *Intentions in Communication*, pages 417–444. The MIT Press, 1990.

[61] A. Haas. A syntactic theory of belief and knowledge. *Artificial Intelligence*, 28(3), 1986.

[62] A. Haddadi. A hybrid architecture for multi-agent systems. In S. M. Deen, editor, *Proceedings of the 1993 Workshop on Cooperating Knowledge Based Systems (CKBS-93)*, pages 13–26, DAKE Centre, University of Keele, UK, 1994.

[63] J. Y. Halpern. Reasoning about knowledge: An overview. In J. Y. Halpern, editor, *Proceedings of the 1986 Conference on Theoretical Aspects of Reasoning About Knowledge*, pages 1–18. Morgan Kaufmann Publishers, Inc., 1986.

[64] J. Y. Halpern. Using reasoning about knowledge to analyze distributed systems. *Annual Review of Computer Science*, 2, 1987.

[65] J. Y. Halpern and Y. Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence*, 54:319–379, 1992.

[66] J. Y. Halpern and M. Y. Vardi. The complexity of reasoning about knowledge and time. I. lower bounds. *Journal of Computer and System Sciences*, 38:195–237, 1989.

[67] D. Harel. Dynamic logic. In D. Gabbay and F. Guenther, editors, *Handbook of Philosophical Logic Volume II — Extensions of Classical Logic*, pages 497–604. D. Reidel Publishing Company, 1984. (Synthese library Volume 164).

[68] J. Hendler, editor. *Artificial Intelligence Planning: Proceedings of the First International Conference*. Morgan Kaufmann Publishers, Inc., 1992.

[69] M. Henz, G. Smolka, and J. Wuertz. Oz — a programming language for multi-agent systems. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI-93)*, pages 404–409, Chambéry, France, 1993.

[70] C. Hewitt. Viewing control structures as patterns of passing messages. *Artificial Intelligence*, 8(3):323–364, 1977.

[71] J. Hintikka. *Knowledge and Belief.* Cornell University Press, 1962.

[72] G. E. Hughes and M. J. Cresswell. *Introduction to Modal Logic.* Methuen and Co., Ltd., 1968.

[73] N. R. Jennings. On being responsible. In E. Werner and Y. Demazeau, editors, *Decentralized AI 3 — Proceedings of the Third European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-91)*, pages 93–102. Elsevier/North Holland, 1992.

[74] N. R. Jennings. Commitments and conventions: The foundation of coordination in multi-agent systems. *Knowledge Engineering Review*, 8(3):223–250, 1993.

[75] L. P. Kaelbling. An architecture for intelligent reactive systems. In M. P. Georgeff and A. L. Lansky, editors, *Proceedings of the 1986 Workshop on Reasoning About Actions and Plans*. Morgan Kaufmann Publishers, Inc., 1986.

[76] L. P. Kaelbling and S. J. Rosenschein. Action and planning in embedded agents. In P. Maes, editor, *Designing Autonomous Agents*, pages 35–48. The MIT Press, 1990.

[77] D. Kinny, M. Ljungberg, A. S. Rao, E. Sonenberg, G. Tidhar, and E. Werner. Planned team activity. In *Proceedings of the Fourth European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-92)*, 1992.

[78] G. Kiss and H. Reichgelt. Towards a semantics of desires. In E. Werner and Y. Demazeau, editors, *Decentralized AI 3 — Proceedings of the Third European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-91)*, pages 115–128. Elsevier/North Holland, 1992.

[79] K. Konolige. A first-order formalization of knowledge and action for a multi-agent planning system. In J. E. Hayes, D. Michie, and Y. Pao, editors, *Machine Intelligence 10*. Ellis Horwood, 1982.

[80] K. Konolige. *A Deduction Model of Belief.* Pitman/Morgan Kaufmann, 1986.

[81] K. Konolige. What awareness isn't: A sentential view of implicit and explicit belief (position paper). In J. Y. Halpern, editor, *Proceedings of the 1986 Conference on Theoretical Aspects of Reasoning About Knowledge*, pages 241–250. Morgan Kaufmann Publishers, Inc., 1986.

[82] K. Konolige and M. E. Pollack. A representationalist theory of intention. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI-93)*, pages 390–395, Chambéry, France, 1993.

[83] S. Kraus and D. Lehmann. Knowledge, belief and time. *Theoretical Computer Science*, 58:155–174, 1988.

[84] S. Kripke. Semantical analysis of modal logic. *Zeitschrift fur Mathematische Logik und Grudlagen der Mathematik*, 9, 1963.

[85] G. Lakemeyer. A computationally attractive first-order logic of belief. In *JELIA-90: Proceedings of the European Workshop on Logics in AI (LNAI Volume 478)*, pages 333–347. Springer-Verlag, 1991.

[86] Y. Lespérance. A formal account of self knowledge and action. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI-89)*, Detroit, MI, 1989.

[87] H. J. Levesque. A logic of implicit and explicit belief. In *Proceedings of the Fourth National Conference on Artificial Intelligence (AAAI-84)*, Austin, TX, 1984.

[88] H. J. Levesque, P. R. Cohen, and J. H. T. Nunes. On acting together. In *Proceedings of the Eighth National Conference on Artificial Intelligence (AAAI-90)*, pages 94–99, Boston, MA, 1990.

[89] D. Mack. Belief as pragmatic information: A new formal model. In S. M. Deen, editor, *Proceedings of the 1993 Workshop on Cooperating Knowledge Based Systems (CKBS-93)*, pages 117–134. DAKE Centre, University of Keele, UK, 1994.

[90] P. Maes, editor. *Designing Autonomous Agents*. The MIT Press, 1990.

[91] P. Maes. Situated agents can have goals. In P. Maes, editor, *Designing Autonomous Agents*, pages 49–70. The MIT Press, 1990.

[92] T. W. Malone, K. R. Grant, K.-Y. Lai, R. Rao, and D. A. Rosenblitt. The information lens: An intelligent system for information sharing and coordination. In M. H. Olson, editor, *Technological Support for Work Group Collaboration*, pages 65–88. Lawrence Erlbaum Associates, 1989.

[93] J. McCarthy. Ascribing mental qualities to machines. Technical report, Stanford AI Lab., 1978.

[94] S. L. McGregor. Prescient agents. In D. Coleman, editor, *Proceedings of Groupware-92*, pages 228–230, 1992.

[95] R. Montague. Syntactical treatments of modality, with corollaries on reflexion principles and finite axiomatizations. *Acta Philosophica Fennica*, 16, 1963.

[96] R. C. Moore. A formal theory of knowledge and action. In J. R. Hobbs and R. C. Moore, editors, *Formal Theories of the Commonsense World*. Ablex Publishing Corporation, 1985.

[97] L. Morgenstern. Knowledge preconditions for actions and plans. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence (IJCAI-87)*, Milan, Italy, 1987.

[98] U. Mukhopadhyay, L. Stephens, and M. Huhns. An intelligent system for document retrieval in distributed office environments. *Journal of the American Society for Information Science*, 37, 1986.

[99] A. Newell and H. A. Simon. Computer science as empirical enquiry. *Communications of the ACM*, 19:113–126, 1976.

[100] N. J. Nilsson. Towards agent programs with circuit semantics. Technical Report STAN–CS–92–1412, Department of Computer Science, Stanford University, January 1992.

[101] D. Perlis. Languages with self reference I: Foundations. *Artificial Intelligence*, 25, 1985.

[102] D. Perlis. Languages with self reference II: Knowledge, belief, and modality. *Artificial Intelligence*, 34, 1988.

[103] M. E. Pollack and M. Riguette. Introducing the tileworld: Experimentally evaluating agent architectures. In *Proceedings of the Eighth National Conference on Artificial Intelligence (AAAI-90)*, Boston, MA, 1990.

[104] A. Rao and M. P. Georgeff. Asymmetry thesis and side-effect problems in linear time and branching time intention logics. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91)*, pages 498–504, Sydney, Australia, 1991.

[105] 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. Morgan Kaufmann Publishers, Inc., April 1991.

[106] A. S. Rao and M. P. Georgeff. Social plans: Preliminary report. In E. Werner and Y. Demazeau, editors, *Decentralized AI 3 — Proceedings of the Third European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-91)*, pages 57–76. Elsevier/North Holland, 1992.

[107] A. S. Rao and M. P. Georgeff. A model-theoretic approach to the verification of situated reasoning systems. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI-93)*, pages 318–324, Chambéry, France, 1993.

[108] H. Reichgelt. A comparison of first-order and modal logics of time. In P. Jackson, H. Reichgelt, and F. van Harmelen, editors, *Logic Based Knowledge Representation*. The MIT Press, 1989.

[109] H. Reichgelt. Logics for reasoning about knowledge and belief. *Knowledge Engineering Review*, 4(2), 1989.

[110] S. Rosenschein. Formal theories of knowledge in AI and robotics. *New Generation Computing*, pages 345–357, 1985.

[111] S. Rosenschein and L. Kaelbling. The synthesis of digital machines with provable epistemic properties. In J. Y. Halpern, editor, *Proceedings of the 1986 Conference on Theoretical Aspects of Reasoning About Knowledge*, pages 83–98. Morgan Kaufmann Publishers, Inc., 1986.

[112] E. Sacerdoti. The non-linear nature of plans. In *Proceedings of the Fourth International Joint Conference on Artificial Intelligence (IJCAI-75)*, Stanford, CA, 1975.

[113] E. Sacerdoti. Planning in a hierarchy of abstraction spaces. *Artificial Intelligence*, 5(2), 1975.

[114] M. D. Sadek. A study in the logic of intention. In C. Rich, W. Swartout, and B. Nebel, editors, *Proceedings of Knowledge Representation and Reasoning (KR&R-92)*, pages 462–473, 1992.

[115] M. J. Schoppers. Universal plans for reactive robots in unpredictable environments. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence (IJCAI-87)*, pages 1039–1046, Milan, Italy, 1987.

[116] J. R. Searle. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, 1969.

[117] N. Seel. *Agent Theories and Architectures*. PhD thesis, Surrey University, Guildford, UK, 1989.

[118] N. Shardlow. Action and agency in cognitive science. Master's thesis, Department of Psychlogy, University of Manchester, Oxford Rd., Manchester M13 9PL, UK, 1990.

[119] Y. Shoham. *Reasoning About Change: Time and Causation from the Standpoint of Artificial Intelligence*. The MIT Press, 1988.

[120] Y. Shoham. Time for action: on the relation between time, knowledge and action. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI-89)*, Detroit, MI, 1989.

[121] Y. Shoham. Agent-oriented programming. Technical Report STAN–CS–1335–90, Department of Computer Science, Stanford University, 1990.

[122] Y. Shoham. Agent-oriented programming. *Artificial Intelligence*, 60(1):51–92, 1993.

[123] M. P. Singh. Towards a theory of situated know-how. In *Proceedings of the Ninth European Conference on Artificial Intelligence (ECAI-90)*. Pitman, 1990.

[124] M. P. Singh. Group ability and structure. In Y. Demazeau and J.-P. Müller, editors, *Decentralized AI 2 — Proceedings of the Second European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-90)*, pages 127–146. Elsevier/North Holland, 1991.

[125] M. P. Singh. Towards a formal theory of communication for multi-agent systems. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91)*, Sydney, Australia, 1991.

[126] M. P. Singh. A critical examination of the Cohen-Levesque theory of intention. In B. Neumann, editor, *Proceedings of the Tenth European Conference on Artificial Intelligence (ECAI-92)*, pages 364–368. John Wiley & Sons, August 1992.

[127] M. P. Singh and N. M. Asher. Towards a formal theory of intentions. In *Logics in AI — Proceedings of the European Workshop JELIA-90 (LNAI Volume 478)*, pages 472–486. Springer-Verlag, 1991.

[128] R. Steeb, S. Cammarata, F. R. Hayes-Roth, P. W. Thorndyke, and R. B. Wesson. Distributed intelligence for air traffic control. Technical Report R-2728-ARPA, Rand Corp., 1981.

[129] L. Steels. Cooperation between distributed agents through self organization. In Y. Demazeau and J.-P. Müller, editors, *Decentralized AI — Proceedings of the First European Workshop on Modelling Autonomous Agents in Multi-Agent Worlds (MAAMAW-89)*, pages 175–196. Elsevier/North Holland, 1990.

[130] S. R. Thomas. *PLACA, an Agent Oriented Programming Language*. PhD thesis, Computer Science Department, Stanford University, Stanford, CA 94305, August 1993. (Available as technical report STAN–CS–93–1487).

[131] S. R. Thomas, Y. Shoham, A. Schwartz, and S. Kraus. Preliminary thoughts on an agent description language. *International Journal of Intelligent Systems*, 6:497–508, 1991.

[132] R. Thomason. A note on syntactical treatments of modality. *Synthese*, 44, 1980.

[133] R. Turner. *Truth and Modality for Knowledge Representation*. Pitman, 1990.

[134] S. Vere and T. Bickmore. A basic agent. *Computational Intelligence*, 6:41–60, 1990.

[135] P. Wavish. Exploiting emergent behaviour in multi-agent systems. In E. Werner and Y. Demazeau, editors, *Decentralized AI 3 — Proceedings of the Third European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-91)*, pages 297–310. Elsevier/North Holland, 1992.

[136] E. Werner. Toward a theory of communication and cooperation for multiagent planning. In M. Y. Vardi, editor, *Proceedings of the Second Conference on Theoretical Aspects of Reasoning About Knowledge*, pages 129–144. Morgan Kaufmann Publishers, Inc., 1988.

[137] E. Werner. Cooperating agents: A unified theory of communication and social structure. In L. Gasser and M. Huhns, editors, *Distributed Artificial Intelligence Volume II*, pages 3–36. Pitman/Morgan Kaufmann, 1989.

[138] E. Werner. What can agents do together: A semantics of co-operative ability. In *Proceedings of the Ninth European Conference on Artificial Intelligence (ECAI-90)*. Pitman, 1990.

[139] E. Werner. A unified view of information, intention and ability. In Y. Demazeau and J.-P. Müller, editors, *Decentralized AI 2 — Proceedings of the Second European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-90)*, pages 109–126. Elsevier/North Holland, 1991.

[140] D. Wilkins. *Practical Planning: Extending the Classical AI Planning Paradigm*. Morgan Kaufmann Publishers, Inc., 1988.

[141] T. Wittig, editor. *ARCHON: An Architecture for Multi-Agent Systems*. Ellis Horwood, 1992.

[142] S. Wood. *Planning and Decision Making in Dynamic Domains*. Ellis Horwood Ltd., 1993.

[143] M. Wooldridge. *The Logical Modelling of Computational Multi-Agent Systems*. PhD thesis, Department of Computation, UMIST, Manchester, UK, October 1992. (Also available as Technical Report MMU–DOC–94–01, Department of Computing, Manchester Metropolitan University, Chester St., Manchester, UK).

[144] M. Wooldridge. Coherent social action. In A. Cohn, editor, *Proceedings of the Eleventh European Conference on Artificial Intelligence (ECAI-94)*. John Wiley & Sons, August 1994.

[145] M. Wooldridge and M. Fisher. A first-order branching time logic of multi-agent systems. In B. Neumann, editor, *Proceedings of the Tenth European Conference on Artificial Intelligence (ECAI-92)*, pages 234–238. John Wiley & Sons, August 1992.

[146] M. Wooldridge and M. Fisher. A decision procedure for a temporal belief logic. In D. Gabbay and H.-J. Ohlbach, editors, *Proceedings of the First International Conference on Temporal Logic (ICTL-94)*. Springer-Verlag, July 1994. (To appear).

[147] M. Wooldridge and N. R. Jennings. Formalizing the cooperative problem solving process. In *Proceedings of the Thirteenth International Workshop on Distributed Artificial Intelligence (IWDAI-94)*, Lake Quinalt, WA, July 1994.

[148] A. Yonezawa, editor. *ABCL — An Object-Oriented Concurrent System*. The MIT Press, 1990.

Bibliographical Remarks

There are a number of sources for work on agent theories, architectures, and languages. The most obvious are the major international and national conferences on AI: the International Joint Conference on AI (IJCAI), held biannually in odd years, the European Conference on AI (ECAI), held biannually in even years, and the (American) National Conference on AI, organised by AAAI, which is held annually in the US except when IJCAI is held on the North American continent. In 1994, the eleventh ECAI conference will be held in Amsterdam, and the eleventh AAAI in Seattle, WA; in 1995, the fourteenth IJCAI will be held in Montreal, Canada. A brief look at the references on this paper will confirm that much of the work cited here appeared in these three conferences. The proceedings of all these conferences are readily available. Turning to more specialist conferences and workshops, there is the International Workshop on Distributed AI (IWDAI), which has been held more-or-less annually since 1979; the thirteenth such workshop will be held at Lake Quinalt, WA, in July 1994. Unfortunately, the proceedings of IWDAI are not published regularly, and can be difficult to get hold of. In Europe, the workshop on Modelling Autonomous Agents in Multi-Agent Worlds (MAAMAW) is held annually; the sixth such workshop will be held in Odense, Denmark, in August 1994. The MAAMAW proceedings are published regularly. Results of interest also appear in other workshops, for example the UK series on Cooperating Knowledge-Based Systems (CKBS), and the conferences on Cooperative Information Systems (CoopIS). The first International Conference on Multi-Agent Systems (ICMAS) will be held in San Francisco in July 1995; further details are not available at the time of writing. Turning specifically to theory, the fifth conference on Theoretical Aspects of Reasoning about Knowledge (TARK) was held in 1994. The European Workshop on Logics in AI (JELIA), and the International Conference on Knowledge Representation and Reasoning (KR&R) are other good sources of theory.

With respect to journals, there is obviously Artificial Intelligence; the IEEE Transactions on Systems, Man and Cybernetics has also had much useful material over the years. The only related specialist publication we know of is the the International Journal on Intelligent and Cooperative Information Systems (IJICIS).

The reader may be interested to note that a BIBTEX format database containing more than five-hundred related references (including all those that appear in this paper) is available on request from the first author. For convenience, this database comes with a POSTSCRIPT format file listing the citation keys of all the entries.