Logics for Multiagent Systems

Wiebe van der Hoek and Michael Wooldridge

■ *We present a brief survey of logics for reasoning about multiagent systems. We focus on two paradigms: logics for cognitive models of agency, and logics used to model the strategic structure of a multiagent system.*

Logic can be a powerful tool for reasoning about multiagent systems. First of all, logics provide a *language* in which to *specify* properties— properties of an agent, of other agents, and of the environment. Ideally, such a language then also provides a means to *implement* an agent or a multiagent system, either by somehow executing the specification, or by transforming the specification into some computational form. Second, given that such properties are expressed as logical formulas that form part of some *inference* system, they can be used to *deduce* other properties. Such reasoning can be part of an individual agent's capabilities, but it can also be done by a system designer or the potential user of the agents.

Third, logics provide a formal *semantics* in which the sentences from the language are assigned a precise meaning: if one manages to come up with a semantics that closely models the system under consideration, one then can *verify* properties either of a particular system (model checking) or of a number of similar systems at the same time (theorem proving). This, in a nutshell, sums up the three main characteristics of any logic (language, deduction, semantics), as well as the three main roles logics play in system development (specification, execution, and verification).

We typically strive for logics that strike a useful balance between *expressiveness* on the one hand and *tractability* on the other: what kind of properties are interesting for the scenarios of interest, and how can they be "naturally" and concisely expressed? How complex are the formalisms, in terms of how costly is it to use the formalism when doing verification or reasoning with them?

In multiagent research, this complexity often depends on a number of issues. Let us illustrate this with a simple example, say the modeling of a set of lifts. If there is only one agent (lift) involved, the kind of things we would like to represent to model the agent's sensing, planning and acting could probably be done in a simple propositional logic, using atoms like $pos(\mathbf{i}, n)$ (currently lift i is positioned at floor n), $callf(n)$ (there is a call from floor n), $callt(n$ (there is a call within the lift to go to destination n), $at(\mathbf{i}, n)$ (lift i is currently at floor n), $op(i)$ (the door of lift i is open) and $up(i)$ (the lift is currently moving up). However, taking the agent perspective seriously, one quickly realizes that we need more: there might be a difference between what is *actually* the case and what the agent *believes* is the case, and also between what the agent believes to hold and what he would *like* to be true (otherwise there would be no reason to act!). So we would like to be able to say things like $\neg callf(n) \wedge B_i callf(n) \wedge D_i(callf(n) \rightarrow up(i))$ (although there is no call from floor n, the agent believes there is one, and desires to establish that in that case the agent moves up).

Obviously, things get more interesting when several agents enter the scene. Our agent i needs not only a model of the environment but also a model of j's mental state, the latter involving a model of i's mental state. We can then express properties like $B_i B_j callf(n) \rightarrow \neg up(i)$ (if i believes that j believes there is call from floor n, i will not go up). Higher-order information enters the picture, and there is no a priori limit to the degree of nesting that one might consider (this is for instance important in reasoning about games: see the discussion on common knowledge for establishing a Nash equilibrium [Aumann and Brandenburger 1995]). In our simple lift scenario, if for some reason lift i would like not to go up to floor n, we might have $B_i callf(n) \wedge B_j callf(n) \wedge B_i B_j callf(n) \wedge D_i B_j \neg B_i callf(n)$ (although both lifts believe there is a call from floor n, i desires that j believe that i is not aware of this call).

Another dimension that complicates multiagent scenarios is their dynamics. The world changes, and the information, desires, and goals of the agents change as well. So we need tools to reason either about time, or else about actions explicitly. A designer of a lift is typically interested in properties like $\forall G \neg(up(i) \wedge op(i))$ (this is a safety property, requiring that in all computations, it is always the case that the lift is not going up with an open door), and $(callf(n) \rightarrow \forall F \bigvee_{i \in Ag} at(i, n))$ (a liveness property, expressing that if there is a call from floor n, one of the lifts will eventually arrive there). Combining such aspects of multiagents and dynamics is where things really become intriguing: there is not just "a future," or a "possible future depending on an agent's choice," but how the future will look depends on the choices of several agents at the same time. We will come across languages in which one can express the following.

Let δ denote the fact that i and j are at different floors: $\delta = \bigvee_{n \neq m,\, i \neq j} (at(i, n) \wedge at(m, j))$. Then $\neg\langle\langle i\rangle\rangle F\delta \wedge \neg\langle\langle j\rangle\rangle F\delta \wedge \langle\langle i, j\rangle\rangle F\delta$ expresses that although i cannot bring about that eventually both lifts are on different floors, and neither can j, by cooperating together, i and j can guarantee that they both in the end serve a different floor.

Logics for multiagent systems are typically *intensional* (in contrast to propositional and first-order logics, which are *extensional*). A logic is extensional if the truth-value of a formula is completely determined by the truth-value of all its components. If we know the truth-value of p and q, we also know that of $(p \wedge q)$, and of $\neg p \rightarrow (q \rightarrow p)$. For logics of agency, extensionality is often not realistic. It might well be that "rain in Oxford" and "rain in Liverpool" are both true, while our agent knows one without knowing the other. Even if one is given the truth value of p and of q, one is not guaranteed to be able to tell whether $B_i\,(p \vee q)$ (agent i believes that $p \vee q$), whether $F\,(p \wedge q)$ (eventually, both p and q), or whether $B_i\, G(p \rightarrow B_h q)$ (i believes that it is always the case that as soon as p holds, agent h believes that q also holds).

Those examples make clear why extensional logics are so successful for reasoning about multiagent systems. However, perhaps the most compelling argument for the use of modal logics for modeling the scenarios we have in mind lies in the *semantics* of modal logic. This is built around the notion of a "state," which can represent the state of a system, of a processor, or some situation. Considering several states at the same time is then rather natural, and usually, they are related: some because they "look the same" for a given agent (they define the agent's beliefs), some because they are very attractive (they comprise the agent's desires), or some of them may represent some state of affairs in the future (they model possible evolutions of the system). Finally, some states are reachable only when certain agents take certain decisions (those states determine what coalitions can achieve). States and their relations are mathematically represented in Kripke models, to be defined shortly.

In the remainder of this section we demonstrate some basic languages, inference systems and semantics that are foundational for logics of agency. The rest of this overview is then organized along two main streams, reflecting the following two key trends in multiagent systems research: cognitive models of rational action and models of the strategic structure of the system.

The first main strand of research in logics for

Knowledge Axioms

*Kn*1 φ where φ is a propositional tautology
*Kn*2 $K_i\,(\varphi \rightarrow \psi) \rightarrow (K_i\varphi \rightarrow K_i\psi)$
*Kn*3 $K_i\,\varphi \rightarrow \varphi$
*Kn*4 $K_i\,\varphi \rightarrow K_i\,K_i\varphi$
*Kn*5 $\neg K_i\,\varphi \rightarrow K_i\neg K_i\varphi$

Rules of Inference

MP $\vdash \varphi, \vdash (\varphi \rightarrow \psi) \Rightarrow \vdash \psi$
Nec $\vdash \varphi\;) \Rightarrow \vdash K_i\,\varphi$

Figure 1. An Inference System for Knowledge.

multiagent systems, *cognitive models of rational action,* focuses on the issue of representing the attitudes of agents within the system: their beliefs, aspirations, intentions, and the like. The aim of such formalisms is to derive a model that predicts how a rational agent would go from its beliefs and desires to actions. Work in this area builds largely on research in the philosophy of mind. The logical approaches presented in the section on cognitive states focus on this trend.

The second main strand of research, models of the strategic structure of the system, focuses on the strategic structure of the environment: what agents can accomplish in the environment, either together or alone. Work in this area builds on models of effectivity from the game theory community, and the models underpinning such logics are closely related to formal games. In the section "Representing the Strategic Structure," presented later on in this article, we present logics that deal with this trend.

A Logical Toolkit

We now very briefly touch upon the basic logics to reason about knowledge, about time and about action. Let *i* be a variable over a set of agents $Ag = \{1, \ldots, m\}$. For reasoning about knowledge or belief of agents (we will not dwell here on their distinction), one usually adds, for every agent *i,* an operator K_i to the language, where $K_i\varphi$ then denotes that agent *i* knows φ. In the best-known epistemic logics (Fagin et al. 1995; Meyer and van der Hoek 1995), we see the axioms as given in figure 1.

This inference system is often referred to as *S*5. Axiom *Kn*1 is obvious, *Kn*2 denotes that an agent can perform deductions upon what he or she knows, *Kn*3 is often referred to as veridicality: what one knows is true. If *Kn*3 is replaced by the weaker constraint *Kn*3′, saying $\neg K_i\bot$, the result is a logic for belief (where "*i* believes φ" is usually written $B_i\varphi$) called KD45. Finally, *Kn*4 and *Kn*5 denote positive and negative introspection, respectively: they state that an agent knows what it knows (*Kn*4) and knows what it doesn't know (*Kn*5), respectively. Modus ponens (MP) is a standard logical rule, and necessitation (Nec) guarantees that it is derivable that agents know all tautologies.

Moving on to the semantics of such a logic, models for epistemic logic are tuples $M = \langle S, R_{i\in Ag}, V\rangle$ (also known as Kripke models), where *S* is a set of states, $R_i \subseteq S \times S$ is a binary relation for each agent *i*, and $V : At \rightarrow S$ gives for each atom $p \in At$ the states $V(p)$ where *p* is true. The fact that $(s, s') \in R_i$ is taken to mean that *i* cannot tell states *s* and *s*′ apart. The truth of φ in a model *M* with state *s,* written as $M, s \models \varphi$, is standard for the classical connectives (compare with figure 2), and the clause $M, s \models K_{i\varphi}$ means that for all *t* with $R_ist, M,t \models \varphi$ holds. In other words, in state *s* agent *i* knows φ iff φ is true in all states *t* that are indistinguishable to *s* for *i*. K_i is called the *necessity operator* for R_i. $M \models \varphi$ means that for all states $s \in S$, $M, s \models \varphi$. Let ***S*5** be all models in which each R_i is an equivalence relation. Let ***S*5** $\models \varphi$ mean that in all models $M \in$ ***S*5**, we have $M \models \varphi$. The system *S*5 is complete for the validities in *S*5, that is, for all φ, $S5 \vdash \varphi$ iff ***S*5** $\models \varphi$.

Notice that it is possible to formalize *group* notions of knowledge, such as $E\varphi$ ("everybody knows φ," that is, $K_1\varphi \wedge \ldots \wedge K_m\varphi$), $D\varphi$ ("it is distributed knowledge that φ," that is, if you would pool all the knowledge of the agents together, φ would follow from it, like in $(K_i\,(\varphi_1 \rightarrow \varphi_2) \wedge K_j\,\varphi_1) \rightarrow D\varphi_2)$, and $C\varphi$ ('it is common knowledge that φ', which is axiomatized such that it resembles the infinite conjunction $E\varphi \wedge EE\varphi \wedge EEE\varphi \wedge \ldots$).

In epistemic logics, binary relations R_i on the set of states represent the agents' ignorance; in temporal logics, however, they represent the flow of time. In the most simple setting, we assume that time has a beginning, and advances linearly and discretely into an infinite future: this is linear-time temporal logic (LTL) (Pnueli 1977). So a simple model for time is obtained by taking as the set of states the natural numbers *N,* and for the accessibility relation "the successor of," that is, $R = \{(n, n + 1) : n \in \mathbf{N}\}$ and *V,* the valuation, can be used to specify specific properties in states. In the language, we then would typically see operators for the "next state," (*X*), for "all states in the future (*G*) and for "some time in the future" (*F*). The truth conditions for those operators, together with an axiom system for them, are given in figure 2. Note that *G* is the reflexive transitive closure of *R,* and *F* is its dual: $F\varphi = \neg G\neg\varphi$.

Often, one wants a more expressive language, adding for instance an operator (for "until"), saying $M, n \models \varphi U\psi$ iff $\exists m \geq n(M, m \models \psi\ \&\ \forall k(n \geq k \geq m \Rightarrow M, k \models \varphi))$.

A rational agent deliberates about choices, and to represent those, branching time seems a more appropriate framework than linear time. To understand branching time-operators though, an understanding of linear time operators is still of benefit. Computational tree logic (CTL), (Emerson 1990) is a branching time logic that uses pairs of operators; the first quantifies over paths, the second is an LTL operator over those paths. Let us demonstrate this by mentioning some properties that are true in the root ρ of the branching time model M of figure 3. Note that on the highlighted path, in ρ, the formula $G\neg q$ is true. Hence, on the branching model M, ρ, we have $\mathsf{E}G\neg q$, saying that in ρ, there exists a path through it, on which q is always false. $\mathsf{A}F\varphi$ means that on every path starting in ρ, there is some future point where φ is true. So, in ρ, $\mathsf{A}F\neg p$ holds. Likewise, $\mathsf{E}pUq$ is true in ρ because there is a path (the path "up," for example) in which pUq is true. We leave it to the reader to check that in ρ, we have $\mathsf{E}F(p \wedge \mathsf{A}G\neg p)$.

Notice that in LTL and CTL, there is no notion of *action:* the state transition structure describes how the world changes over time, but gives no indication of what causes transitions between states. We now describe frameworks in which we can be explicit about *how* change is brought about. The basic idea is that each state transition is labeled with an action — the action that causes the state transition. More generally, we can label state transitions with pairs (i, α), where i is an agent and α an action. The formalism discussed is based on dynamic logic (Harel, Kozen, and Tiuryn 2000).

Actions in the set Ac are either atomic actions ($a, b, \ldots$) or composed ($\alpha, \beta, \ldots$) by means of test of formulas ($\varphi?$), sequencing ($\alpha; \beta$), conditioning (if φ then α else β) and repetition (while φ do α). The informal meaning of such constructs is as follows:

> $\varphi?$ denotes a "test action" φ, while $\alpha; \beta$ denotes α followed by β. The conditional action if φ then α else β means that if φ holds, α is executed, else β. Finally, the repetition action while φ do α means that as long as φ is true, α is executed.

Here, the test must be interpreted as a test by the system; it is not a so-called knowledge-producing action (like observations or communication) that can be used by the agent to acquire knowledge.

These actions α can then be used to build new formulas to express the possible result of the execution of α by agent i (the formula $\langle do_i(\alpha)\rangle\varphi$ denotes that φ is a result of i's execution of α), and the opportunity for i to perform α (that is, $\langle do_i (\alpha)\rangle\top$). The formula $[do_i(\alpha)\}\varphi$ is shorthand for $\neg[do_i(\alpha)]\neg\varphi$, thus expressing that one possible result of performance of α by i implies φ. In the Kripke semantics, we then assume relations R_a for individual actions, where the relations for compositions are then recursively defined: for instance $R_{\alpha;\beta}$ st iff for some state u, $R_\alpha su$ and $R_\beta ut$. Indeed, $[do_i(\alpha)]$ is then the necessity operator for R_α.

Truth Conditions of LTL

$M, n \models p$	iff	$n \in V(p)$
$M, n \models \neg\varphi$	iff	not $M, n \models \varphi$
$M, n \models \varphi \wedge \psi$	iff	$M, n \models \varphi$ **and** $M, n \models \psi$
$M, n \models X\varphi$	iff	$M, n + 1 \models \varphi$
$M, n \models G\varphi$	iff	$\forall m \geq n, M, m \models \varphi$
$M, n \models F\varphi$	iff	$\exists m \geq n, M, m \models \varphi$

Figure 2. Semantics of Linear Temporal Logic.

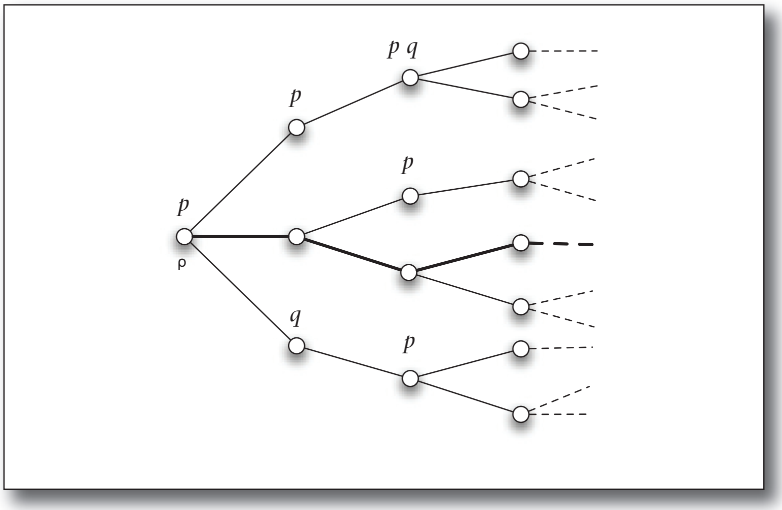

Figure 3. A Branching Time Model.

Representing Cognitive States

In attempting to understand the behavior of agents in the everyday world, we frequently make use of folk psychology, by which we predict and explain the behavior of agents with reference to their beliefs, desires, and intentions. The philosopher Dennett coined the phrase intentional system to refer to an entity that can be understood in terms of folk-psychology notions such as beliefs, desires, and the like (Dennett 1987). The intentional stance is essentially nothing more than an abstraction tool. If we accept the usefulness of the intentional stance for characterizing the properties of rational agents, then the next step in developing a formal theory of such agents is to identify the components of an agent's state. There are many possible mental states that we might choose to characterize an agent: beliefs, goals, desires, intentions, commitments, fears, hopes are just a few. We can identify several important categories of such attitudes, for example:

Information attitudes: those attitudes an agent has towards information about its environment. The most obvious members of this category are knowledge and belief.

Pro attitudes: those attitudes an agent has that tend to lead it to perform actions. The most obvious members of this category are goals, desires, and intentions.

Normative attitudes: including obligations, permissions and authorization.

Much of the literature on developing formal theories of agency has been taken up with the relative merits of choosing one attitude over another, and investigating the possible relationships between these attitudes.

Knowledge and Change

Having epistemic and dynamic operators, one has already a rich framework to reason about agent's knowledge about doing actions. For instance, a property like perfect recall

$$\mathbf{K}_i[do_i(\alpha)]\varphi \rightarrow [do_i(\alpha)]\mathbf{K}_i\varphi,$$

which semantically implies some grid structure on the set of states: If $R_\alpha st$ and $R_i tu$ then for some v, we also have $R_i sv$ and $\mathrm{R}_\alpha vu$. For temporal epistemic logic, perfect recall is characterized by the axiom $K_i X\varphi \rightarrow XK_i\varphi$, while its converse, no learning, is $XK_i \varphi \rightarrow K_i X\varphi$. It is exactly such interaction properties that can make a multiagent logic complex, both conceptually and computationally.

For studying the way that actions and knowledge interact, Robert Moore (1977, 1990) argued that one needs to identify two main issues. The first is that some actions produce knowledge, and therefore their effects must be formulated in terms of the epistemic states of participants. The second is that of knowledge preconditions: what an agent needs to know in order to be able to perform an action. A simple example is that in order to unlock a safe, one must know the combination for the lock. Using these ideas, Moore formalized a notion of ability. He suggested that in order for an agent to be able to achieve some state of affairs φ, the agent must either know the identity of an action α, (that is, have an "executable description" of an action α) such that after α is performed, φ holds; or else know the identity of an action α such that after α is performed, the agent will know the identity of an action α' such that after α' is performed, φ holds.

The point about "knowing the identity" of an action is that, in order for me to be able to become rich, it is not sufficient for me simply to know that there exists some action I could perform that would make me rich; I must either know what that action is (the first clause above), or else to able to perform some action that would furnish me with the information about which action to perform in order to make myself rich. This subtle notion of knowing an action is rather important, and it is related to the distinction between knowledge de re (which involves knowing the identity of a thing) and de dicto (which involves knowing that something exists) (Fagin et al. 1995, p. 101). In the safe example, most people would have knowledge de dicto to open the safe, but only a few would have knowledge de re.

It is often the case that actions are ontic: they bring about a change in the world, like assigning a value to a variable, moving a block, or opening a door. However, dynamic epistemic logic (DEL) (van Ditmarsch, van der Hoek, and Kooi 2007) studies actions that bring about mental change: change of knowledge in particular. So in DEL the actions themselves are epistemic. A typical example is publicly announcing φ in a group of agents: $[\varphi]\psi$ would then mean that after announcement of φ, it holds that ψ. Surprisingly enough, the formula $[\varphi]K_i\varphi$ (after the announcement that φ, agent i knows that φ), is not a validity, a counterexample being the infamous Moore (1942) sentences $\varphi = (\neg K_i\, p \wedge p)$: "although i does not know it, p holds"). Public announcements are a special case of DEL, which is intended to capture the interaction between the actions that an agent performs and its knowledge.

There are several variants of dynamic epistemic logic in the literature. In the language of van Ditmarsch, van der Hoek, and Kooi (2007), apart from the static formulas involving knowledge, there is also the construct $[\alpha]\varphi$, meaning that after execution of the epistemic action α, statement φ is true. Actions α specify who is informed by what. To express "learning," actions of the form $L_B\beta$ are used, where β again is an action: this expresses the fact that "coalition B learns that β takes place". The expression $L_B(\alpha\ !\ \beta)$, means the coalition B learns that either α or β is happening, while in fact α takes place.

To make the discussion concrete, assume we have two agents, 1 and 2, and that they commonly know that a letter on their table contains either the information p or $\neg p$ (but they don't know, at this stage, which it is). Agent 2 leaves the room for a minute, and, when he or she returns, is unsure whether or not 1 read the letter. This action would be described as

$$L_{12}(L_1\ ?p \cup L_1\ ?\neg p \cup\ !\top)$$

which expresses the following. First of all, in fact nothing happened (this is denoted by $!\top$). However, the knowledge of both agents changes: they commonly learn that 1 might have learned p, and he or she might have learned $\neg p$.

We now show how the example can be interpreted using the appealing semantics of Baltag and Moss (2004). In this semantics, both the uncertainty about the state of the world and that of the action taking place are represented in two independent Kripke models. The result of performing an epistemic action in an epistemic state is then computed as a "cross-product" (see figure 4). Model N in figure 4 represents that it is common knowledge among 1 and 2 that both are ignorant about p. The triangular-shaped model N is the action model that represents the knowledge and ignorance when L_{12} (L_1 ?$p \cup L_1$?$\neg p \cup$!T) is carried out. The points a, b, c of the model N are also called *actions,* and the formulas accompanying the name of the actions are called *preconditions:* the condition that has to be fulfilled in order for the action to take place. Since we are in the realm of truthful information transfer, in order to perform an action that reveals p, the precondition p must be satisfied, and we write pre(b) = p. For the case of nothing happening, only the precondition T need be true. Summarizing, action b represents the action that agent 1 reads p in the letter, action c is the action when $\neg p$ is read, and a is for nothing happening. As with "static" epistemic models, we omit reflexive arrows, so that N indeed represents that p or $\neg p$ is learned by 1, or that nothing happens: moreover it is commonly known between 1 and 2 that 1 knows which action takes place, while for 2 they all look the same

Now let $M, w = (W, R_1, R_2, \ldots R_m, \pi)$, w be a static epistemic state, and M, w an action in a finite action model. We want to describe what $M, w \oplus \mathsf{M}$, $\mathsf{w} = (W', R'_1, R'_2, \ldots R'_m, \pi')$, w', looks like — the result of "performing" the action represented by M, w in M, w. Every action from M, w that is executable in any state $v \in W$ gives rise to a new state in W': we let $W' = \{(v, \mathsf{v}) \mid v \in W, M, v \vDash \mathrm{pre}(\mathsf{v})\}$. Since epistemic actions do not change any objective fact in the world, we stipulate $\pi'(v, \mathsf{v}) = \pi(v)$. Finally, when are two states (v, v) and (u, u) indistinguishable for agent i? Well, agent i should be both unable to distinguish the originating states ($R_i uv$), and unable to know what is happening ($\mathsf{R}_i\mathsf{uv}$). Finally, the new state w' is of course (w, w). Note that this construction indeed gives $N, s \oplus \mathsf{N}$, $\mathsf{a} = N'$, (s, a), in our example of figure 4. Finally, let the action α be represented by the action model state M, w. Then the truth definition under the action model semantics reads that $M, w \vDash [\alpha]\varphi$ iff $M, w \vDash \mathrm{pre}(\mathsf{w})$ implies $(M, w) \oplus (\mathsf{M}, \mathsf{w}) \vDash \varphi$. In our example: $N, s \vDash [L_{12}\ (L_1\ ?p \cup L_1\ ?\neg p \cup\ !\mathrm{T})]\varphi$ iff $N', (s, \mathsf{a}) \vDash \varphi$.

Note that the accessibility relation in the resulting model is defined as

$$R_i\,(u, \mathsf{u})(v, \mathsf{v}) \Leftrightarrow R_i uv \ \&\ \mathsf{R}_i\mathsf{uv} \qquad (1)$$

This means that an agent cannot distinguish two states after execution of an action α, if and only if the agent could not distinguish the "sources" of those states, and he does not know which action exactly takes place. Put differently: if an agent knows the difference between two states s and t, then they can never look the same after performing an action, and likewise, if two indistinguishable actions α and β take place in a state s, they will give rise to new states that can be distinguished.

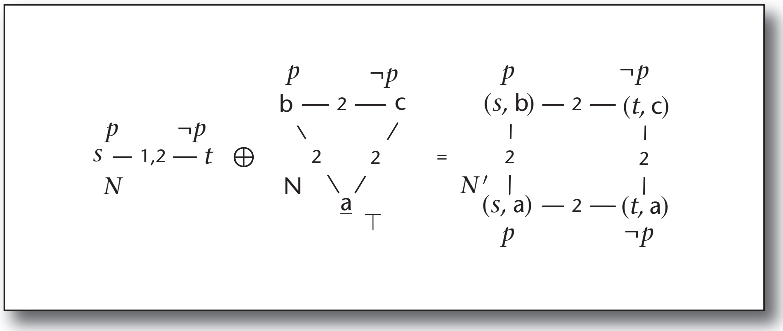

Figure 4. Computing ⊕.

Intention Logic

One of the best known, and most sophisticated attempts to show how the various components of an agent's cognitive makeup could be combined to form a logic of rational agency is discussed in Cohen and Levesque (1990). The logic has proved to be so useful for specifying and reasoning about the properties of agents that it has been used in an analysis of conflict and cooperation in multiagent dialogue (Galliers 1988), as well as in several studies in the theoretical foundations of cooperative problem solving (Levesque, Cohen, and Nunes 1990). This subsection will focus on the use of the logic in developing a theory of intention. The first step is to lay out the criteria that a theory of intention must satisfy.

When building intelligent agents — particularly agents that must interact with humans — it is important that a rational balance be achieved between the beliefs, goals, and intentions 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 forego 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 (Cohen and Levesque 1990, p. 214).

Following Bratman (1987), Cohen and Levesque identify seven specific properties that must be satisfied by a reasonable theory of intention. Given these criteria, they adopt a two-tiered approach to the problem of formalizing a theory of intention. First, they construct the logic of rational agency, "being careful to sort out the relationships among the basic modal operators" (Cohen and Levesque 1990, p. 221). On top of this framework, they introduce a number of derived constructs, which

Operator	Meaning
(Bel i φ)	agent i believes φ
(Goal i φ)	agent i has goal of φ
(Happens α)	action α will happen next
(Done α)	action α has just happened

Table 1. Atomic Modalities in Cohen and Levesque's Logic.

$$
\begin{aligned}
(\text{P-Goal}\ i\ p)\ \hat{=}\ & (\text{Goal}\ i\ (\text{Later}\ p)) \\
& (\text{Bel}\ i\ \neg p) \\
& \begin{bmatrix} \text{Before} \\ \quad ((\text{Bel}\ i\ p) \vee (\text{Bel}\ i\ G\neg p)) \\ \quad \neg(\text{Goal}\ i\ (\text{Later}\ p)) \end{bmatrix}
\end{aligned}
$$

Figure 5. Persistent Goal.

constitute a partial theory of rational action; intention is one of these constructs.

Syntactically, the logic of rational agency is a many-sorted, first-order, multimodal logic with equality, containing four primary modalities (see table 1). The semantics of Bel and Goal are given through 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. A world in this formalism is a discrete sequence of events, stretching infinitely into past and future. The system is only defined semantically, and Cohen and Levesque derive a number of properties from that. In the semantics, a number of assumptions are implicit, and one might vary them. For instance, there is a fixed domain assumption, giving us such properties as

$$\forall x(\text{Bel}\ i\ \varphi(x)) \rightarrow (\text{Bel}\ i\ \forall x\varphi(x)) \qquad (2)$$

The philosophically oriented reader will recognize a Barcan formula in equation 2, which in this case expresses that the agent is aware of all the elements in the domain. Also, agents "know what time it is," from which we immediately obtain the validity of formulas like 2.30pm/3/6/85 → (Bel i 2.30pm/3/6/85).

Intention logic has two basic operators to refer to actions, Happens and Done. The standard future time operators of temporal logic, "G" (always), and "F" (sometime) can be defined as abbreviations, along with a "strict" sometime operator, Later:

$$
\begin{aligned}
F\alpha &\triangleq \exists x \cdot (\text{Happens}\ x;\ \alpha?) \\
G\alpha &\triangleq \neg F \neg\alpha \\
(\text{Later}\ p) &\triangleq \neg p \wedge Fp
\end{aligned}
$$

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:

$$F\ \neg(\text{Goal}\ x\ (\text{Later}\ p))$$

The first major derived construct is a persistent goal (figure 5). 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; and (2) before it drops the goal, one of the following conditions must hold: (a) the agent believes the goal has been satisfied; or (b) 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 Cohen and Levesque [1990]). An agent i intends to perform action α if it has a persistent goal to have brought about a state where it had just believed it was about to perform α, and then did α.

$$
\begin{aligned}
(\text{Intend}\ i\ \alpha)\ \triangleq\ & (\text{P-Goal}\ i \\
& [\text{Done}\ i\ (\text{Bel}\ i\ (\text{Happens}\ \alpha))?;\ \alpha])
\end{aligned}
$$

Basing the definition of intention on the notion of a persistent goal, Cohen and Levesque are able to avoid overcommitment or undercommitment. An agent will only drop an intention if it believes that the intention has either been achieved, or is unachievable.

BDI Logic

One of the best-known approaches to reasoning about rational agents is the belief desire intention (BDI) model (Bratman, Israel, and Pollack 1988). The BDI model gets its name from the fact that it recognizes the primacy of beliefs, desires, and intentions in rational action.

Intuitively, an agent's *beliefs* correspond to information the agent has about the world. These beliefs may be incomplete or incorrect. An agent's desires represent states of affairs that the agent would, in an ideal world, wish to be brought about. (Implemented BDI agents require that desires be consistent with one another, although human desires often fail in this respect.) Finally, an agent's intentions represent desires that it has committed to achieving. The intuition is that an agent will not, in general, be able to achieve all its desires, even if these desires are consistent. Ultimately, an agent must therefore fix upon some subset of its desires and commit resources to achieving them.

These chosen desires, to which the agent has some commitment, are intentions (Cohen and Levesque 1990). The BDI theory of human rational action was originally developed by Michael Bratman (Bratman 1987). It is a theory of practical reasoning — the process of reasoning that we all go through in our everyday lives, deciding moment by moment which action to perform next.

There have been several versions of BDI logic, starting in 1991 and culminating in the work by Rao and Georgeff (1998); a book-length survey was written by Wooldridge (2000). We focus on the latter.

Syntactically, BDI logics are essentially branching time logics (CTL or CTL*, depending on which version one is reading about), enhanced with additional modal operators Bel, Des, and Intend, for capturing the beliefs, desires, and intentions of agents respectively. The BDI modalities are indexed with agents, so for example the following is a legitimate formula of BDI logic:

$$(\mathsf{Bel}\ i\ (\mathsf{Intend}\ j\ \mathsf{A}Fp)) \rightarrow (\mathsf{Bel}\ i\ (\mathsf{Des}\ j\ \mathsf{A}Fp))$$

This formula says that if i believes that j intends that p is inevitably true eventually, then i believes that j desires p is inevitable. Although they share much in common with Cohen and Levesque's intention logics, the first and most obvious distinction between BDI logics and the Cohen-Levesque approach is the explicit starting point of CTLlike branching time logics. However, the differences are actually much more fundamental than this. The semantics that Rao and Georgeff give to BDI modalities in their logics are based on the conventional apparatus of Kripke structures and possible worlds. However, rather than assuming that worlds are instantaneous states of the world, or even that they are linear sequences of states, it is assumed instead that worlds are themselves branching temporal structures: thus each world can be viewed as a Kripke structure for a CTLlike logic. While this tends to rather complicate the semantic machinery of the logic, it makes it possible to define an interesting array of semantic properties, as we shall see next.

We now summarize the key semantic structures in the logic. Instantaneous states of the world are modeled by time points, given by a set T; the set of all possible evolutions of the system being modeled is given by a binary relation $R \subseteq T \times T$. A world (over T and R) is then a pair (T', R'), where $T' \subseteq T$ is a nonempty set of time points, and $R' \subseteq R$ is a branching time structure on T'. Let W be the set of all worlds over T. A pair (w, t), where $w = (T_w, R_w) \in W$ and $t \in T_w$, is known as a *situation*. If $w \in W$, then the set of all situations in w is denoted by S_w. We have belief accessibility relations B, D, and I, modeled as functions that assign to every agent a relation over situations. Thus, for example:

$$B\ : Ag \rightarrow \wp(W \times T \times W)$$

We write $B(w, t, i)$ to denote the set of worlds accessible to agent i from situation (w, t): $B(w, t, i) = \{w' \mid (w, t, w') \in B(i)\}$. We define $D(w, t, i)$ and $I\ (w, t, i)$ in the obvious way. The semantics of belief, desire and intention modalities are then given in the conventional manner, for example,

$$(w, t) \models (\mathsf{Bel}\ i\ \varphi) \text{ iff } (w', t) \models \varphi \text{ for all } w' \in B(w, t, i).$$

The primary focus of Rao and Georgeff's early work was to explore the possible interrelationships between beliefs, desires, and intentions from the perspective of semantic characterization. For instance, we can also consider whether the intersection of accessibility relations is empty or not. For example, if $B(w, t, i) \cap I(w, t, i) \neq \emptyset$, for all i, w, t, then we get the following interaction axiom.

$$(\mathsf{Intend}\ \mathrm{i}\ \varphi) \rightarrow \neg(\mathsf{Bel}\ i\ \neg\varphi)$$

This axiom expresses an intermodal consistency property.

But because the worlds in this framework have a rich internal structure, we can undertake a more fine-grained analysis of the basic interactions among beliefs, desires, and intentions by considering this structure, and so we are also able to undertake a more fine-grained characterization of intermodal consistency properties by taking into account the structure of worlds. Without going in the semantic details, syntactically this allows one to have properties like $(\mathsf{Bel}\ i\ \mathsf{A}(\varphi)) \rightarrow (\mathsf{Des}\ i\ \mathsf{A}(\varphi))$: a relation between belief and desires that only holds for properties true on all future paths. Using this machinery, one can define a range of BDI logical systems (see Rao and Georgeff [1998], p. 321).

Representing the Strategic Structure

The second main strand of research that we describe focuses not on the cognitive states of agents, but on the strategic structure of the environment: what agents can achieve, either individually or in groups. The starting point for such formalisms is a model of strategic ability.

Over the past three decades, researchers from many disciplines have attempted to develop a general-purpose logic of strategic ability. Within the artificial intelligence (AI) community, it was understood that such a logic could be used in order to gain a better understanding of planning systems (Fikes and Nilsson 1971, Lifschitz 1986). The most notable early effort in this direction was Moore's dynamic epistemic logic, referred to earlier (Moore 1977, 1990). Moore's work was subsequently enhanced by many other researchers, perhaps most notably Morgenstern (1986). The distinctions made by Moore and Morgenstern also informed later attempts to integrate a logic of ability into more general logics of rational action in autonomous agents — see Wooldridge and Jennings (1995) for a survey of such logics.

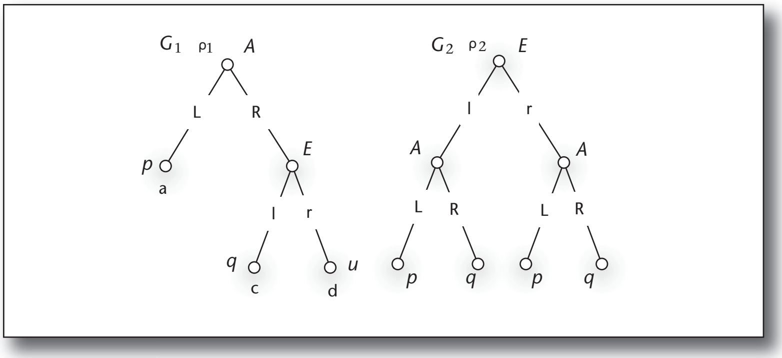

Figure 6. Two Games G_1 *and* G_2 *That Are the Same in Terms of Effectivity.*

In a somewhat parallel thread of research, researchers in the philosophy of action developed a range of logics underpinned by rather similar ideas and motivations. A typical example is that of Brown, who developed a logic of individual ability in the mid-1980s (Brown 1988). Brown's main claim was that modal logic was a useful tool for the analysis of ability, and that previous — unsuccessful — attempts to characterize ability in modal logic were based on an oversimple semantics. Brown's account (Brown 1988, p. 5) of the semantics of ability was as follows:

> [An agent can achieve *A*] at a given world iff there exists a relevant cluster of worlds, at every world of which *A* is true.

Notice the ∃∀ pattern of quantifiers in this account. Brown immediately noted that this gave the resulting logic a rather unusual flavor, neither properly existential nor properly universal (Brown 1988, p. 5):

> Cast in this form, the truth condition [for ability] involves two metalinguistic quantifiers (one existential and one universal). In fact, [the character of the ability operator] should be a little like each.

More recently, there has been a surge of interest in logics of strategic ability, which has been sparked by two largely independent developments: Pauly's development of coalition logic (Pauly 2001), and the development of ATL by Alur, Henzinger, and Kupferman (Alur, Henzinger, and Kupferman 2002; Goranko 2001). Although these logics are very closely related, the motivation and background to the two systems is strikingly different.

Coalition Logic

Pauly's coalition logic was developed in an attempt to shed some light on the links between logic – and in particular, modal logic – and the mathematical theory of games (Osborne and Rubinstein 1994). Pauly showed how the semantic structures underpinning a family of logics of cooperative ability could be formally understood as games of various types; he gave correspondence results between properties of the games and axioms of the logic, gave complete axiomatizations of the various resulting logics, determined the computational complexity of the satisfiability and model checking problems for his logics, and in addition, demonstrated how these logics could be applied to the formal specification and verification of social choice procedures. The basic modal operator in Pauly's logic is of the form $[C]\varphi$, where C is a set of agents (that is, a subset of the grand coalition Ag), and φ is a sentence; the intended reading is that "C can cooperate to ensure that φ."

The semantics of cooperation modalities are given in terms of an effectivity function, which defines for every coalition C the states that C can cooperate to bring about; the effectivity function $\mathsf{E} : S \rightarrow (\mathcal{P}(Ag) \rightarrow \mathcal{P}(\mathcal{P}(S)))$, gives, for any state t and coalition C a set of sets of end-states $\mathsf{E}_C(t)$, with the intended meaning of $S \in \mathsf{E}_C(t)$ that C can enforce the outcome to be in S (although C may not be able to pinpoint the exact outcome that emerges with this choice; this generally depends on the choices of agents outside C, or "choices" made by the environment). This effectivity function comes on a par with a modal operator $[C]$ with truth definition

$t \models [C\,]\varphi$ iff for some $S \in \mathsf{E}C(t)$:
for all $s(s \models \varphi$ iff $s \in S)$

In words: coalition is effective for, or can enforce φ if there is a set of states S that it is effective for, that is, which it can choose, which is exactly the denotation of φ: $S = [|\varphi|]$. It seems reasonable to say that C is also effective for φ if it can choose a set of states S that "just" guarantees φ, that is, for which we have $S \subseteq [|\varphi|]$. This will be taken care of by imposing monotonicity on effectivity functions: we will discuss constraints on effectivity at the end of this section.

In games and other structures for cooperative and competitive reasoning, effectivity functions are convenient when one is interested in the outcomes of the game or the encounter, and not so much about intermediate states, or how a certain state is reached. Effectivity is also a level in which on can decide whether two interaction scenarios are the same. The two games G_1 and G_2 from figure 6 are "abstract" in the sense that they do not lead to payoffs for the players but rather to states that satisfy certain properties, encoded with propositional atoms *p, q* and *u.* Such atoms could refer to which player is winning, but also denote other properties of an end-state, such as some distribution of resources, or "payments." Both games are two-player games: in G_1, player *A* makes the first move, which he choses from *L* (Left) and *R* (Right). In that game, player *E* is allowed to chose between *l* and *r,* respectively, but only if *A* plays *R:* otherwise the game ends after one move in the state satisfying *p.* In game G_2, both players have the same repertoire of choices, but the order in which the players choose is different. It looks like in G_1 player A can hand over control to E, while the converse seems to be true for G_2. Moreover, in G_2, the player who is not the initiator (that is, player A), will be allowed to make a choice, no matter the choice of his opponent.

Despite all these differences between the two games, when we evaluate them with respect to what each coalition can achieve, they are the same! To become a little more precise, let us define the powers of a coalition in terms of effectivity functions E. In game G_1, player A's effectivity gives E_A (ρ1) = {{a}, {c, d}}. Similarly, player E's effectivity yields {{a, c}, {a, d}}: player E can enforce the game to end in a or c (by playing l), but can also force the end-state among a and d (by playing r). Obviously, we also have $\mathsf{E}\{_{A,E}\}$(ρ1) = {{a}, {c}, {d}}: players A and E together can enforce the game to end in any end-state. When reasoning about this, we have to restrict ourselves to the properties that are true in those end states. In coalition logic, what we have just noted semantically would be described as:

$$G_1 \vDash \quad [A]p \wedge [A](q \vee u) \wedge [E](p \vee q) \wedge [E](p \vee u)$$
$$\wedge [A, E]p \wedge [A, E]q \wedge [A, E]r$$

Being equipped with the necessary machinery, it now is easy to see that the game G_2 verifies the same formula; indeed, in terms of what propositions can be achieved, we are in a similar situation as in the previous game: E is effective for $\{p, q\}$ (by playing l) and also for $\{p, u\}$ (play r). Likewise, A is effective for $\{p\}$ (play L) and for $\{q, u\}$ (play R). The alert reader will have recognized the logical law $(p \wedge (q \vee r)) \equiv ((p \wedge q) \vee (p \wedge u))$ resembling the "equivalence" of the two games: $(p \wedge (q \vee r))$ corresponds to A's power in G_1, and $((p \wedge q) \vee (p \wedge u))$ to A's power in G_2. Similarly, the equivalence of E's powers is reflected by the logical equivalence $(p \vee (q \wedge r)) \equiv ((p \vee q) \wedge (p \vee u))$.

At the same time, the reader will have recognized the two metalinguistic quantifiers in the use of the effectivity function E, laid down in its truth-definition. A set of outcomes S is in E_C iff for some choice of C, we will end up in S, under all choices of the complement of C (the other agents). This notion of so-called α-effectivity uses the ∃∀-order of the quantifiers: what a coalition can establish through the truth-definition above, their α-ability, is sometimes also called ∃∀-ability. Implicit within the notion of α-ability is the fact that C have no knowledge of the choice that the other agents make; they do not see the choice of C^c (that is, the complement of C), and then decide what to do, but rather they must make their decision first. This motivates the notion of β-ability (that is, "∀∃"-ability): coalition C is said to have the β-ability for φ if for every choice σ_D available to C^c, there exists a choice σ_C for C such that if C^c choose σ_D and C choose σ_C, then φ will result. Thus C being β-able to φ means that no matter what the other agents do, C have a choice such that, if they make this choice, then φ will be true. Note the "∀∃" pattern of quantifiers: C are implicitly allowed to make their choice while being aware of the choice made by C^c. We will come back to the pairs of α and β ability in the CL-PC subsection.

(⊥)	$\neg[C]\bot$
(N)	$\neg[\emptyset]\neg\varphi \rightarrow [Ag]\varphi$
(M)	$[C](\varphi \wedge \psi) \rightarrow [C]\psi$
(S)	$([C_1]\varphi_1 \wedge [C_2]\varphi_2) \rightarrow [C_1 \cup C_2](\varphi_1 \wedge \varphi_2)$ where $C_1 \cap C_2 = \emptyset$
(MP)	from φ and $\varphi \rightarrow \psi$ infer ψ
(Nec)	from φ infer $[C]\varphi$

Figure 7. The Axioms and Inference Rules of Coalition Logic.

We end this section by mentioning some properties of α-abilities. The axioms for $[C]\varphi$ based on α-effectivity (or effectivity, for short) are summarized in figure 7; see also Pauly (2002). The two extreme coalitions ∅ and the grand coalition Ag are of special interest. $[Ag]\varphi$ expresses that some end-state satisfies φ, whereas $[\emptyset]\varphi$ holds if no agent needs to do anything for φ to hold in the next state.

Some of the axioms of coalition logic correspond to restrictions on effectivity functions $\mathsf{E}: S \rightarrow (\mathcal{P}(Ag) \rightarrow \mathcal{P}(\mathcal{P}(\mathrm{S})))$. First of all, we demand that $\emptyset \notin \mathsf{E}_C$ (this guarantees axiom ⊥). The function E is also assumed to be monotonic: For every coalition $C \subseteq \mathrm{Ag}$, if $X \subseteq X' \subseteq S$, $X \in \mathsf{E}(C)$ implies $X' \in \mathsf{E}(C)$. This says that if a coalition can enforce an outcome in the set X, it also can guarantee the outcome to be in any superset X' of X (this corresponds to axiom (M)). An effectivity function E is C-maximal if for all X, if $X^c \notin \mathsf{E}(C^c)$ then $X \in \mathsf{E}(C)$. In words, if the other agents C^c cannot guarantee an outcome outside X (that is, in X^c), then C is able to guarantee an outcome in X. We require effectivity functions to be Ag-maximal. This enforces axiom (N) — Pauly's symbol for the grand coalition is N: if the empty coalition cannot enforce an outcome satisfying φ, then the grand coalition Ag can enforce φ. The final principle governs the formation of coalitions. It states that coalitions can combine their strategies to (possibly) achieve more: E is superadditive if for all X_1, X_2, C_1, C_2 such that $C_1 \cap C_2 = \emptyset$, $X_1 \in \mathsf{E}(C_1)$ and $X_2 \in \mathsf{E}(C_2)$ imply that $X_1 \cap X_2 \in \mathsf{E}(C_1 \cup C_2)$. This obviously corresponds to axiom (S).

Strategic Temporal Logic: ATL

In coalition logic one reasons about the powers of coalitions with respect to final outcomes. However, in many multiagent scenarios, the strategic considerations continue during the process. It would be interesting to study a representation language for interaction that is able to express the temporal differences in the two games G_1 and G_2 of figure 6. Alternating-time temporal logic (ATL) is intended for this purpose.

Although it is similar to coalition logic, ATL emerged from a very different research community, and was developed with an entirely different set of motivations in mind. The development of ATL is closely linked with the development of branching-time temporal logics for the specification and verification of reactive systems (Emerson 1990; Vardi 2001). Recall that CTL combines path quantifiers "A" and "E" for expressing that a certain series of events will happen on all paths and on some path respectively, and combines these with tense modalities for expressing that something will happen eventually on some path (*F*), always on some path (*G*) and so on. Thus, for example, using CTL logics, one may express properties such as "on all possible computations, the system never enters a fail state" ($\mathsf{A}G\neg \mathit{fail}$). CTLlike logics are of limited value for reasoning about multiagent systems, in which system components (agents) cannot be assumed to be benevolent, but may have competing or conflicting goals. The kinds of properties we wish to express of such systems are the powers that the system components have. For example, we might wish to express the fact that "agents 1 and 2 can cooperate to ensure that the system never enters a fail state." It is not possible to capture such statements using CTLlike logics. The best one can do is either state that something will inevitably happen, or else that it may possibly happen: CTL-like logics have no notion of agency.

Alur, Henzinger, and Kupferman developed ATL in an attempt to remedy this deficiency. The key insight in ATL is that path quantifiers can be replaced by cooperation modalities: the ATL expression $\langle\langle C\rangle\rangle\varphi$, where C is a group of agents, expresses the fact that the group C can cooperate to ensure that φ. (Thus the ATL expression $\langle\langle C\rangle\rangle\varphi$ corresponds to Pauly's $[C]\varphi$.) So, for example, the fact that agents 1 and 2 can ensure that the system never enters a fail state may be captured in ATL by the following formula: $\langle\langle 1, 2\rangle\rangle\, G\neg \mathit{fail}$. An ATL formula true in the root ρ_1 of game G_1 of figure 6 is $\langle\langle A\rangle\rangle\, \mathrm{X}\, \langle\langle E\rangle\rangle\, \langle Xq$: A has a strategy (that is, play R in ρ_1) such that in the next time, E has a strategy (play l) to enforce u.

Note that ATL generalizes CTL because the path quantifiers A ("on all paths… ") and E ("on some paths …") can be simulated in ATL by the cooperation modalities $\langle\langle \emptyset\rangle\rangle$ ("the empty set of agents can cooperate to …") and $\langle\langle Ag\rangle\rangle$ ("the grand coalition of all agents can cooperate to …").

One reason for the interest in ATL is that it shares with its ancestor CTL the computational tractability of its model-checking problem (Clarke, Grumberg, and Peled 2000). This led to the development of an ATL model-checking system called MOCHA (Alur et al. 2000).

ATL has begun to attract increasing attention as a formal system for the specification and verification of multiagent systems. Examples of such work include formalizing the notion of role using ATL (Ryan and Schobbens 2002), the development of epistemic extensions to ATL (van der Hoek and Wooldridge 2002), and the use of ATL for specifying and verifying cooperative mechanisms (Pauly and Wooldridge 2003).

A number of variations of ATL have been proposed over the past few years, for example to integrate reasoning about obligations into the basic framework of cooperative ability (Wooldridge and van der Hoek 2005), to deal with quantification over coalitions (Agotnes, van der Hoek, and Wooldridge 2007), adding the ability to refer to strategies in the object language (van der Hoek, Jamroga, and Wooldridge 2005), and adding the ability to talk about preferences or goals of agents (Agotnes, van der Hoek, and Wooldridge 2006b; 2006a).

CL-PC

Both ATL and coalition logic are intended as general-purpose logics of cooperative ability. In particular, neither has anything specific to say about the origin of the powers that are possessed by agents and the coalitions of which they are a member. These powers are just assumed to be implicitly defined within the effectivity structures used to give a semantics to the languages. Of course, if we give a specific interpretation to these effectivity structures, then we will end up with a logic with special properties. In a paper by van der Hoek and Wooldridge (2005b), a variation of coalition logic was developed that was intended specifically to reason about control scenarios, as follows. The basic idea is that the overall state of a system is characterized by a finite set of variables, which for simplicity are assumed to take Boolean values. Each agent in the system is then assumed to control some (possibly empty) subset of the overall set of variables, with every variable being under the control of exactly one agent. Given this setting, in the coalition logic of propositional control (CL-PC), the operator $F_C\varphi$ means that there exists some assignment of values that the coalition C can give to the variables under its control such that, assuming everything else in the system remains unchanged, then if they make this assignment, then φ would be true. The box dual $G_C\varphi$ is defined in the usual way with respect to the diamond ability operator F_C. Here is a simple example:

Suppose the current state of the system is that variables *p* and *q* are false, while variable *r* is true, and further suppose then agent 1 controls *p* and *r,* while agent 2 controls *q*. Then in this state, we have for example: $F_1(p \wedge r)$, $\neg F_1 q$, and $F_2(q \wedge r)$. Moreover, for any satisfiable propositional logic formula ψ over the variables *p, q,* and *r,* we have $F_{1,2}\psi$.

The ability operator F_C in CL-PC thus captures contingent ability, rather along the lines of "classical planning" ability (Lifschitz 1986): ability under the assumption that the world only changes by the actions of the agents in the coalition operator F_C. Of course, this is not a terribly realistic type of ability, just as the assumptions of classical planning are not terribly realistic. However, in CL-PC, we can define α effectivity operators $\langle\langle C\rangle\rangle_\alpha\varphi$, intended to capture something along the lines of the ATL $\langle\langle C\rangle\rangle X\varphi$, as follows:

$$\langle\langle C\rangle\rangle_\alpha\varphi \triangleq F_C G_D \varphi, \text{ where } D = C^c$$

Notice the quantifier alternation pattern ∃∀ in this definition, and compare this to our discussion regarding α- and β-effectivity. Building on this basic formalism, van der Hoek and Wooldridge (2005a) investigate extensions into the possibility of dynamic control, where variables can be "passed" from one agent to another.

Conclusion and Further Reading

In this article, we have motivated and introduced a number of logics of rational agency; moreover, we have investigated the roles that such logics might play in the development of artificial agents. We hope to have demonstrated that logics for rational agents are a fascinating area of study, at the confluence of many different research areas, including logic, artificial intelligence, economics, game theory, and the philosophy of mind. We also hope to have illustrated some of the popular approaches to the theory of rational agency.

There are far too many research challenges open to identify in this article. Instead, we simply note that the search for a logic of rational agency poses a range of deep technical, philosophical, and computational research questions for the logic community. We believe that all the disparate research communities with an interest in rational agency can benefit from this search.

We presented logics for MAS from the point of view of modal logics. A state-of-the-art book on modal logic was written by Blackburn, van Benthem, and Wolter (2006), and, despite its maturity, the field is still developing. The references Fagin et al. (1995) and Meyer and van der Hoek (1995) are reasonably standard for epistemic logic in computer science: the modal approach modeling knowledge goes back to Hintikka (1962) though. In practical agent applications, information is more quantitative then just binary represented as knowledge and belief though. For a logical approach to reasoning about probabilities see Halpern (2003).

In the Representing Cognitive States section, apart from the references mentioned, there are many other approaches: in the KARO framework (van Linder, van der Hoek, and Meyer 1998) for instance, epistemic logic and dynamic logic are combined (there is work on programming KARO agents [Meyer et al. 2001] and on verifying them [Hustadt et al. 2001]). Moreover, where we indicated in the logical toolkit above how epistemic notions can have natural "group variants," Aldewereld, van der Hoek, and Meyer (2004) define some group proattitudes in the KARO setting. And, in the same way as epistemics becomes interesting in a dynamic or temporal setting (see our toolkit), there is work on logics that address the temporal aspects and the dynamics of intentions as well (van der Hoek, Jamroga, and Wooldridge 2007) and, indeed, on the joint revision of beliefs and intentions (Icard, Pacuit, and Shoham 2010).

Where our focus in the section on cognitive states was on logics for cognitive attitudes, due to space constraints we have neglected the social attitude of norms. Deontic logic is another example of a modal logic with roots in philosophy with work by von Wright (1951), which models attitudes like permissions and obligations for individual agents. For an overview of deontic logic in computer science, see Meyer and Wieringa (1993), for a proposal to add norms to the social, rather than the cognitive aspects of a multiagent system, see, for example, van der Torre (2001).

There is also work on combining deontic concepts with for instance knowledge (Lomuscio and Sergot 2003) and the ATL-like systems we presented in the section on strategic structures: (Agotnes et al. 2009) for instance introduce a multidimensional CTL, where roughly, dimensions correspond with the implementation of a norm. The formal study of norms in multiagent systems has arguably set off with work by Shoham and Tennenholtz (1992b, 1992a). In normative systems, norms are studied more from the multiagent collective perspective, where questions arise like: which norms will emerge, why would agents adhere to them, when is a norm "better" than another one.

There is currently a flurry of activity in logics to reason about games (see van der Hoek and Pauly [2006] for an overview paper) and modal logics for social choice (see Daniëls [2010] for an example). Often those are logics that refer to the information of the agents ("players," in the case of games), and their actions ("moves" and "choices," respectively). The logics for such scenarios are composed from the building blocks described in this article, with often an added logical representation of preferences (van Benthem, Girard, and Roy 2009) or expected utility (Jamroga 2008).

References

Agotnes, T.; van der Hoek, W.; Rodríguez-Aguilar, J.; Sierra, C.; and Wooldridge, M. 2009. Multi-Modal CTL: Completeness, Complexity, and an Application. *Studia Logica* 92(1): 1–26.

Agotnes, T.; van der Hoek, W.; and Wooldridge, M. 2006a. On the Logic of Coalitional Games. In *Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems*. New York: Association for Computing Machinery.

Agotnes, T.; van der Hoek, W.; and Wooldridge, M. 2006b. Temporal Qualitative Coalitional Games. In *Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems*. New York: Association for Computing Machinery.

Agotnes, T.; van der Hoek, W.; and Wooldridge, M. 2007. Quantified Coalition Logic. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*. Menlo Park, CA: AAAI Press.

Aldewereld, H.; van der Hoek, W.; and Meyer, J.-J. 2004. Rational Teams: Logical Aspects of Multi-Agent Systems. *Fundamenta Informaticae* 63(2-3): 159–183.

Alur, R.; de Alfaro, L.; Henzinger, T. A.; Krishnan, S. C.; Mang, F. Y. C.; Qadeer, S.; Rajamani, S. K.; and Tasiran, S. 2000. MOCHA User Manual. Technical Report, Department of Electrical Engineering and Computer Science, University of California, Berkeley.

Alur, R.; Henzinger, T. A.; and Kupferman, O. 2002. Alternating-Time Temporal Logic. *Journal of the ACM* 49(5): 672–713.

Aumann, R., and Brandenburger, A. 1995. Epistemic Conditions for Nash Equilibrium. *Econometrica* 63(5): 1161–1180.

Baltag, A., and Moss, L. S. 2004. Logics for Epistemic Programs. *Synthese* 139(2): 165–224.

Blackburn, P.; van Benthem, J.; and Wolter, F., eds. 2006. *Handbook of Modal Logic*. Amsterdam: Elsevier.

Bratman, M. E.; Israel, D. J.; and Pollack, M. E. 1988. Plans and Resource-Bounded Practical Reasoning. *Computational Intelligence* 4(4): 349–355.

Bratman, M. E. 1987. *Intention, Plans, and Practical Reason*. Cambridge, MA: Harvard University Press

Brown, M. A. 1988. On the Logic of Ability. *Journal of Philosophical Logic* 17(1): 1–26.

Clarke, E. M.; Grumberg, O.; and Peled, D. A. 2000. *Model Checking*. Cambridge, MA: The MIT Press.

Cohen, P. R., and Levesque, H. J. 1990. Intention Is Choice with Commitment. *Artificial Intelligence* 42(2–3): 213–261.

Daniëls, T. R. 2010. Social Choice and the Logic of Simple. *Journal of Logic and Computation* 20(4).

Dennett, D. C. 1987. *The Intentional Stance*. Cambridge, MA: The MIT Press.

Drimmelen, G. 2003. Satisfiability in Alternating-time Temporal Logic. In *Eighteenth Annual IEEE Symposium on Logic in Computer Science* (LICS 2003), 208–217. Piscataway, NJ: Institute of Electrical and Electronics Engineers.

Emerson, E. A. 1990. Temporal and Modal Logic. In *Handbook of Theoretical Computer Science* Volume B: Formal Models and Semantics, ed. J. van Leeuwen, 996–107. Amsterdam: Elsevier.

Fagin, R.; Halpern, J. Y.; Moses, Y.; and Vardi, M. Y. 1995. *Reasoning About Knowledge*. Cambridge, MA: The MIT Press

Fikes, R. E., and Nilsson, N. 1971. STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving. *Artificial Intelligence* 2(3–4): 189–208.

Galliers, J. R. 1988. A Theoretical Framework for Computer Models of Cooperative Dialogue, Acknowledging Multi-Agent Conflict. Ph.D. Dissertation, Open University, UK.

Goranko, V. 2001. Coalition Games and Alternating Temporal Logics. In *Proceedings of the 8th Conference on Theoretical Aspects of Rationality and Knowledge,* 259–272. San Francisco: Morgan Kaufmann Publishers.

Halpern, J. 2003. *Reasoning about Uncertainty*. Cambridge, MA: The MIT Press.

Harel, D.; Kozen, D.; and Tiuryn, J. 2000. *Dynamic Logic*. Cambridge, MA: The MIT Press.

Hintikka, J. 1962. *Knowledge and Belief*. Ithaca, NY: Cornell University Press.

Hustadt, U.; Dixon, C.; Schmidt, R.; Fisher, J.-J. M.; and van der Hoek, W. 2001. Verification with the KARO Agent Theory (extended abstract). In *Formal Approaches to Agent-Based Systems, First International Workshop, FAABS 2000,* Lecture Notes in Computer Science 1871, 33–47. Berlin: Springer.

Icard, T.; Pacuit, E.; and Shoham, Y. 2010. Joint Revision of Beliefs and Intention. In *Proceedings of the 12th International Conference on Principles of Knowledge Representation and Reasoning,* 572–574. Menlo Park, CA: AAAI Press.

Jamroga, W. 2008. A Temporal Logic for Markov Chains. In *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems,* 697–704. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems.

Levesque, H. J.; Cohen, P. R.; and Nunes, J. H. T. 1990. On Acting Together. In *Proceedings of the 8th National Conference on Artificial Intelligence,* 94–99. Menlo Park, CA: AAAI Press.

Lifschitz, V. 1986. On the Semantics of STRIPS. In *Reasoning About Actions and Plans: Proceedings of the 1986 Workshop,* ed. M. P. Georgeff and A. L. Lansky, 1–10. San Mateo, CA: Morgan Kaufmann Publishers.

Lomuscio, A., and Sergot, M. 2003. Deontic Interpreted Systems. *Studia Logica* 75(1): 63–92.

Meyer, J.-J. C., and van der Hoek, W. 1995. *Epistemic Logic for AI and Computer Science*. Cambridge: Cambridge University Press.

Meyer, J.-J. C., and Wieringa, R. J., eds. 1993. *Deontic Logic in Computer Science — Normative System Specification*. New York: John Wiley & Sons.

Meyer, J.-J. C.; de Boer, F.; van Eijk, R.; Hindriks, K.; and van der Hoek, W. 2001. On Programming KARO Agents. *Logic Journal of the IGPL* 9(2): 245–256.

Moore, G. E. 1942. A Reply to My Critics. In *The Philosophy of G. E. Moore, The Library of Living Philosophers,* vol 4, 535–677. Evanston, Illinois: Northwestern University.

Moore, R. C. 1977. Reasoning about Knowledge and

Action. In *Proceedings of the 5th International Joint Conference on Artificial Intelligence.* Los Altos, CA: William Kaufmann, Inc.

Moore, R. C. 1990. A Formal Theory of Knowledge and Action. In *Readings in Planning,* ed. J. F. Allen, J. Hendler, and A. Tate. San Mateo, CA: Morgan Kaufmann Publishers, 480–519.

Morgenstern, L. 1986. A First-Order Theory of Planning, Knowledge, and Action. In *Proceedings of the 1st Conference on Theoretical Aspects of Reasoning about Knowledge,* 99–114. San Mateo, CA: Morgan Kaufmann Publishers.

Osborne, M. J., and Rubinstein, A. 1994. *A Course in Game Theory.* The MIT Press: Cambridge, MA.

Pauly, M., and Wooldridge, M. 2003. Logic for Mechanism Design — A Manifesto. Paper presented at the 2003 Workshop on Game Theory and Decision Theory in Agent Systems, Melbourne, Australia, 14 July.

Pauly, M. 2001. Logic for Social Software. Ph.D. Dissertation, University of Amsterdam. ILLC Dissertation Series 2001-10. Amsterdam, The Netherlands.

Pauly, M. 2002. A Modal Logic for Coalitional Power in Games. *Journal of Logic and Computation* 12(1): 149–166.

Pnueli, A. 1977. The Temporal Logic of Programs. In *Proceedings of the Eighteenth IEEE Symposium on the Foundations of Computer Science,* 46–57. Piscataway, NJ: Institute of Electrical and Electronics Engineers.

Rao, A. S., and Georgeff, M. 1998. Decision Procedures for BDI Logics. *Journal of Logic and Computation* 8(3):293–344.

Ryan, M., and Schobbens, P.-Y. 2002. Agents and Roles: Refinement in Alternating-Time Temporal Logic. In *Intelligent Agents VIII,* Lecture Notes in Computer Science 2333, ed. J. Meyer and M. Tambe, 100–114. Berlin: Springer.

Shoham, Y., and Tennenholtz, M. 1992a. Emergent Conventions in Multi-Agent Systems. In *Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning,* 225–231. San Francisco: Morgan Kaufmann.

Shoham, Y., and Tennenholtz, M. 1992b. On the Synthesis of Useful Social Laws for Artificial Agent Societies. In *Proceedings of the Tenth National Conference on Artificial Intelligence*. Menlo Park, CA: AAAI Press.

van Benthem, J.; Girard, P.; and Roy, O. 2009. Everything Else Being Equal: A Modal Logic Approach to Ceteris Paribus Preferences. *Journal of Philosophical Logic* 38(1): 83–125.

van der Hoek, W., and Pauly, M. 2006. Modal Logic for Games and Information. In *Handbook of Modal Logic,* ed. P. Blackburn, J. van Benthem, and F. Wolter, 1077–1148. Amsterdam: Elsevier.

van der Hoek, W., and Wooldridge, M. 2002. Tractable Multi-Agent Planning for Epistemic Goals. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems,* 1167–1174. New York: Association for Computing Machinery.

van der Hoek, W., and Wooldridge, M. 2005a. On the Dynamics of Delegation, Cooperation, and Control: A Logical Account. In *Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems,* 701–708. New York: Association for Computing Machinery

van der Hoek, W., and Wooldridge, M. 2005b. On the Logic of Cooperation and Propositional Control. *Artificial Intelligence* 164(1-2): 81–119.

van der Hoek, W.; Jamroga, W.; and Wooldridge, M. 2005. A Logic for Strategic Reasoning. In *Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems,* 157–153. New York: Association for Computing Machinery.

van der Hoek, W.; Jamroga, W.; and Wooldridge, M. 2007. Towards a Theory of Intention Revision. *Synthese* 155(2): 265–290.

van der Torre, L. 2001. Contextual Deontic Logic: Normative Agents, Violations and Independence. *Annals of Mathematics and Artificial Intelligence* 37(1): 33–63.

van Ditmarsch, H.; van der Hoek, W.; and Kooi, B. 2007. *Dynamic Epistemic Logic*. Berlin: Springer.

van Linder, B.; van der Hoek, W.; and Meyer, J.-C. 1998. Formalizing Abilities and Opportunities of Agents. *Fundameta Informaticae* 34(1, 2): 53–101.

Vardi, M. Y. 2001. Branching Versus Linear Time: Final Showdown. In *Tools and Algorithms for the Construction and Analysis of Systems,* Lecture Notes in Computer Science, Volume 2031, 1–22. Berlin: Springer-Verlag.

von Wright, G. 1951. Deontic Logic. Mind 60(237): 1–15.

Wooldridge, M., and Jennings, N. R. 1995. Intelligent Agents: Theory and Practice. The *Knowledge Engineering Review* 10(2): 115–152.

Wooldridge, M., and van der Hoeck, W. 2005. On Obligations and Normative Ability. *Journal of Applied Logic* 3(3–4): 396–420.

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

Zalta, E. N. *Stanford Encyclopedia of Philosophy.* Stanford, CA: Stanford University (plato.stanford.edu).

Wiebe van der Hoek is a professor in the Agent, Research and Technology (ART) Group in the Department of Computer Science, University of Liverpool, UK. He studied mathematics in Groningen, and earned his PhD at the Free University of Amsterdam. He is interested in logics for knowledge and interaction. He is currently associate editor of *Studia Logica*, editor of *Agent and Multi-Agent Systems,* and editor-in-chief of *Synthese*. He can be contacted at wiebe@csc.liv.ac.uk.

Michael Wooldridge is a professor of computer science at the University of Oxford, UK. His main research interests are in the use of formal methods of one kind or another for specifying and reasoning about multiagent systems, and computational complexity in multiagent systems. He is an associate editor of the *Journal of Artificial Intelligence Research* and the *Artificial Intelligence* journal and he serves on the editorial boards of the *Journal of Applied Logic, Journal of Logic and Computation, Journal of Applied Artificial Intelligence*, and *Computational Intelligence*. He can be contacted at mjw@cs.ox.ac.uk.