# A FIRST-ORDER BRANCHING TIME LOGIC OF MULTI-AGENT SYSTEMS

**Michael Wooldridge**[*]
Department of Computation
UMIST
PO Box 88, Sackville St
Manchester M60 1QD
U.K.
EMAIL: M.J.Wooldridge@umist.ac.uk

**Michael Fisher**[†]
Department of Computer Science
University of Manchester
Oxford Road
Manchester M13 9PL
U.K.
EMAIL: michael@cs.man.ac.uk

## Abstract

This paper presents a first-order branching time temporal logic that is suitable for describing and reasoning about a wide class of computational multi-agent systems. The logic is novel in that it supports reasoning about the beliefs, actions, goals, abilities and structure of groups of agents. A sound proof system for the logic is presented, and some short examples are given, showing how the logic might be used to specify desirable properties of multi-agent systems.

## 1 Introduction

This paper presents a logic that is suitable for describing and reasoning about multi-agent systems (MAS). We take a MAS to be one composed of a number of computational entities built along the lines of classical AI research, which communicate through point-to-point message passing. Our strategy is to construct an abstract formal theory of MAS, and then build a logic corresponding to this theory [Wooldridge, 1992]. Our theory of MAS is in the spirit of [Konolige, 1986], in that it is explicitly architectural — the architectural commitment is, however, slight. The approach might be contrasted with the "standard" possible worlds semantics for models of agency (e.g., [Cohen and Levesque, 1990]).

The logic we develop is based on a branching model of time. There are several good reasons for supposing that branching time temporal logics are appropriate for reasoning about cooperative systems:

1. MAS are inherently reactive (in the Pnuelian sense [Pnueli, 1986]), and temporal logic has been shown to be well-suited for reasoning about such systems;

2. the semantic structures which underly branching time logics closely resemble *game trees*, the extensive representation formalism developed by game theorists for describing game-like multi-agent interactions (see, for example, [Ladner and Reif, 1986]);

3. branching time logics have certain inherent advantages over their linear counterparts, for example they allow the consideration of what *might* happen, rather than just what *actually* happens — this allows us to reason about, for example, the ability and goals of a group of agents.

This paper is structured as follows. The theory of MAS is outlined in §2. Theories of cooperative ability and cooperative goals are developed in §3. The logic QBAL (Quantified Branching Agent Logic) is developed in §4 and some short examples of the use of QBAL to specify desirable

[*] Michael Wooldridge was supported by an SERC PhD quota award.

[†] Michael Fisher supported by SERC Research Grant GR/H/18449.

properties of MAS are presented in §5. The paper concludes with a brief discussion and review of related work.

## 2 Outline of a Theory of MAS

In this section we outline the theory of MAS that underpins the logic we develop. Space restrictions mean we can do no more than give a brief summary of the theory; proofs and technical definitions are omitted. The interested reader is urged to consult [Wooldridge, 1992] for details.

### Agents

The basic components of the theory are *agents*. Each agent possesses a set of *beliefs*, which are explicitly represented in some internal language $L$ — this language is assumed to be at least a first-order logical language with a well defined domain. Agents reason about their beliefs by employing *deduction rules*. This is the model of belief proposed by Konolige [Konolige, 1986].

Agents are able to perform two kinds of action: *cognitive* (employing computational resources — e.g., a database agent performing a "retrieve" operation), and *communicative* (sending messages). The result of performing an action is an *epistemic input* [Gärdenfors, 1988]. An epistemic input is a "new piece of evidence" which an agent may subsequently believe. Epistemic inputs have two sources: performing a cognitive action generates an epistemic input for the actor; sending a message causes an epistemic input for the recipient.

A cognitive action $\alpha$ is simply a function from belief sets to epistemic inputs. Communicative acts are modeled as the exchange of messages — we assume that message passing is point-to-point, and that delivery is guaranteed. In order to "route" messages to the intended recipient, each agent is assigned a unique agent identifier. A message is a triple $\langle i, j, \xi \rangle$, where $i$ is the sender, $j$ is the recipient agent id, $i \neq j$, and $\xi$ is the content. The content of a message is a formula of a common communication language, which we usually take to be either equal to, or a subset of, the internal language $L$. The effect of receiving a message is modeled by giving each agent an *interpretation*: an interpretation is a function from belief sets and messages to epistemic inputs. Epistemic inputs are incorporated into a belief set via an *epistemic commitment function* (ECF) — an ECF takes a belief set and set of epistemic inputs, and returns a new belief set (which isn't quite as suggested in [Gärdenfors, 1988]). Once an agent has processed its epistemic inputs, it derives the closure of its beliefs under its deduction rules.

Actions cannot always be successfully applied. For this reason, actions are associated with conditions, the condition dictating when the action is applicable. A condition/action pair is a *rule*. A condition is a formula of the internal language, and is satisfied if it is believed. Each agent will possess a set of rules.

### Environments

An *environment* $\mathsf{E}$ is just a collection of interacting agents. We make the simplifying assumption that agents act in synchrony. We define the *state*, $\sigma$, of an environment to be a map from agent ids to belief sets — thus $\sigma(i)$ gives the belief set of agent $i$. The *initial state* of an environment is that where each agent has its initial belief set, closed under its deduction rules.

States change by virtue of each agent performing a *move*. A move is a tuple of actions, one of each type (cognitive and communicative). On each "cycle" of the system, every agent chooses a move to perform. A move is *legal* just in case each of its components is. The cognitive action performed by an agent in a move is given by the selector function *action*. An agent's move alone does not uniquely define the next state of the environment, as its moves combine with those of others. A *transition*, $\tau$, is a map from agent ids to moves — thus $\tau(i)$ gives the move of agent $i$. A transition is legal just in case each of the moves it dictates are; the *possible transitions* of an environment are the legal transitions available to it. The set of all messages sent in a transition is given by the selector function *sent*.

A *world*, $w$, is a pair consisting of a state and the transition that caused that state. The type for worlds is $\mathsf{World}$. (In the initial world of an environment each agent is considered to have performed a "*nil*" move.)

### QBAL Frames

We can now define QBAL-frames, the precursors to model structures of the language QBAL.

**Definition 1** *A QBAL-frame is a pair $\langle W, R \rangle$ where: $W \subseteq \mathsf{World}$ is a non-empty set of worlds and $R \subseteq W \times W$ is a binary relation on $W$.*

It is possible to precisely define the conditions under which a QBAL-frame "models" an environment [Wooldridge, 1992]. If a QBAL-frame is a model of an environment, then we say it is *ordinary*. Ordinary QBAL-frames have two important properties:

- if $\langle W, R \rangle$ is an ordinary QBAL-frame then $R$ is total;
- if a transition is *possible* from some world in an ordinary frame, then the transition is actually realized in the frame.

Next we present some utility definitions. Let $W$ be a non-empty set, and $R$ be a total binary relation on $W$. A *path*, $q$, on $W, R$ is an infinite sequence $(w_u : u \in \mathbf{N})$ such that $w_0 \in W$ and $(w_u, w_{u+1}) \in R$, for all $u \in \mathbf{N}$. The *head* of a path $q = (w_0, w_1, \ldots)$, denoted *head*$(q)$, is the first element, $w_0$. A path, $q$, is *w-rooted* if, and only if, *head*$(q) = w$. The path obtained from $q$ by omitting the first $u$ elements is denoted $q^u$. The set of all paths on $W, R$ is denoted *paths*$(W, R)$. The paths we have defined are sometimes termed *fullpaths* [Emerson and Halpern, 1986].

## 3 Cooperative Ability and Cooperative Goals

This section shows how the semantic structures developed in previous sections can be used to examine *cooperative ability* and *cooperative goals*.

### Cooperative Ability

What does it mean for an agent (or group of agents) to have the ability to achieve a goal? We say an agent is able to achieve a goal if there is a plan telling the agent what to do such that if the agent follows the plan then the goal is *guaranteed* to be achieved. To use this definition, we must state what we mean by a "plan", and what we mean by a goal being "guaranteed". The AI planning community generally views a plan as a partially ordered sequence of actions. Rather than choose a literal representation, we fix on abstractions of plans called *strategies*, a concept developed by game theorists. A strategy can be thought of as a strong kind of abstract conditional plan.

Now, what do we mean by "guaranteed"? Take some world in a QBAL frame, some agent and some strategy. From that world, a set of *futures* emerge, the paths rooted in the world. On some of these futures, the moves performed by the agent will correspond to those "suggested" by the strategy. Call these paths the "futures of the strategy". If the goal is achieved in *every* future of the strategy then the goal is a necessary consequence of following the strategy. This is what we mean by a strategy guaranteeing a goal.

This definition rests on a rather subtle property of QBAL frames, that they contain all the possible legal ways a system might evolve. This can be shown formally; it follows from the properties of ordinary frames described in §2. Note that this definition of ability can be applied just as easily to groups of agents.

We model a strategy, $s$, as a function from belief sets to moves. A strategy is *sound* for an agent if, and only if, it never dictates an illegal move for the agent, and *holds* on a path just in case the moves dictated by the strategy correspond to those performed by the agent on the path. Associated with a strategy will be a set of *futures* — each future represents one possible way things could turn out if the strategy is followed. (This is what Werner calls "potential" [Werner, 1990].) We say a path $q$ is a *future* of a strategy $s$ if, and only if, $s$ holds on $q$. These definitions can be generalized to the multi-agent case by introducing two new objects into our theory. A group of agents, or *coalition*, is a collection of agent ids denoted by the symbol $G$. Joint strategies are just multi-agent strategies.

**Definition 2** *A* joint strategy *js for a group of agents G is a tuple $js = \langle s_1, \ldots, s_n \rangle$, where $s_i$ is a strategy for agent i, for all $i \in G$. A joint strategy is sound just in case each of its components is; a joint strategy holds on a path just in case each of its com-*

*ponents does; and a path is a future of a joint strategy just in case it is a future of each of its components.*

We assume a function, $J$, which returns the set of sound joint strategies for every group of agents. A strategy achieves a goal if the goal is a necessary consequence of following the strategy; necessary truths are trivially achieved. Use of the term "necessary" indicates (correctly) that there are possible worlds lurking in this definition. These possible worlds are actually paths, related by virtue of being futures of some joint strategy. To conclude, we say a group of agents can achieve some goal if there is a sound joint strategy for the group such that on every future of the strategy, the goal is achieved.

### Cooperative Goals

This section develops a theory of cooperative goals. This theory allows the observer of a system to *attribute* goals to agents or groups of agents. The theory does not posit the existence of distinguished cognitive "goal states", either in individual agents or groups of agents. It is not necessary to identify any direct representation of a goal in the *structure* of an agent: there is no internalized collection of possible worlds with a goal relation holding between them (cf. [Cohen and Levesque, 1990]), nor is there a "goal stack" (cf. [Georgeff and Lansky, 1987]). An agent — or group of agents — only has goals by virtue of our attributing them.

How are we to go about attributing goals? The idea we adopt was inspired by Seel [Seel, 1991]. The semantics of goals are developed via possible worlds, but rather than "gratuitously inflict" possible worlds onto our semantic structures, we instead look to our framework to see where such worlds might lie latent. If we can find a suitable candidate, then they can be used to give a semantics to goals.

The worlds that we pick out for the semantics of "goal" are paths in QBAL structures. The idea is that to attribute a goal to an agent you must look at what it has actually done. Doing involves choosing. Choices mean expressing a preference for the consequences of one move over another. Preferences are themselves dictated by goals. In making a choice, say choosing move $m_1$ over move $m_2$, an agent is preferring the consequences of $m_1$ to $m_2$. Crudely, an agent has a goal of the *necessary consequences* of its actions. Since the consequences of all moves are contained in a QBAL frame, it ought to be possible to somehow "decode" a path and deduce what goals an agent had.

We define what it means for worlds and paths to agree on the actions of an agent and group of agents.

**Definition 3** *Let $w, w'$ be worlds, and $i$ be an agent. Then $w$ and $w'$ agree on the actions of $i$ if, and only if, the move performed by $i$ in $w$ is the same as the move performed by $i$ in $w'$. The worlds agree on the actions of a group of agents if, and only if, they agree on the actions of all the agents in the group. Two paths $q, q'$ agree on the actions of $G$ if, and only if, every world $w_u$ in $q$ agrees on the actions of $G$ with every world $w'_u$ in $q'$, for all $u \in \mathbf{N}$.*

To conclude, we say an agent (group of agents) has a goal on some path, if on all futures that agree with path on the actions of the agent (group of agents), the goal is achieved.

## 4 The First Order Branching Time Logic QBAL

In this section we introduce the first order branching time temporal logic QBAL.

### Syntax

QBAL is an *hierarchical language*, in that it is built "on top of" the internal language, $L$.

**Definition 4** *The language of QBAL based on internal language $L$ contains the following symbols:*

1. *A denumerable set of constant symbols* $\mathsf{Const}$ *made up of the disjoint sets* $\mathsf{Const}_{\mathsf{Ag}}$ *(agent constants),* $\mathsf{Const}_{\mathsf{Ac}}$ *(action constants),* $\mathsf{Const}_{\mathsf{U}}$*, (constants for elements in the domain of $L$), and* $\mathsf{Const}_{G}$ *(group constants);*

2. *A denumerable set of individual variables* $\mathsf{Var}$*, made up of the sets* $\mathsf{Var}_{\mathsf{Ag}}$*,* $\mathsf{Var}_{\mathsf{Ac}}$*,* $\mathsf{Var}_{G}$*, and* $\mathsf{Var}_{\mathsf{U}}$*;*

3. *All formulae of $L$;*

4. *The symbols* $\{\top, \mathsf{Believe}, \mathsf{Send}, \mathsf{Performs}, =, \in, \mathsf{Can}, \mathsf{Goal}\}$*;*

5. *The propositional connectives* $\{\neg, \vee\}$;
6. *The temporal connectives* $\{\bigcirc, \mathcal{U}\}$, *and path quantifier* $\mathsf{A}$;
7. *The punctuation symbols* $\{), (, \cdot\}$;
8. *The quantifier symbols* $\{\forall, \exists\}$.

**Definition 5** *A* term *is either a variable or a constant. A term is of sort* $\mathsf{Ag}$, $\mathsf{Ac}$, $G$ *or* $\mathsf{U}$. *To indicate that a term* $t$ *is of sort* $s$ *we write* $t_s$.

**Definition 6** *The well-formed formulae (wff) of QBAL (based on* L*) are defined by the following rules.*[1]

1. *Let* $\xi$ *be a wff of* L*. The set of atomic formulae of QBAL, which represent a subset of the wff of QBAL, contains all formulae of the form:*

$$(t_s = t'_s) \qquad (t_{\mathsf{Ag}} \in t_G)$$
$$(\mathsf{Believe}\ t_{\mathsf{Ag}}\ \xi) \qquad (\mathsf{Send}\ t_{\mathsf{Ag}}\ t_{\mathsf{Ag}}\ \xi)$$
$$(\mathsf{Performs}\ t_{\mathsf{Ag}}\ t_{\mathsf{Ac}})$$

2. *If* $\phi$ *is a wff of QBAL, then the following are wff of QBAL:*

$$(\mathsf{Can}\ t_G\ \phi) \qquad (\mathsf{Goal}\ t_G\ \phi)$$

3. *If* $\phi$, $\psi$ *are wff of QBAL, then the following are also wff of QBAL:*

$$\neg\phi \quad \phi \vee \psi \quad \top \quad \mathsf{A}\phi \quad \phi\mathcal{U}\psi \quad \bigcirc\phi$$

4. *If* $\phi$ *is a wff of QBAL, with* $x$ *free in* $\phi$, *then the following are wff of QBAL:*

$$\forall x \cdot \phi \quad \exists x \cdot \phi$$

The temporal component of QBAL is based on the logic CTL* [Emerson and Halpern, 1986]; below, we introduce some derived temporal operators for QBAL.

$$\begin{array}{rcl}
\mathsf{E}\phi & \stackrel{\text{def}}{=} & \neg\mathsf{A}\neg\phi \\
\Diamond\phi & \stackrel{\text{def}}{=} & \top\mathcal{U}\phi \\
\Box\phi & \stackrel{\text{def}}{=} & \neg\Diamond\neg\phi \\
\phi\mathcal{W}\psi & \stackrel{\text{def}}{=} & \Box\phi \vee \phi\mathcal{U}\psi \\
\overset{\infty}{\Box}\phi & \stackrel{\text{def}}{=} & \Diamond\Box\phi \\
\overset{\infty}{\Diamond}\phi & \stackrel{\text{def}}{=} & \Box\Diamond\phi
\end{array}$$

The semantics of the basic connectives and operators is described below.

## Model Structures

As might be expected, QBAL models contain QBAL frames. Additionally, the semantics of QBAL require four non-empty sets of elements to appear in model structures, one for each sort: $\mathsf{Ag}$ — agents, $\mathsf{Ac}$ — actions, $2^{\mathsf{Ag}}$ — the set of groups of agents, and $\mathsf{U}$ — a universe of individuals (the domain of $L$). The *domain of quantification* $D$ is the union of these sets. Constants are assumed to be *standard names* for the objects they denote — this greatly simplifies the technical apparatus of quantification. A bijective map $I$ from $\mathsf{Const}$ to $D$ is an *interpretation for constant symbols*. The inverse $N$ of $I$ is a *naming map*, which assigns each element of $D$ a unique standard name. A map from $\mathsf{Var}$ to $D$ is a variable assignment. We define the following transformation on formulae of $L$.

**Definition 7** *Let* $V$ *be a variable assignment,* $N$ *be a naming map, and* $\xi$ *be an arbitrary formula of* L*. By* $\xi^{N,V}$ *we mean the formula obtained from* $\xi$ *by replacing every variable* $x$ *which occurs free in* $\xi$ *by* $N(V(x))$.

So every free variable is replaced by the standard name associated with the object the variable denotes. We define a function returning the denotation of a term.

**Definition 8** *Let* $I$ *be an interpretation,* $V$ *be a variable assignment, and* $t$ *be a term. The denotation of* $t$ *is given by* $[\![t]\!]_{I,V}$:

$$[\![t]\!]_{I,V} \stackrel{\text{def}}{=} \begin{cases} I(t) & \text{if } t \in \mathsf{Const} \\ V(t) & \text{if } t \in \mathsf{Var}. \end{cases}$$

*Where* $I, V$ *are understood we write* $[\![t]\!]$.

We now define QBAL models, over which formulae of QBAL are interpreted.

**Definition 9** *A QBAL-model* $M$ *is a structure* $M = \langle W, R, \mathsf{Ag}, \mathsf{Ac}, 2^{\mathsf{Ag}}, \mathsf{U}, I, J\rangle$ *where* $\langle W, R\rangle$ *is QBAL frame,* $\mathsf{Ag}, \mathsf{Ac}, 2^{\mathsf{Ag}}$ *and* $\mathsf{U}$ *are non-empty domains for the sorts* $\mathsf{Ag}, \mathsf{Ac}, G$ *and* $\mathsf{U}$ *respectively,* $I$ *interprets constant symbols and* $J$ *returns, for each group of agents, that group's set of sound joint strategies.*

As with frames, there is a close relationship between *models* and *environments*: this correspondence can be formalized in a *modeling* relation.

[1] The term "free in ..." is assumed to have its usual meaning.

We say a model for QBAL is *ordinary* if, and only if, it is the model of some environment. We are concerned *solely* with ordinary models, as these are the "models of" environments as they appear in our theory of MAS. In the following, the term "model" is assumed to mean "ordinary model".

### Semantics

We present the semantics of QBAL via the satisfaction relation "$\models$" in the usual way. Well-formed formulae (hereafter, referred to simply as "formulae") are interpreted over a triple consisting of a model, a variable assignment, and a reference path. (Recall that $\sigma$ maps agents to belief sets, $\tau$ maps agents to moves, *sent* returns the set of messages sent in a transition, and *action* returns the action performed by an agent in a move.) Where $w$ is a world, $\sigma_w$ and $\tau_w$ are the state and transition of $w$ respectively.

$M, V, q \models (\mathsf{Believe}\ t_{\mathsf{Ag}}\ \xi)$
iff $\xi^{V,N} \in \sigma_{head(q)}(\llbracket t_{\mathsf{Ag}} \rrbracket)$

$M, V, q \models (\mathsf{Performs}\ t_{\mathsf{Ag}}\ t_{\mathsf{Ac}})$
iff $action(\tau_{head(q)}(\llbracket t_{\mathsf{Ag}} \rrbracket)) = \llbracket t_{\mathsf{Ac}} \rrbracket$

$M, V, q \models (\mathsf{Send}\ t_{\mathsf{Ag}}\ t'_{\mathsf{Ag}}\ \xi)$
iff $\langle \llbracket t_{\mathsf{Ag}} \rrbracket, \llbracket t'_{\mathsf{Ag}} \rrbracket, \xi^{V,N} \rangle \in sent(\tau_{head(q)})$

The first operator is essentially the belief operator from Konolige's logic $L^{\mathrm{Bq}}$, simplified by the use of standard names (see [Konolige, 1986]); the second operator describes the performance of a cognitive act; and the third describes the communicative act. Using these operators the beliefs, actions and communications of a particular agent can be specified and reasoned about. For a detailed discussion of the use of such operators, see [Wooldridge, 1992]. In this paper, however, we will concentrate on the goals and abilities of *groups* of agents. Consequently, we introduce the following operators.

$M, V, q \models (\mathsf{Can}\ t_G\ \phi)$
iff there is a sound joint strategy in $J(\llbracket t_G \rrbracket)$ such that for all $head(q)$-rooted futures $q' \in paths(W, R)$ of the strategy, $M, V, q' \models \phi$

$M, V, q \models (\mathsf{Goal}\ t_G\ \phi)$
iff for all paths $q' \in paths(W, R)$ that agree with $q$ on the actions of $\llbracket t_G \rrbracket$, $M, V, q' \models \phi$

The operator Can is intended to match our intuitions about ability. The formula $(\mathsf{Can}\, G\ \phi)$ is read "the group $G$ can bring about a world where $\phi$ is true." The formula $(\mathsf{Goal}\ G\ \phi)$ is read "the group $G$ have a goal of achieving $\phi$." This operator is perhaps slightly more difficult to understand than Can, as it expresses a property of *paths*. It is an impartial assessment of what an agent's goals are if it follows a particular course of action. It is important to realize that Goal is *not* described by the equivalence $(\mathsf{Goal}\ G\ \phi) \Leftrightarrow \Diamond\phi$, as there will be some formulae that are satisfied the reference path but are *not* goals of the group.

$M, V, q \models (t_s = t'_s)$ iff $\llbracket t_s \rrbracket = \llbracket t'_s \rrbracket$

$M, V, q \models (t_{\mathsf{Ag}} \in t_G)$ iff $\llbracket t_{\mathsf{Ag}} \rrbracket \in \llbracket t_G \rrbracket$

The "=" operator is first order equality: its arguments must be of the same sort. The "$\in$" operator allows us to reason about the members of a coalition in a simple way: the formula $(i \in G)$ is read "$i$ is a member of the group $G$."

The operators $\vee$, $\neg$, and $\top$ have standard semantics; $\wedge$, $\Rightarrow$, $\Leftrightarrow$ and $\perp$ are defined as abbreviations in the usual way.

$M, V, q \models \mathsf{A}\phi$
iff for all $head(q)$-rooted paths $q'$ in $paths(W, R)$, $M, q' \models \phi$

The A operator is read "on all paths": $\mathsf{A}\phi$ will be satisfied if $\phi$ is satisfied on all paths that have the same head as the reference path. The E operator (recall that E is defined as $\neg\mathsf{A}\neg$) is read "on some path": $\mathsf{E}\phi$ will be satisfied if $\phi$ is satisfied on at least one path that has the same head as the reference path.

$M, V, q \models \bigcirc\phi$ iff $M, q^1 \models \phi$

$M, V, q \models \phi\,\mathcal{U}\,\psi$
iff $M, q^u \models \psi$ for some $u \in \mathbf{N}$ and $M, q^v \models \phi$ for all $0 \leq v < u$

These rules define the standard "future time" operators of discrete temporal logic. Thus $\bigcirc\phi$ is satisfied if $\phi$ is satisfied in the next state of the reference path, and $\phi\,\mathcal{U}\,\psi$ is satisfied if $\psi$ is satisfied at some future point on the path and $\phi$ is satisfied until that point. Of the derived operators, $\Box\ \phi$ is satisfied if $\phi$ is satisfied now and at all points along the reference path, $\Diamond\phi$ is satisfied if $\phi$ is satisfied at some future point on the reference path, and $\mathcal{W}$ is the "weak until" operator, with the same meaning as the $\mathcal{U}$ operator, except that its second argument need never be satisfied. Quantified formulae are interpreted using the following semantic rules.

$$
\begin{array}{l}
M, V, q \models \forall x \cdot \phi \\
\quad \text{iff } M, V \dagger \{x \mapsto d\}, q \models \phi \text{ for all } d \in D
\end{array}
$$

$$
\begin{array}{l}
M, V, q \models \exists x \cdot \phi \\
\quad \text{iff } M, V \dagger \{x \mapsto d\}, q \models \phi \text{ for some } d \in D
\end{array}
$$

*Satisfiability* and *validity* are defined in the normal way. A formula is satisfiable if it is satisfied in some model, variable assignment, and path, and valid if it is satisfied for all models, variable assignments and paths. This completes our presentation of the semantics of QBAL.

### Proof Theory

In this section, we present sound proof system for our logic, QBAL. We begin by noting that all axioms of classical propositional calculus are axioms of QBAL[2].

$\vdash \phi$ where $\phi$ is a propositional tautology (PL).

The following temporal axioms are taken from [Stirling, 1988].

$$
\begin{array}{l}
\vdash \Box(\phi \Rightarrow \psi) \Rightarrow (\Box\phi \Rightarrow \Box\psi) \\
\vdash \bigcirc(\phi \Rightarrow \psi) \Rightarrow (\bigcirc\phi \Rightarrow \bigcirc\psi) \\
\vdash \Box\phi \Rightarrow \bigcirc\phi \wedge \bigcirc\Box\phi \\
\vdash \Box(\phi \Rightarrow \bigcirc\phi) \Rightarrow (\bigcirc\phi \Rightarrow \Box\phi) \\
\vdash \mathsf{A}(\phi \Rightarrow \psi) \Rightarrow (\mathsf{A}\phi \Rightarrow \mathsf{A}\psi) \\
\vdash \mathsf{E}\phi \Rightarrow \mathsf{A}\ \mathsf{E}\phi \\
\vdash \mathsf{A}\bigcirc\phi \Rightarrow \bigcirc\mathsf{A}\phi
\end{array}
$$

We now turn to the axiomatization of the Can operator.

$$
\begin{array}{l}
\vdash (\mathsf{Can}\ G\ \phi \Rightarrow \psi) \Rightarrow ((\mathsf{Can}\ G\ \phi) \Rightarrow (\mathsf{Can}\ G\ \psi)) \\
\vdash \exists x \cdot (\mathsf{Can}\ x\ \phi) \Leftrightarrow \mathsf{E}\phi \\
\vdash \neg(\mathsf{Can}\ G\ \phi) \Rightarrow (\mathsf{Can}\ G\ \neg(\mathsf{Can}\ G\ \phi)) \\
\vdash \forall x \cdot (\mathsf{Can}\ x\ \phi) \Rightarrow \forall y \cdot (x \subseteq y) \Rightarrow (\mathsf{Can}\ y\ \phi)
\end{array}
$$

The first axiom is analogous to the $K$ axiom of modal logic. The second might be called a *principle of achievability*: if something is possible, then there is a group of agents that can achieve it (if something is satisfied on just one path then it requires the "grand coalition" of all agents to achieve it). The third axiom is analogous to the $5$ axiom of modal logic, while the fourth axiom might be called a *principle of added value*: any group to which an agent adds its efforts can achieve anything the agent can achieve on its own. This axiom is similar to the "superadditivity" principle in game theory (which states that the utility of a coalition is equal to the sum of the utilities of the member agents working in isolation). Finally, we give an axiomatization for the Goal operator.

$$
\begin{array}{l}
\vdash (\mathsf{Goal}\ G\ \phi \Rightarrow \psi) \Rightarrow ((\mathsf{Goal}\ G\ \phi) \Rightarrow (\mathsf{Goal}\ G\ \psi)) \\
\vdash (\mathsf{Goal}\ G\ \phi) \Rightarrow \phi \\
\vdash \neg(\mathsf{Goal}\ G\ \phi) \Rightarrow (\mathsf{Goal}\ G\ \neg(\mathsf{Goal}\ G\ \phi))
\end{array}
$$

We now present some inference rules for QBAL. We begin by noting that modus ponens is sound.

(MP) From $\vdash \phi$ and $\vdash \phi \Rightarrow \psi$ infer $\vdash \psi$

The following rules are from Stirling's axiomatization of CTL$^*$ [Stirling, 1988] (on which QBAL is based).

(Nec) From $\vdash \phi$ infer $\vdash \Box\phi$

(Gen) From $\vdash \phi$ infer $\vdash \mathsf{A}\phi$

The following is called the *attachment rule*, and allows us to make inferences about what agents believe (cf. [Konolige, 1986, pp34–35 and p62]).[3]

(AR) From $\vdash (\mathsf{Believe}\ i\ \Xi)$ and $\Xi \vdash_{\rho(i)} \lambda$
infer $\vdash (\mathsf{Believe}\ i\ \lambda)$

Necessary truths can be vacuously achieved — coalitions vacuously have goals of necessary truths.

(Can Nec) From $\vdash \phi$ infer $\vdash \forall x \cdot (\mathsf{Can}\ x\ \phi)$

(Goal Nec) From $\vdash \phi$ infer $\vdash \forall x \cdot (\mathsf{Goal}\ x\ \phi)$

## 5 Examples

We now present some short examples which demonstrate how QBAL can be used to specify desirable properties of cooperative MAS.

[2] QBAL actually has the axioms and inference rules of a many sorted first-order logic — here we focus on just the main components of proof theory.

[3] $\rho(i)$ returns the deduction rules of agent $i$ [Konolige, 1986].

The property of always being able to avoid doing $\phi$ is given by

$$\mathsf{A}\Box(\mathsf{Can}\ G\ \neg\phi). \tag{1}$$

Similarly, the idea that a group can avoid $\phi$ until $\phi$ becomes true is given by

$$\mathsf{A}((\mathsf{Can}\ G\ \neg\phi)\mathcal{U}\phi). \tag{2}$$

The idea of a group of agents not being opposed to some goal is given by WeakGoal, the dual of Goal.

$$(\mathsf{WeakGoal}\ G\ \phi) \overset{\text{def}}{=} \neg(\mathsf{Goal}\ G\ \neg\phi). \tag{3}$$

QBAL can be used to capture the idea of an agent being *indispensable* for some goal.

$$(\mathsf{Reqd}\ i\ \phi) \overset{\text{def}}{=} \forall x \cdot ((\mathsf{Can}\ x\ \phi) \Rightarrow (i \in x)). \tag{4}$$

Subsequent examples will make use of the following definition, intended to capture the idea of a group being *committed* to a goal.

$$(\mathsf{Commit}\ G\ \phi) \overset{\text{def}}{=} \mathsf{A}(\mathsf{Goal}\ G\ \phi). \tag{5}$$

Suppose the set $\{\phi_1, \cdots, \phi_n\}$ represent the *common goals* of the system. It is possible to specify that coalitions should form to achieve goals wherever possible, except where this would make other goals unachievable.

$$\mathsf{A}\Box \bigwedge_{1\le i\le n} \forall x \cdot \left[\begin{array}{l} (\mathsf{Can}\ x\ \phi_i) \quad \wedge \\ ((\mathsf{Commit}\ x\ \phi_i) \Rightarrow \\ \qquad\qquad \bigwedge_{1\le j\le n} \mathsf{E}\Diamond\phi_j\ ) \end{array}\right] \Rightarrow (\mathsf{Commit}\ x\ \phi_i)$$

By combining the Can and Goal operators with the Believe, Send, and Performs operators, it is possible to specify more complex properties of cooperative systems. Unfortunately, lack of space prevents us from presenting such examples here (the interested reader should refer to [Wooldridge, 1992]).

## 6 Discussion

QBAL is similar to Werner's language LT$\Box$CAN [Werner, 1990], which contains a CAN operator on which we have based the semantics of our Can. Werner also tries to capture a stronger notion of cooperative ability in an operator COOPCAN, which we have not attempted to do. Although Werner's language is richer in this respect, it is not first order, does not allow reasoning about possible futures, and does not permit reasoning about the structure of problem solving groups. Moreover, the theory which underlies Werner's work is quite different to the one presented here.

The theory of MAS is intended to be general, and has been used to model a number of MAS, including [Shoham, 1990; Fisher and Barringer, 1991]. QBAL is only one of a family of temporal logics developed >from the theory of MAS outlined in §2. Full details of the theory and logics, together with numerous examples may be found in [Wooldridge, 1992].

To conclude, we have developed a logic which can be used to describe and reason about the time varying properties of multi-agent systems. This logic provides a versatile framework which can be used to specify the beliefs, actions, abilities, and goals of agents and groups of agents within cooperative problem-solving systems.

## Acknowledgements

Michael Wooldridge would like to thank Greg O'Hare for his support and encouragement throughout this research.

# References

[Cohen and Levesque, 1990] P. Cohen and H. Levesque. Intention is Choice With Commitment. *Artificial Intelligence*, 42, 1990.

[Emerson and Halpern, 1986] 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), 1986.

[Fisher and Barringer, 1991] M. Fisher and H. Barringer. Concurrent METATEM Processes — A Language for Distributed AI. In *European Simulation Multiconference*, Copenhagen, Denmark, June 1991.

[Gärdenfors, 1988] P. Gärdenfors. *Knowledge in Flux*. Bradford Books/MIT Press, 1988.

[Georgeff and Lansky, 1987] M. P. Georgeff and A. L. Lansky. Reactive Reasoning and Planning. In *Proceedings AAAI–87*. Morgan Kaufmann, 1987.

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

[Ladner and Reif, 1986] R. E. Ladner and J. H. Reif. The Logic of Distributed Protocols: preliminary report. In *Proceedings of the 1986 Conference on Theoretical Aspects of Reasoning About Knowledge*. Morgan Kaufmann, 1986.

[Pnueli, 1986] A. Pnueli. Specification and Development of Reactive Systems. In *Information Processing 86*. Elsevier/North Holland, 1986.

[Seel, 1991] N. Seel. A Framework for Agent Theory. In Y. Demazeau and J. P. Muller, editors, *Decentralized AI — Proceedings of the 2nd European Workshop on Modeling Autonomous Agents and Multi-Agent Worlds*. Elsevier/North Holland, 1991.

[Shoham, 1990] Y. Shoham. Agent Oriented Programming. Technical Report STAN–CS–1335–90, Dept. of Computer Science, Stanford University, Cal: USA, 1990.

[Stirling, 1988] C. Stirling. Completeness Results for Full Branching Time Logic. In *REX School/Workshop on Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency*, Noordwijkerhout, Netherlands, 1988.

[Werner, 1990] E. Werner. What Can Agents Do Together: A semantics of co-operative ability. In *Proceedings of the 1990 European Conference on AI*. Pitman, 1990.

[Wooldridge, 1992] M. J. Wooldridge. *(In Preparation)*. PhD thesis, Dept. of Computation, UMIST, Manchester: UK, October 1992.