A Logic for Strategic Reasoning

Wiebe van der Hoek
Dept. of Computer Science
University of Liverpool
Liverpool L69 7ZF, UK
wiebe@csc.liv.ac.uk

Wojciech Jamroga
Institute of Computer Science
Clausthal Univ. of Technology
38 678 Clausthal, Germany
wjamroga@in.tu-clausthal.de

Michael Wooldridge
Dept. of Computer Science
University of Liverpool
Liverpool L69 7ZF, UK
mjw@csc.liv.ac.uk

ABSTRACT

Rational strategic reasoning is the process whereby an agent reasons about the best strategy to adopt in a given multi-agent scenario, taking into account the likely behaviour of other participants in the scenario, and, in particular, how the agent's choice of strategy will affect the choices of others. We present CATL, a logic that is intended to facilitate such reasoning. CATL is an extension of Alternating-time Temporal Logic (ATL), which supports reasoning about the abilities of agents and their coalitions in game-like multi-agent systems. CATL extends ATL with a ternary *counterfactual commitment* operator of the form $C_i(\sigma, \varphi)$, with the intended reading "if it were the case that agent i committed to strategy σ, then φ". By using this operator in combination with the ability operators of ATL, it is possible to reason about the implications of different possible choices by agents. We illustrate the approach by showing how CATL may be used to express properties of games such as Nash equilibrium and Pareto efficiency. We also show that the model checking problem for CATL is tractable, and hence that efficient implementations of strategic reasoners based on CATL are feasible.

Categories and Subject Descriptors

I.2.11 [**Distributed artificial intelligence**]: multiagent systems; F.3.1 [**Logics and Meanings of Programs**]: Specifying and Verifying and Reasoning about Programs—*logics of programs*

General Terms

Theory

Keywords

multi-agent systems, strategic reasoning, modal logic, game theory

AAMAS'05, July 25-29, 2005, Utrecht, Netherlands.

1. INTRODUCTION

Strategic reasoning is commonplace in the literature of game theory and multi-agent systems. Strategic reasoning from the well-known one-shot prisoner's dilemma game is a good example here [18]:

> Suppose my opponent cooperates. Then my best response is to defect, since that way I get the best possible outcome. But suppose he defects. Then, again, I would get the best outcome by defecting.

Strategic reasoning of this kind is *counterfactual*, since it involves suppositions ('suppose he cooperates… suppose he defects …') that may be false or that may have an undetermined truth value [17, 22]. For example, the statement "if Napoleon won in 1812 then we all would speak French" is counterfactual, since it involves a supposition ("Napoleon won in 1812") that is in fact false. Classical logic is of no use when analysing such reasoning, since in classical logic, any implication with a false antecedent is by definition true. So, to capture strategic reasoning of the kind above – in which, in order to determine the best choice of action, we must make assumptions that may be true, false, or undetermined – we need some sort of counterfactual construction. Our aim in this paper is to present and evaluate a logic that supports precisely this kind of reasoning.

CATL (which stands for *Counterfactual* ATL, but it can be also read as *Commitment* ATL) is based on ATL, the Alternating-time Temporal Logic of Alur, Henzinger, and Kupferman [1, 2], a logic which supports reasoning about the abilities of agents and coalitions of agents in game-like multi-agent systems. CATL extends ATL with ternary *counterfactual commitment* operators, of the form $C_i(\sigma, \varphi)$, with the intended reading "if it were the case that agent i committed to strategy σ, then φ would hold". The $C_i(\sigma, \varphi)$ operators are counterfactual because they involve a supposition (that agent i commits to following strategy σ) which is not known to be true or false; we say they are *commitment* operators because they capture the notion of an agent committing to follow a particular strategy.[1] A formula $C_i(\sigma, \varphi)$ will be true in a state q of a system M iff φ is true at state q in the system M', where M' is exactly like M *except* that agent i is only able to perform the actions dictated by strategy σ.

Our work makes three key contributions to the area of logics for multi-agent systems:

- First, CATL is, to the best of our knowledge, the first logic

[1]Note that this is a rather different sense of the term commitment to that which is more commonly used in the multi-agent systems literature [16], in particular because commitment as represented in our counterfactual commitment operators is *irrevocable*. We present our preliminary approach to a modal logic of revocable commitments in another paper [15].

which combines reasoning about strategic ability with counterfactual reasoning.

- Second, although there has been previous work on logical characterisations of game-theoretic solution concepts, we believe that the combination of ability operators and the strategic counterfactual operator enables to express these properties much more elegantly and intuitively than has hitherto been possible.

- Third, our language extends ATL by introducing strategies as first-class components of the language, in much the same way that programs are first class components of the language of dynamic logic [12]. The resulting language not only enables one to reason about *what* coalitions can achieve, but also *how* they can achieve them. As we shall see in Section 4, the ability to name strategies explicitly within the language seems essential if we are to express properties such as Nash equilibrium.

The remainder of the paper is structured as follows. We begin by introducing *Action-based Alternating Transition Systems* (AATSs) which are used to give a semantics to CATL. Next, we describe the formal syntax and semantics of CATL and show that the model checking problem for CATL is tractable (i.e., can be solved in deterministic polynomial time). To illustrate the power of the logic, we introduce a simple formal model of games, and show how CATL can be used to reason about such games. In particular, we define a notion of correspondence between games and models in the logic, and show how game-theoretic concepts such as dominated strategies, Pareto optimality, and Nash equilibrium can be expressed as formulas of CATL. Finally, we present some conclusions. We do not include proofs of propositions and CATL properties due to lack of space.

2. ACTION-BASED ATS

Several semantic structures have been proposed for ATL, most of them equivalent (cf. [11]). As the notion of *action* plays such an important role in our framework, we find it convenient to work with yet another version of such structures, in which actions and action pre-conditions are first-class citizens. We refer to these structures as *Action-based Alternating Transition Systems* (AATSs), and emphasise that they are for most purposes equivalent to "conventional" ATL models. Formally, an AATS is a tuple:

$$M = \langle Q, q_0, \Phi, \pi, Ag, Ac_1, \ldots, Ac_n, \rho, \tau, \Upsilon_1, \ldots, \Upsilon_n, \|\cdot\| \rangle$$

where:

- Q is a non-empty (and usually finite) set of *states* of the system. We assume that, at any moment, the system is in one of the states;

- $q_0 \in Q$ is the *initial state*;

- Φ is a finite, non-empty set of *atomic propositions*;

- $\pi : Q \to 2^\Phi$ is an interpretation function, which gives the set of primitive propositions satisfied in each state: if $p \in \pi(q)$, then proposition p is true in state q;

- $Ag = \{1, \ldots, n\}$ is a finite, non-empty set of all *agents*. A *coalition* of agents is simply a subset of Ag, i.e. $G \subseteq Ag$, and set Ag is sometimes called the *grand coalition of agents*;

- Each agent $i \in Ag$ is associated with a set Ac_i of possible actions, and we assume that these sets of actions are pairwise disjoint. Formally, Ac_i is a finite, non-empty set of *actions*, for each $i \in Ag$, where $Ac_i \cap Ac_j = \emptyset$ for all $i \neq j \in Ag$;

  We denote the set of actions associated with a coalition $G \subseteq Ag$ by Ac_G, so $Ac_G = \bigcup_{i \in G} Ac_i$. A *joint action* for a coalition G is a tuple $\langle \alpha_1, \ldots, \alpha_k \rangle$, where $\alpha_i \in Ac_i$, for each $i \in G$. We denote the set of all joint actions for coalition G by J_G, so $J_G = \prod_{i \in G} Ac_i$. Given an element j of J_G and agent $i \in G$, we denote i's component of j by j_i.

- $\rho : Ac_{Ag} \to 2^Q$ is an *action precondition function*, which for each action $\alpha \in Ac_{Ag}$ defines the set of states $\rho(\alpha)$ from which α may be executed;

- $\tau : Q \times J_{Ag} \rightharpoonup Q$ is a partial *system transition function*, which defines the state $\tau(q, j)$ that would result by the performance of j from state q – note that, as this function is partial, not all joint actions are possible in all states (cf. the pre-condition function above). Note also that the function defines deterministic transitions: for a particular state q and a tuple of valid decisions from all the agents in q, the next state is completely determined;

- $\Upsilon_1, \ldots, \Upsilon_n$ are the sets of *strategy terms* for agents $1, \ldots, n$ respectively. We will define *strategies* for agents later in this section. For now, however, all we need to know about strategies is that we name them in formulae of CATL via strategy terms, and that for each agent $i \in Ag$, Σ_i will denote the set of strategies for agent i (as we will see below, given any model M, the set Σ_i will be well-defined). We call a strategy term from Υ_i, i.e. one that will be interpreted as a strategy for agent i, simply an *i-strategy term*. As with the sets of actions for agents, we assume that all sets Υ_i and Υ_j are disjoint for $i \neq j$, and define the set of all such terms $\Upsilon = \bigcup_{i \in Ag} \Upsilon_i$;

- $\|\cdot\|_M$: $\Upsilon \to (\bigcup_{i \in Ag} \Sigma_i)$ gives the denotation $\|\sigma\|_M$ of every strategy term $\sigma \in \Upsilon$ in model M. We will often omit the subscript M and just write $\|\sigma\|$.

We require that AATSs satisfy two coherence constraints:

1. *Non-triviality*. Agents always have at least one legal action: $\forall q \in Q, \forall i \in Ag, \exists \alpha \in Ac_i$ s.t. $q \in \rho(\alpha)$

2. *Consistency*. The ρ and τ functions agree on actions that may be performed: $\forall q \in Q, \forall j \in J_{Ag}, (q, j) \in \text{dom}\,(\tau)$ iff $\forall i \in Ag, q \in \rho(j_i)$

Given an agent $i \in Ag$ and a state $q \in Q$, we denote the actions available to i in q by $options(i, q)$, collecting all $\alpha \in Ac_i$ for which $q \in \rho(\alpha)$. We then say that a *strategy* for an agent $i \in Ag$ is a function: $\sigma_i : Q \to Ac_i$ which must satisfy the *legality* constraint that $\sigma_i(q) \in options(i, q)$ for all $q \in Q$. Thus, a strategy may be thought of as a conditional plan indicating how an agent is to act in any given state of the system. A *strategy profile* for a coalition $G = \{a_1, \ldots, a_k\} \subseteq Ag$ is a tuple of strategies $\langle \sigma_1, \ldots, \sigma_k \rangle$, one for each agent $a_i \in G$. We denote by Σ_G the set of all strategy profiles for coalition $G \subseteq Ag$; if $\sigma_G \in \Sigma_G$ and $i \in G$, then we denote i's component of σ_G by σ_G^i.

REMARK 1. *This is a deviation from the original semantics of* ATL *[2], where strategies assign agents' choices to* sequences *of states, which suggests that agents can recall the whole history of*

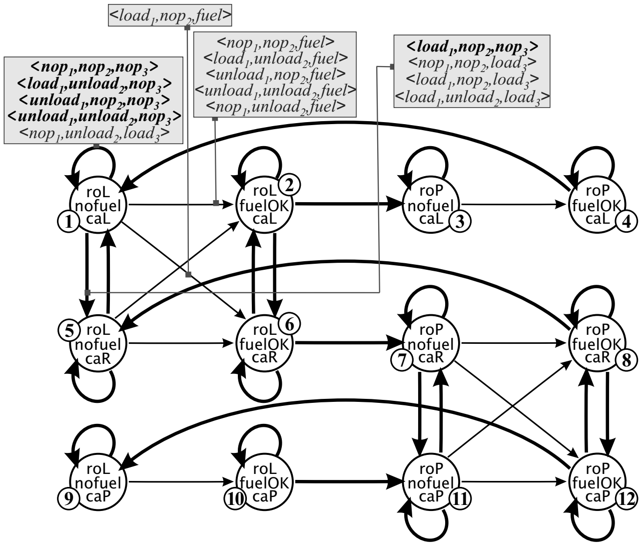

Figure 1: The Rocket Domain. The "bold" transitions are the ones in which agent 3 always chooses *nop*$_3$.

each game. In this paper, on the other hand, we employ "memoryless" strategies. While the choice of one or another notion of strategy affects the semantics of the full ATL* *and most* ATL *variants for games with incomplete information, perfect and imperfect recall strategies eventually yield equivalent semantics for the "pure"* ATL *[21].*

A *computation* is an infinite sequence of states $\lambda = q_0, q_1, \ldots$. A computation $\lambda \in Q^\omega$ starting in state q is referred to as a *q-computation*; if $u \in \mathbb{N}$, then we denote by $\lambda[u]$ the component indexed by u in λ (thus $\lambda[0]$ denotes the first element, $\lambda[1]$ the second, and so on). Given a strategy profile σ_G for some coalition G, and a state $q \in Q$, we denote by $comp(\sigma_G, q)$ the set of possible computations that may occur if every agent $a_i \in G$ follows the corresponding strategy σ_i, starting when the system is in state $q \in Q$. Notice that, for any grand coalition strategy profile σ_{Ag} and state q, the set $comp(\sigma_{Ag}, q)$ will be singleton.

2.1 A Running Example: The Rocket Domain

As an example, consider a modified version of the Simple Rocket Domain from [6]. The task is to ensure that a cargo eventually arrives in Paris (proposition caP), and there is a rocket that can be used to accomplish the task. Initially, the cargo is at the London airport (caL); during the game, it may also lie inside the rocket (caR). Accordingly, the rocket can be moved between London (roL) and Paris (roP).

There are three agents: 1 who can load the cargo, unload it, or move the rocket; 2 who can unload the cargo or move the rocket, and 3 who can load the cargo or supply the rocket with fuel. Every agent can also decide to do nothing at a particular moment (the *nop* – "no-operation" actions). The agents act simultaneously. The "moving" action has the highest priority (so, if one agent tries to move the rocket and another one wants to, say, load the cargo, then only the moving is executed). "Loading" is effected when the rocket does not move and more agents try to load than to unload. "Unloading" works in a similar way (in a sense, the agents "vote" whether the cargo should be loaded or unloaded). If the number of agents trying to load and unload is the same, then the cargo remains where it was. Finally, "fueling" can be accomplished only when the rocket tank is empty (alone or in parallel with loading or unloading). The rocket can move only if it has some fuel (fuelOK), and the fuel must be refilled after each flight. We assume that all the agents move with the rocket when it flies to another place. The AATS for the domain is shown in Figure 1 (we will refer to this system as M_1 throughout the rest of the paper). States of the system are labeled with natural numbers; we assume that the initial state is $q_0 = 1$. All the transitions for state 1 (the cargo and the rocket are in London, no fuel in the rocket) are labeled. Output of agents' choices for other states is analogous. We do not give the algebraic definition of M_1 here due to lack of space, but it can be easily extracted from the description.

2.2 Committing to Strategies

We now want to consider the idea of an agent *committing* to, or *choosing* a strategy. In committing to a strategy, an agent changes the structure of the AATS in which it is involved. This is because it *eliminates* certain possibilities from that structure: if agent i commits to σ, then in future it must choose actions that are consistent with σ. When *every* agent has made up its mind, the future of the system is determined: there will be just one possible computation of the system. To capture commitment formally, we introduce a commitment operation $\dagger_i$, where $M \dagger_i \sigma$ is the AATS obtained from M by eliminating from it all transitions in which agent i makes a choice that is not consistent with σ. Formally, if $M = \langle Q, q_0, \Phi, \pi, Ag, Ac_1, \ldots, Ac_n, \rho, \tau, \Upsilon_1, \ldots, \Upsilon_n, \|\cdot\|\rangle$ is an AATS, and σ is a strategy on M, then:

$$M \dagger_i \sigma = \langle Q, q_0, Ag, Ac_1, \ldots, Ac_n, \rho', \tau', \Phi, \pi, \|\|\|_{M'}\rangle,$$

where:

1. $\forall \alpha \in Ac_i$: $\rho'(\alpha) = \{q \mid \sigma(q) = \alpha\}$
2. $\forall q \in Q, \forall j \in J_{Ag}$:
$$\tau'(q, j) = \begin{cases} \tau(q, j) & \text{if } (q, j) \in \text{dom}\,\tau \;\&\; j_i = \sigma(q) \\ \text{undefined} & \text{otherwise} \end{cases}$$
3. $\Upsilon'_i = \{\sigma\}$
4. All other components of $M \dagger_i \sigma$ are as in M.

Thus the $\dagger_i$ operator represents an *update* on systems. Note, however, that this update does not delete *states*: only *transitions* between states. The operator is very similar to the model update operator already proposed in [14] for the implementation of social laws in ATL, and has essentially the same properties.

EXAMPLE 1. *Let σ be the "lazy" strategy for agent 3, i.e. $\sigma(q) = nop_3$ for every q. System $M_1 \dagger_3 \sigma$ includes only the transitions that are indicated with bold face font and thick arrows in Figure 1.*

3. CATL

Alternating-time Temporal Logic (ATL) [1, 2] can be understood as a generalisation of the well-known branching time temporal logic CTL [9], in which path quantifiers are replaced by *cooperation modalities*. A cooperation modality $\langle\langle G\rangle\rangle\varphi$, where G is a coalition, expresses that the coalition G can cooperate to ensure that φ; more precisely, that there exists a collective plan (*strategy profile*) for G such that by following this plan, G can ensure φ. Thus, for example, the system requirement "agents 1 and 2 can cooperate to ensure that the system never enters a fail state" may be captured by the ATL formula $\langle\langle 1, 2\rangle\rangle \Box\neg fail$. The $\Box$ temporal operator means "now and forever more"; other temporal connectives in ATL are $\mathcal{U}$ ("until") and $\bigcirc$ ("in the next state"). Additional operator $\Diamond$ ("either now or at some point in the future") can be defined as $\Diamond\varphi \mathrel{\hat{=}} \top\,\mathcal{U}\,\varphi$. Every occurrence of a temporal operator is preceded by exactly one cooperation modality in ATL (which is sometimes called "vanilla" ATL). The broader language of ATL*, in which no such restriction is imposed, is not discussed here. It is worth pointing out that CATL, proposed in this paper, makes use of terms that describe strategies, and in this sense is very different to ATL, in which strategies appear only in the semantics and are *not* referred to in the object language. In order to capture consequences of an agent's commitment to execute a particular strategy, we introduce a ternary modal operator $C_i(\sigma,\varphi)$ with the intended meaning: "suppose that agent i chooses the strategy denoted by σ; then φ holds". Having added formulas of this kind to ATL, we obtain a new logic that we call "Counterfactual ATL" or "ATL with Commitment" – CATL in short. Formally, the syntax of CATL, (with respect to a set of agents Ag, primitive propositions Φ, and strategy terms $\Upsilon = \bigcup_{i\in Ag}\Upsilon_i$), is given by the following grammar:

$$\varphi ::= p|\neg\varphi|\varphi\vee\varphi|C_i(\sigma_i,\varphi)|\langle\langle G\rangle\rangle\bigcirc\varphi|\langle\langle G\rangle\rangle\Box\varphi|\langle\langle G\rangle\rangle\varphi\,\mathcal{U}\,\varphi$$

where $p\in\Phi$ is a propositional variable, $i\in Ag$ is an agent, $G\subseteq Ag$ is a set of agents, and $\sigma_i\in\Upsilon_i$ is an i-strategic term. For reasons that will become clear shortly, we require that no i-strategy term τ_i occurs in φ in the formula $C_i(\sigma_i,\varphi)$. We now first define the semantics of CATL formulas, and then discuss strategic terms and their denotations.

3.1 Semantics of CATL

We now give the truth definition of CATL formulas on an AATS M and a state q:

$M,q\models p$ iff $p\in\pi(q)$ (where $p\in\Phi$);

$M,q\models\neg\varphi$ iff $M,q\not\models\varphi$;

$M,q\models\varphi\vee\psi$ iff $M,q\models\varphi$ or $M,q\models\psi$;

$M,q\models C_i(\sigma,\varphi)$ iff $(M\dagger_i\|\sigma\|),q\models\varphi$;

$M,q\models\langle\langle G\rangle\rangle\bigcirc\varphi$ iff $\exists\sigma_G\in\Sigma_G$, such that $\forall\lambda\in comp(\sigma_G,q)$, we have $M,\lambda[1]\models\varphi$;

$M,q\models\langle\langle G\rangle\rangle\Box\varphi$ iff $\exists\sigma_G\in\Sigma_G$, such that $\forall\lambda\in comp(\sigma_G,q)$, we have $M,\lambda[u]\models\varphi$ for all $u\in\mathbb{N}$;

$M,q\models\langle\langle G\rangle\rangle\varphi\,\mathcal{U}\,\psi$ iff $\exists\sigma_G\in\Sigma_G$, such that $\forall\lambda\in comp(\sigma_G,q)$, there exists some $u\in\mathbb{N}$ such that $M,\lambda[u]\models\psi$, and for all $0\le v<u$, we have $M,\lambda[v]\models\varphi$.

The other connectives ("$\wedge$", "$\rightarrow$", "$\leftrightarrow$") are assumed to be defined as abbreviations in terms of $\neg,\vee$. Also, $\langle\langle G\rangle\rangle\Diamond\varphi$ is shorthand for $\neg\langle\langle G\rangle\rangle\top\,\mathcal{U}\,\varphi$. We omit set brackets in cooperation modalities, writing $\langle\langle i,\ldots,k\rangle\rangle$ rather than $\langle\langle\{i,\ldots,k\}\rangle\rangle$. Validity and satisfiability are defined as usual for a modal logic: we write $\models\varphi$ to indicate that φ is valid.

Two cooperation modalities play a special role in the remainder of the paper, and are worth singling out for special attention. The cooperation modality $\langle\langle\rangle\rangle$ ("the empty set of agents can cooperate to...") asserts that its argument is true on all computations, and thus acts like CTL's universal path quantifier A. Similarly, the cooperation modality $\langle\langle Ag\rangle\rangle$ asserts that its argument is satisfied on at least one computation, and thus acts like the CTL path quantifier E.

The following example shows that sometimes, a coalition can achieve more if another agent commits himself to a strategy.

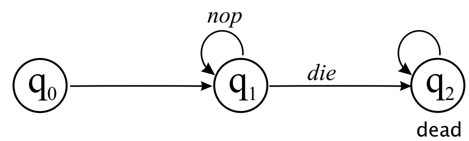

Figure 2: The single agent system

EXAMPLE 2. *Let nop_3 be the term denoting the "lazy" strategy for agent 3, i.e., the strategy in which he always chooses to do nothing. Then, $M_1,1\models C_3(nop_3,\langle\langle\rangle\rangle\Box\mathsf{roL})$, because the rocket will never move away from London in the system from Figure 1. Similarly, $M_1,1\models C_1(nop_1,\langle\langle 2,3\rangle\rangle\Diamond\mathsf{caP})$, although $M_1,1\models\neg\langle\langle 2,3\rangle\rangle\Diamond\mathsf{caP}$ when no commitment is considered.*

We can now explain why we forbid i-strategic terms τ_i to occur in φ, in the commitment formula $C_i(\sigma_i,\varphi)$. Conceptually this makes sense because, once i commits to strategy σ_i, there is no need to reason about other strategies of i anymore. Technically, recall that $M\dagger_i\sigma_i$ is the model that "cuts out" all transitions from M for i, that do not accord with σ_i. Hence, in that updated model, a strategy τ_i would not have an interpretation any more. Another option to deal with this would be to allow for partial strategies (cf. [15]), but for the moment we feel we can stick to our definition of full strategies as given in Section 2. Given a set of strategy terms Υ, we say that the AATS model M complies with Υ if every term $\sigma\in\Upsilon$ corresponds to a strategy $\|\sigma\|$ in M. It is then easy to verify that if M complies with Υ, and M' is the result of updating M in order to evaluate $C_i(\sigma_i,\varphi)$, then M' complies with the set Υ' of all strategies named in φ.

Note that, so far, we have been only describing individual commitments. Collective commitments can be defined on top of them:

$$C_{\{1\}}(\langle\sigma_1\rangle,\varphi) \mathrel{\hat{=}} C_1(\sigma_1,\varphi)$$
$$C_{\{1,..,k\}}(\langle\sigma_1,..,\sigma_k\rangle,\varphi) \mathrel{\hat{=}} C_1(\sigma_1,C_{\{2,..,k\}}(\langle\sigma_2,..,\sigma_k\rangle,\varphi))$$

This notion is well defined because of the following:

PROPOSITION 1. *Commitments are commutative:*

$$\models C_{a_1}(\sigma_{a_1},C_{a_2}(\sigma_{a_2},\varphi))\leftrightarrow C_{a_2}(\sigma_{a_2},C_{a_1}(\sigma_{a_1},\varphi)).$$

3.2 Properties under Commitment

For an AATS M with only finitely many states, actions and agents, there are clearly only finitely many strategies. Suppose that the set Υ_i is rich enough so that it includes a strategy term for *every* i's strategy from Σ_i. We might expect the following equivalence to hold in M:

$$\langle\langle i\rangle\rangle\varphi\leftrightarrow\bigvee_{\sigma_i\in\Upsilon_i}C_i(\sigma_i,\varphi)$$

The above property expresses that what i can achieve in a system M, is exactly all those results that will hold after i has committed to one of his available strategies. Unfortunately, this is in general not true in a framework with *irrevocable* commitments, which we can already demonstrate in the one agent case. Consider the AATS M from Figure 2. Let us abbreviate $\mathsf{free}\equiv(\langle\langle i\rangle\rangle\bigcirc\mathsf{dead}\wedge\langle\langle i\rangle\rangle\bigcirc\neg\mathsf{dead})$. Note that $M,q_1\models\mathsf{free}$, i.e. the agent has a free choice to die or stay alive in state q_1. On the other hand, as soon as he irrevocably commits to either of the two available strategies (σ_1 with $\sigma_1(q_1)=nop$, and σ_2 with $\sigma_2(q_1)=die$), he does not have the free choice any more:

$$M,q_0\models\langle\langle i\rangle\rangle\bigcirc\mathsf{free}\wedge\neg C_i(\sigma_1,\mathsf{free})\wedge\neg C_i(\sigma_2,\mathsf{free}).$$

This makes our approach similar to the perspective offered by *strategic reasoning* in game theory: we can see agents' commitments as the result of pre-play reasoning; once the players choose their best strategies, the structure of the game does not matter any more and the play is already settled before it begins. We will come back to the relationship between CATL and game theory in Section 4.

We have pointed out that the commitment semantics we adopt has essentially the same properties as the constraint implementation operator from [14]. We now briefly mention two such properties of the commitment modality in CATL. First, let us point out that commitment preserves universal properties of transition systems. Moreover, existential properties that hold under commitments, apply to the whole system as well. To this end, we define a *universal* and an *existential* sublanguage of ATL, denoted $\mathcal{L}^u$ and $\mathcal{L}^e$, respectively. These languages are defined by the following grammars:

$$\begin{array}{ll} \upsilon ::= & p \mid \neg p \mid \upsilon \wedge \upsilon \mid \upsilon \vee \upsilon \mid \langle\langle\rangle\rangle \bigcirc \upsilon \mid \langle\langle\rangle\rangle \Box \upsilon \mid \langle\langle\rangle\rangle \upsilon \mathcal{U} \upsilon \\ \epsilon ::= & p \mid \neg p \mid \epsilon \wedge \epsilon \mid \epsilon \vee \epsilon \mid \langle\langle Ag\rangle\rangle \bigcirc \epsilon \mid \langle\langle Ag\rangle\rangle \Box \epsilon \mid \langle\langle Ag\rangle\rangle \epsilon \mathcal{U} \epsilon \end{array}$$

where $p \in \Phi$, $\upsilon \in \mathcal{L}^u$ and $\epsilon \in \mathcal{L}^e$.

PROPOSITION 2. *Let σ be a strategic term for i in M, let q be a state in M, and let $\upsilon \in \mathcal{L}^u$ and $\epsilon \in \mathcal{L}^e$. Then:*

1. $M, q \models \upsilon \rightarrow C_i(\sigma, \upsilon)$.
2. $M, q \models C_i(\sigma, \epsilon) \rightarrow \epsilon$.

The proofs are analogous to those of [14].

Note that when every agent has committed to a strategy the future of the system is determined, and we have the following:

PROPOSITION 3. $\models C_{Ag}(\sigma_{Ag}, \langle\langle\rangle\rangle \bigcirc \varphi \leftrightarrow \langle\langle Ag\rangle\rangle \bigcirc \varphi)$

3.3 PDL-like Reasoning about Strategies

One of the advantages of CATL is that the logic enables explicit reasoning about actions and strategies in the style of Propositional Dynamic Logic [12]. Reasoning about (single-step) actions can be naturally extended to reasoning about strategies (that are being played *ad infinitum*), yielding a kind of "extended dynamic logic":

$$\begin{array}{lcl} [\sigma_G]\bigcirc\varphi & \hat{=} & C_G(\sigma_G, \langle\langle\rangle\rangle \bigcirc \varphi) \\ [\sigma_G]\Box\varphi & \hat{=} & C_G(\sigma_G, \langle\langle\rangle\rangle \Box \varphi) \\ [\sigma_G]\varphi \mathcal{U} \psi & \hat{=} & C_G(\sigma_G, \langle\langle\rangle\rangle \varphi \mathcal{U} \psi) \end{array}$$

EXAMPLE 3. *Coming back to our rocket example: every execution of action fuel by agent 3 in state 1 makes the tank of the rocket full: $M_1, 1 \models [fuel]\mathsf{fuelOK}$. Moreover, if 3 chooses the "lazy" strategy and the initial state is 1, then the rocket will inevitably stay in London for ever: $M_1, 1 \models [nop_3]\Box \mathsf{roL}$.*

In [15], we propose a richer language of strategy terms, and discuss the relationship to PDL in more detail.

3.4 Model Checking and Satisfiability

The *model checking problem* for CATL is the problem of determining, for any given CATL formula φ, AATS M, and state q in M, whether or not $M, q \models \varphi$. If M is an AATS and φ is a formula then we say that φ is *initially satisfied* in M if $M, q_0 \models \varphi$; we indicate this by writing $M \models \varphi$. Given M, i, and σ, computing $M \dagger_i \sigma$ can be done in time $O(m)$ (where m is the number of transitions in M). All we need do is go through the model deleting transitions where i performs an action other than that which is dictated by σ (i.e., "trim" the model). Finally, to model-check a CATL formula φ, it is sufficient to use the ATL model checking algorithm from [2], and call the "trimming" procedure every time a subformula of shape $C_i(\sigma, \varphi$ occurs. As the ATL model checking algorithm enjoys complexity of $O(ml)$, we obtain the following result.

PROPOSITION 4. *Model checking a CATL formula φ in model M can be done in time $O(ml)$, where m is the number of transitions in M, and l is the length of φ.*

A formula φ is *satisfiable* if there is some AATS M and state q in M such that $M, q \models \varphi$. The *satisfiability problem* for ATL is the problem of determining, for any given ATL formula, whether this formula is satisfiable or not.

4. STRATEGIC REASONING IN CATL

AATSs encapsulate intuitions related to extensive form games as well as strategic form games: on one hand, every agent acts through multiple subsequent moves; on the other, many agents play simultaneously at each state, and the outcome of a move depends on the actions of the other players. Thus, on one hand, ATL formulas of type $\langle\langle G\rangle\rangle \bigcirc \varphi$ can be seen as a formalization of reasoning about a single move in a (possibly more complex) game, and operators $\langle\langle G\rangle\rangle \Box$ and $\langle\langle G\rangle\rangle \Diamond$ as referring to an analysis of the entire game. In this sense, ATL formalizes reasoning about different aspects of *extensive game forms*, representing sequences of moves, collectively effected by the players' actions. Alternatively, formulas $\langle\langle G\rangle\rangle \bigcirc \varphi$ can be understood as expressing agents' powers to force outcomes in a single game, and operators $\langle\langle G\rangle\rangle \Box$ and $\langle\langle G\rangle\rangle \Diamond$ as referring to a collection of games played repeatedly *ad infinitum*. Thus, ATL can be also interpreted in terms of *strategic game forms*, in a way similar to the perspective of Coalition Logic [19]. In our introduction, we claimed that CATL is appropriate for reasoning about the outcome of strategies in game-like encounters. We now justify this claim by showing how CATL can be used to express important properties of games. We will focus on games in strategic form, but the concepts can be extended to perfect information games in extensive form in a natural way (cf. [15]).

4.1 Games and Correspondence

We will compare strategic games with AATS models. To keep them clearly apart, all entities in the game will have a *hat*. Note that in a strategic game we do not have to distinguish *actions* from *strategies*. We model a k-player strategic game $\hat{\Gamma}$ as a tuple $\hat{\Gamma} = \langle \hat{Ag}, \{\hat{Ac_i}\}, \{\hat{u}_i\}\rangle$, where: $\hat{Ag} = \{\hat{1}, \ldots, \hat{k}\}$ is the set of players, $\hat{Ac_i}$ is the set of actions (or strategies) for player i, $\hat{J}_{\hat{Ag}} = \hat{Ac}_1 \times \cdots \times \hat{Ag}_k$ is the set of all possible combinations of strategies, and $\hat{u}_i : \hat{J}_{\hat{Ag}} \rightarrow \mathbb{R}$ is player i's utility function, which assigns a real-valued utility to each combination of players' strategies. Notice that games in this sense, and our AATSs are very similar: the main differences are that (i) games are not state dependent, in that the outcome of the game depends only on the choices of actions made by agents, and not on the current state of the system; and, more significantly (ii) agents have preferences over outcomes, determined by their utility functions.

We make the relationship between games and ATSs precise with the notion of *correspondence*. Informally, this relationship should be clear from the notation, but to define the correspondence formally, we need to introduce into ATSs a mechanism for capturing utilities. One approach to this problem would be to extend the framework of coalition logic with desire or preference modalities of some kind. Logics of desire have been widely studied (see e.g. [20, 24]), and modal operators for desires were successfully used in [13] for a modal characterisation of Nash equilibrium; an attempt to give a game-theoretic foundation to a logic of desire was also presented in [23]. The disadvantage of such an approach is that it complicates the underlying logical framework. An alternative, which we adopt in this paper, is to label states with propositions that capture agent's utilities in these states. This approach is perhaps less elegant than the modal alternative [5, pp.308–309], but it is nevertheless simpler from a logical perspective, as we need not complicate the logic with additional modalities or other connectives.

We follow the approach of Baltag [3]. Let U denote the set of all utilities that may be assigned by $\hat{u}_i$ functions in $\hat{\Gamma}$. For each utility value $v \in U$ and agent $\hat{\imath} \in \hat{Ag}$, we introduce a proposition $(u_i \geq v)$ into our set Φ of primitive propositions of the corresponding AATS model M, and fix the valuation function π so that $(u_i \geq v)$ is satisfied in state s iff i gets at least v in s. Additionally, we define $u_i > v$ as a shorthand for $\bigvee_{v' > v} u_i \geq v'$.

Consider a strategic game $\hat{\Gamma} = \langle \hat{Ag}, \{\hat{Ac}_i\}, \{\hat{u}_i\} \rangle$, and a model $M = \langle Q, q_0, \Phi, \pi, Ag, Ac_1, \ldots, Ac_n, \rho, \tau, \Upsilon_1, \ldots, \Upsilon_n, \| \cdot \| \rangle$ with state $q \in Q$. We write $\hat{\Gamma} \simeq (M, q)$ to denote the fact that $\hat{\Gamma}$ *corresponds to* M, q, in the sense that:

1. the sets of agents are the same: $\hat{Ag} = Ag$;
2. strategies in $\hat{\Gamma}$ correspond to actions that can be executed in M, q: $\hat{Ac}_i = options(i, q)$;
3. Υ has a term α for every strategy $\hat{\alpha}$ in $\hat{\Gamma}$. More precisely, we require that for every $\hat{\alpha} \in \hat{Ac}_i$ there is an $\alpha \in \Upsilon_i$ such that $\|\alpha\|\ (q) = \hat{\alpha}$, i.e. we can use strategy terms to address every single-step action in q;
4. $\Phi \supseteq \{(u_j \geq v) \mid \hat{u}_j \in \{\hat{u}_i\}, v \in U\}$
5. for all $j \in J_{Ag}$ with $q' = \tau(q, j)$, we have: $(u_i \geq v) \in \pi(q')$ iff $\hat{u}_i(\hat{\jmath}) \geq v$.

Thus, states in M are mainly used to represent various possible outcomes of the strategic game $\hat{\Gamma}$.

LEMMA 1. *Let $\hat{\Gamma}$ be a game, M be an* AATS *with a state q such that $\hat{\Gamma} \simeq (M, q)$; let $\hat{\jmath}, \hat{\jmath}' \in \hat{J}_{\hat{Ag}}$ be strategy profiles in $\hat{\Gamma}$, and i be an agent in $Ag = \hat{Ag}$. Then:*

1. *$\hat{u}_i(\hat{\jmath}) > \hat{u}_i(\hat{\jmath}')$ iff for some $v \in U$ we have $M, \tau(q, j) \models (u_i \geq v)$ and $M, \tau(q, j') \models \neg(u_i \geq v)$;*
2. *$\hat{u}_i(\hat{\jmath}) \geq \hat{u}_i(\hat{\jmath}')$ iff for all $v \in U$, if $M, \tau(q, j') \models (u_i \geq v)$ then also $M, \tau(q, j) \models (u_i \geq v)$.*

To keep the definitions and results as readable as possible, we will from now assume that games have just two agents, 1 and 2. Moreover, we will write $\hat{\Gamma} \simeq (M, q)$ without explicitly saying that $\hat{\Gamma}$ is a game, and M a system with state q. Since the agents in the game $\hat{\Gamma}$ and the model M are the same, we omit the hat-notation for them. Finally, in the two-agent case, if i is an agent, k refers to the *other* agent.

To look at a simple example, consider a version of the Prisoner's Dilemma (PD) presented in Figure 3A. The outcome pair $(-5, 0)$, for instance, represents that when player 1 cooperates (C) while player 2 defects (d), the sentence for player 1 is 5 years in prison, while 2 can go without any punishment. The corresponding AATS model might have a root q_0 and four states: $q_1, \ldots, q_4$, where q_1 would be obtained from q under the profile $\langle C, c \rangle$ and satisfy propositions $u_1 \geq -5$, $u_1 \geq -4$, $u_1 \geq -2$, $u_2 \geq -5$, $u_2 \geq -4$ and, finally $u_2 \geq -2$; the state q_2 would be obtained from q under the profile $\langle C, d \rangle$ etc. The other game in Figure 3B represents the "Bach or Stravinsky" game, also known as the "Battle of the Sexes" (BoS). A corresponding AATS is presented in Figure 3C.

(A)

$1 \setminus 2$	c	d
C	$-2, -2$	$-5, 0$
D	$0, -5$	$-4, -4$

(B)

$1 \setminus 2$	b	s
B	2,1	0,0
S	0,0	1,2

(C)

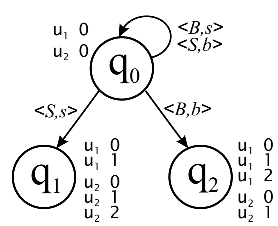

Figure 3: Strategic games: (A) "Prisoner's Dilemma", (B) "Bach or Stravinsky"; (C) example transition systems corresponding to BoS

4.2 Dominant and Dominated Strategies

The first concept we capture in our logic is that of a strategy being *dominant*. Intuitively, a strategy is dominant if it is the best response to *all* strategies that may be played by one's opponents. The presence of a dominant strategy makes an agent's reasoning process simpler: a rational agent will always play a dominant strategy. Notice that "defect" (playing D and d) are dominant strategies for players 1 and 2 in PD. We focus on *weakly dominant* strategies, which are "at least as good as" the alternatives. Formally, $\hat{\alpha}$ is weakly dominant for i in $\hat{\Gamma}$ iff for all $\hat{\alpha}' \neq \hat{\alpha} \in \hat{Ac}_i$, and for all $\hat{\beta} \in \hat{Ac}_k$, we have $\hat{u}_i(\hat{\alpha}, \hat{\beta}) \geq \hat{u}_i(\hat{\alpha}', \hat{\beta})$.

To capture weak dominance in the corresponding model, we define a unary predicate $WD_i(\alpha)$, the idea being that $WD_i(\alpha)$ is satisfied in a state iff α is weakly dominant in that state. We start with a predicate $wd_i(\alpha)$, which appears promising, but is in fact not sufficiently strong:

$$wd_i(\alpha) \hat{=} \bigwedge_{v \in U} (\langle\langle i \rangle\rangle \bigcirc (u_i \geq v) \rightarrow C_i(\alpha, \langle\langle\rangle\rangle \bigcirc (u_i \geq v)))$$

The predicate $wd_i(\alpha)$ expresses that if i can guarantee a value v, he can already guarantee it using his strategy α. Unfortunately, this is too weak to make α weakly dominant, which we will demonstrate using the BoS example. The best that player 1 can guarantee is a value 0. He can also obtain this using his strategy $\hat{\alpha} = B$, still in this case B is not a dominant strategy (BoS lacks dominant strategies). The problem lies in the fact that the outcomes of α with respect to every opponent's response β should be considered *separately*. This suggests that we need to be able to quantify over strategies. Indeed, if the number of strategies (actions) for i is finite, we can express weak dominance by the following:

PROPOSITION 5. *For finite $\hat{\Gamma}$ and M, if $\hat{\Gamma} \simeq (M, q)$ then $\hat{\alpha}$ is weakly dominant for i in $\hat{\Gamma}$ iff $M, q \models WD_i(\alpha)$, where:*

$$WD_i(\alpha) \hat{=} \bigwedge_{\beta \in \Upsilon_j} C_j(\beta, wd_i(\alpha)).$$

$WD_i(\alpha)$ above expresses that α is the best response to every particular opponent's strategy β. Notice that characterising weak dominance requires the ability both to quantify over the possible choices of agents in a system (which is possible in "pure" ATL), and also to address properties of *named* strategies (which is not possible in "pure" ATL, but can be done in CATL).

The notion of a *dominated strategy* is related to that of a dominant strategy, although when having more than two strategies for

$1\backslash 3$	nop_3	$load_3$	$fuel$
nop_1	$0,1$	$7,0$	$1,2$
$load_1$	$7,0$	$7,0$	$0,1$
$unload_1$	$0,1$	$0,1$	$1,2$

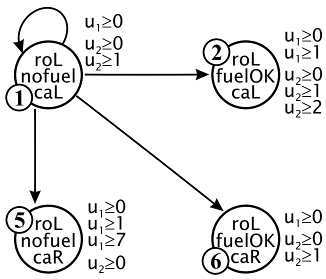

Figure 4: A rocket game

one player, the notions are independent. It is well-known that iteratively eliminating their dominated strategies may lead the game to an equilibrium state.

We want $bt_i(\alpha_1, \alpha_2)$ to mean that α_1 guarantees (strictly) better outcome than α_1 for i, $DO_i(\alpha_1, \alpha_2)$ to mean that strategy α_1 is (strongly) dominated by strategy α_2 for agent i, and $D_i(\alpha)$ to denote that α is a (strongly) dominated strategy. We define these predicates as follows:

$$
\begin{aligned}
bt_i(\alpha_1, \alpha_2) &\hat{=} \bigvee_{v \in U} (C_i(\alpha_1, \langle\langle\rangle\rangle \bigcirc (u_i \geq v)) \wedge \\
&\qquad \neg C_i(\alpha_2, \langle\langle\rangle\rangle \bigcirc (u_i \geq v))) \\
DO_i(\alpha_1, \alpha_2) &\hat{=} \bigwedge_{\beta \in \Upsilon_j} C_j(\beta, \neg bt_i(\alpha_1, \alpha_2)) \wedge \bigvee_{\beta \in \Upsilon_j} C_j(\beta, bt_i(\alpha_2, \alpha_1)) \\
D_i(\alpha) &\hat{=} \bigvee_{\alpha' \in \Upsilon_i} DO_i(\alpha, \alpha')
\end{aligned}
$$

PROPOSITION 6. *Suppose that $\hat{\Gamma} \simeq (M, q)$, and let $\hat{\alpha}$ be a strategy for agent i in $\hat{\Gamma}$. Then:*

1. *$\hat{\alpha_1}$ is dominated by $\hat{\alpha_2}$ in $\hat{\Gamma}$ iff $M, q \models DO_i(\alpha_1, \alpha_2)$*
2. *$\hat{\alpha}$ is a dominated strategy in $\hat{\Gamma}$ iff $M, q \models D_i(\alpha)$.*

EXAMPLE 4. *Let us consider again the "rocket agents" from Section 2.1. This time, we would like to add some information about the agents' preferences (utilities). Suppose that the cargo contains some materials that can incriminate agent 1 before the French police. Thus, 1 does not want the cargo to stand freely at the London airport, but he is much more afraid that the cargo may arrive in Paris. The best option for him is when the cargo is inside the rocket and the rocket cannot fly (i.e., it has its tank empty); also, he has a slight preference for the situation when the rocket is supplied with fuel and the cargo is outside (then the rocket can be moved to Paris in the next step, which guarantees that the cargo will remain away from Paris for some time. Agent 3, on the other hand, gets more bonus when the rocket tank is full; also, he is responsible for cargo which is on board of the rocket, so he prefers when no cargo is loaded. We assume that agent 2 chooses to do nothing throughout the game, and that the agents, cargo and rocket are initially in London.* AATS *$M_2 = M_1 \dagger_2 nop_2$ can be augmented with utility-defining propositions to correspond to the game; Figure 4 shows the table of utilities for the game as well as the relevant part of the resulting transition system.*

Let $\overline{\alpha}$ denote an arbitrary strategy in M_2 for which $\overline{\alpha}(q) = \alpha$. Note that $M_2, 1 \models WD_3(\overline{fuel})$ and indeed fuel is a dominant action for agent 3. Note also that 1 has no dominant action, and $\neg WD_1(\overline{nop}_1) \wedge \neg WD_1(\overline{load}_1) \wedge \neg WD_1(\overline{unload}_1)$ appropriately holds in $M_2, 1$.

Furthermore, $M_2, 1 \models D_i(\overline{unload}_1) \wedge \neg D_i(\overline{nop}_1) \wedge \neg D_i(\overline{load}_1)$, and indeed $unload_1$ is the only dominated action of player 1.

Unfortunately, as witnessed by BOS, dominant or dominated strategies seldom exist, and hence alternative solution concepts have been developed. Of these, Nash equilibrium is the best known and most important.

4.3 Nash Equilibrium

We say that a pair of strategies $\langle \hat{\alpha_1}, \hat{\alpha_2} \rangle$ for the grand coalition $\{1, 2\}$ is in Nash equilibrium in game $\hat{\Gamma}$ if these strategies are each the best response to each other, i.e., if $\hat{u_1}(\langle \hat{\alpha_1}, \hat{\alpha_2} \rangle) \geq \hat{u_1}(\langle \hat{\alpha'_1}, \hat{\alpha_2} \rangle)$ for all $\hat{\alpha'_1} \in \hat{Ac_1}$, and, similarly, $\hat{u_2}(\langle \hat{\alpha_1}, \hat{\alpha_2} \rangle) \geq u_2(\langle \hat{\alpha_1}, \hat{\alpha'_2} \rangle)$ for all $\hat{\alpha'_2} \in \hat{Ac_2}$. The profile $\langle D, d \rangle$ is the only Nash equilibrium in PD, while BOS has two Nash equilibria, namely $\langle B, b \rangle$ and $\langle s, S \rangle$.

To express Nash equilibrium, we first characterise the notion of one strategy being a best response to another. We write $BR_i(\alpha_k, \alpha_i)$ to denote the fact that i's best response to k playing α_k is α_i, and define it as:

$$
\begin{aligned}
&BR_i(\alpha_G, \alpha_i) \hat{=} \\
&\qquad C_k(\alpha_k, \bigwedge_{v \in U} (((\langle\langle i \rangle\rangle \bigcirc (u_i \geq v)) \rightarrow C_i(\alpha_i, \langle\langle\rangle\rangle \bigcirc (u_i \geq v)))
\end{aligned}
$$

This says that if k plays α_k, then every utility v that can be achieved by i can already be achieved by i playing α_i.

PROPOSITION 7. *Suppose $\hat{\Gamma} \simeq (M, q)$; let $\hat{\alpha}_k$ be a strategy for k in $\hat{\Gamma}$, and $\hat{\alpha}_i$ a strategy for agent i. Then $M, q \models BR_i(\alpha_k, \alpha_i)$ iff the best outcome i can obtain assuming k plays $\hat{\alpha}_k$ is obtained by playing $\hat{\alpha}_i$.*

When the number of strategies if finite, we could have also characterised $BR_i(\alpha_k, \alpha_i)$ as follows, saying that α_i is i best response to α_k, if every utility achieved when i plays an arbitrary β against α_k, can also be achieved by playing α_i against α_k.

PROPOSITION 8. *For finite $\hat{\Gamma}$ and M, suppose $\hat{\Gamma} \simeq (M, q)$ and let α_k be a strategy for k in $\hat{\Gamma}$, and $\hat{\alpha}_i$ a strategy for agent i. Then the following two statements are equivalent:*

1. *$M, q \models BR_i(\alpha_k, \alpha_i)$*
2. *$M, q \models \bigwedge_{\beta_i \in \Upsilon_i} \bigwedge_{v \in U} (C_i(\beta, C_k(\alpha_k, (u_i \geq v)) \rightarrow C_i(\alpha_i, C_k(\alpha_k, (u_i \geq v))))$.*

We can now define a proposition $NE(\alpha_G)$ to denote the fact that strategy profile α_G is in Nash equilibrium.

$$NE(\alpha_1, \alpha_2) \hat{=} BR_1(\alpha_2, \alpha_1) \wedge BR_2(\alpha_1, \alpha_2)$$

The following is now immediate.

PROPOSITION 9. *Suppose $\hat{\Gamma} \simeq (M, q)$ and let α_i be a strategy for i in $\hat{\Gamma}$. Then $M, q \models NE(\alpha_1, \alpha_2)$ iff $\langle \hat{\alpha_1}, \hat{\alpha_2} \rangle$ is a Nash equilibrium in $\hat{\Gamma}$.*

EXAMPLE 5. *Consider the game and the corresponding* AATS *from Example 4 and Figure 4. Strategy pair $\langle \widehat{unload_1}, \widehat{fuel} \rangle$ is a Nash equilibrium here, and accordingly $M_2, 1 \models NE(\overline{unload}_1, \overline{fuel})$. Also, $M_2, 1 \models \neg NE(\overline{load}_1, \overline{load}_3)$ and indeed $\langle \widehat{load_1}, \widehat{load_3} \rangle$ is not a Nash equilibrium point because agent 3 can get a better payoff against $\widehat{load_1}$ by playing $\widehat{fuel}$.*

4.4 Pareto Optimality

The next concept we capture is that of *Pareto optimality* [18]. A strategy profile $\langle \hat{\alpha}, \hat{\beta} \rangle$ is Pareto optimal if there is no other strategy profile $\langle \hat{\alpha}', \hat{\beta}' \rangle$ for $\{1, 2\}$ that will lead to an increase in utility for some members of $\{1, 2\}$ without any of them suffering a decrease in utility. The game of BoS has two Pareto optimal profiles: $\langle B, b \rangle$ and $\langle S, s \rangle$, while all combinations of strategies in PD except for $\langle D, d \rangle$ are Pareto optimal.

Formally, $PO(\alpha, \beta)$ is defined as below. We use $\langle u_1, u_2 \rangle \geq \langle v_1, v_2 \rangle$ as a shorthand for $((u_1 \geq v_1) \wedge (u_2 \geq v_2))$.

$$
\begin{array}{l}
PO(\alpha, \beta) \hat{=} \\
\bigwedge_{v_1} \bigwedge_{v_2} \\
\langle\langle 1, 2 \rangle\rangle \bigcirc (\langle u_1, u_2 \rangle \geq \langle v_1, v_2 \rangle) \rightarrow \\
\quad C_{\{1,2\}}(\langle \alpha, \beta \rangle, \langle\langle\rangle\rangle \bigcirc (\langle u_1, u_2 \rangle \geq \langle v_1, v_2 \rangle \vee ((u_1 > v_1) \vee (u_2 > v_2))))
\end{array}
$$

The displayed formula expresses that if a collective utility *cannot* be achieved by coalition $\{1, 2\}$ while playing strategies α and β respectively (i.e. at least one of the players is bound to get a worse outcome), then it cannot be achieved by coalition $\{1, 2\}$ in general.

PROPOSITION 10. *Suppose that $\hat{\Gamma} \simeq (M, q)$ and let $\langle \hat{\alpha}, \hat{\beta} \rangle$ be a strategy profile for the grand coalition $\{1, 2\}$ in $\hat{\Gamma}$. Then $M, q \models PO(\langle \alpha, \beta \rangle)$ iff $\langle \hat{\alpha}, \hat{\beta} \rangle$ is a Pareto optimal strategy profile for $\{1, 2\}$ in $\hat{\Gamma}$.*

EXAMPLE 6. *The following strategy profiles in the game from Example 4 are Pareto optimal: $\langle \hat{nop}_1, \hat{load}_3 \rangle$, $\langle \hat{nop}_1, \hat{fuel} \rangle$, $\langle \hat{load}_1, \hat{nop}_3 \rangle$, $\langle \hat{load}_1, \hat{load}_3 \rangle$ and $\langle \hat{unload}_1, \hat{fuel} \rangle$. Accordingly, $M_2, 1 \models PO(\overline{nop_1}, \overline{load_3}) \wedge PO(\overline{nop}_1, \overline{fuel})$ etc. In the same way, $M_2, 1 \models \neg PO(\overline{nop}_1, \overline{nop}_3) \wedge \neg PO(\overline{load}_1, \overline{fuel})$ and so on.*

5. CONCLUSIONS

In recent years, there has been much interest in the use of logic for representing and reasoning about game-like interactions. Examples include the development of logics intended for reasoning about coalitional power in games [1, 2, 19], the use of dynamic epistemic logics to capture properties of games [4, 3], Bonanno's work on the relationship of branching time logic to extensive form games [7], and of course the use of epistemic logic for capturing such game theoretic concepts as perfect recall [10].

Our logic CATL adds strategy terms to the vocabulary of modal logic, enabling one to reason about "what-if" scenarios, which correspond to agents choosing a particular strategy. We showed how this gives a natural framework within which several well-known solution concepts from game theory can be expressed. Directions for future research are manyfold. First of all, reasoning about choices of agents can be done in a more fine-tuned way if we allow for more structure in strategic terms. This would make the link with Propositional Dynamic Logic PDL more explicit. Second, yet we only demonstrated the possible *use* of CATL; properties in terms of axiomatization and computational complexity are not resolved. Finally, it may be worthwhile to use CATL in the area of extensive games, with utilities assigned to arbitrary states and various notions of outcome.

6. REFERENCES

[1] R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-time Temporal Logic. In *Proceedings of FOCS*, pages 100–109, 1997.

[2] R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-time temporal logic. *Journal of the ACM*, 49(5):672–713, September 2002.

[3] A. Baltag. A logic for suspicious players. *Bulletin of Economic Research*, 54(1):1–46, 2002.

[4] J. van Benthem. Games in dynamic epistemic logic. *Bulletin of Economic Research*, 53(4):219–248, 2001. Proceedings of LOFT-4.

[5] J. van Benthem. Extensive games as process models. *Journal of Logic, Language, and Information*, 11(3):289–313, 2002.

[6] A. L. Blum and M. L. Furst. Fast planning through graph analysis. *Artificial Intelligence*, 90:281–300, 1997.

[7] G. Bonanno. Branching time logic, perfect information games and backward induction. *Games and Economic Behavior*, 36(1):57–73, July 2001.

[8] G. van Drimmelen. Satisfiability in alternating-time temporal logic. In *Eighteenth Annual IEEE Symposium on Logic in Computer Science (LICS 2003)*, pages 208–217, Ottawa, Canada, 2003.

[9] E. A. Emerson. Temporal and modal logic. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science Volume B: Formal Models and Semantics*, pages 996–1072. Elsevier Science Publishers B.V.: Amsterdam, The Netherlands, 1990.

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

[11] V. Goranko and W. Jamroga. Comparing semantics of logics for multi-agent systems. *Synthese*, 139(2):241–280, 2004.

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

[13] P. Harrenstein, W. van der Hoek, J.-J Meyer, and C. Witteveen. On modal logic interpretations for games. In *Proceedings of the Fifteenth European Conference on Artificial Intelligence (ECAI-2002)*, pages 28–32, Lyon, France, 2002.

[14] W. van der Hoek, M. Roberts, and M. Wooldridge. Social laws in alternating time: Effectiveness, feasibility and synthesis. *Synthese*, 2005.

[15] W. Jamroga, W. van der Hoek, and M. Wooldridge. Intentions and strategies in game-like scenarios. Technical Report IfI-05-08, Clausthal University of Technology, 2005.

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

[17] D. Lewis. *Counterfactuals*. Blackwells: Oxford, 1973.

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

[19] M. Pauly. A modal logic for coalitional power in games. *Journal of Logic and Computation*, 12(1):149–166, 2002.

[20] A.S. Rao and M.P. Georgeff. Modeling rational agents within a BDI-architecture. In *Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning*, pages 473–484, 1991.

[21] P. Y. Schobbens. Alternating-time logic with imperfect recall. *Electronic Notes in Theoretical Computer Science*, 85(2), 2004.

[22] R. Stalnaker. Knowledge, belief and counterfactual reasoning in games. *Economics and Philosophy*, 12:133–163, 1996.

[23] S. van Otterloo, W. van der Hoek, and M. Wooldridge. Preferences in game logics. In *Proceedings of AAMAS-04*, 2004.

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