Intentions and Strategies in Game-Like Scenarios

Wojciech Jamroga[1], Wiebe van der Hoek[2], and Michael Wooldridge[2]

[1] Institute of Computer Science, Clausthal University of Technology, Germany
wjamroga@in.tu-clausthal.de
[2] Department of Computer Science, University of Liverpool, UK
{wiebe, mjw}@csc.liv.ac.uk

Abstract. In this paper, we investigate the link between logics of games and "mentalistic" logics of rational agency, in which agents are characterized in terms of attitudes such as belief, desire and intention. In particular, we investigate the possibility of extending the logics of games with the notion of agents' intentions (in the sense of Cohen and Levesque's BDI theory). We propose a new operator $(\mathbf{str}_a\sigma)$ that can be used to formalize reasoning about outcomes of strategies in game-like scenarios. We briefly discuss the relationship between intentions and goals in this new framework, and show how to capture dynamic logic-like constructs. Finally, we demonstrate how game-theoretical concepts like Nash equilibrium can be expressed to reason about rational intentions and their consequences.

Keywords: Multi-agent systems, strategic reasoning, common sense reasoning.

1 Introduction

In this paper, we investigate the link between logics of games (in particular, ATL – the temporal logic of coalitional strategic ability) and "mentalistic" logics of rational agency, in which agents are characterized in terms of attitudes such as belief, desire and intention. It is our contention that successful knowledge representation formalisms for multi-agent systems would ideally embrace both traditions. Specifically, we propose to extend ATL with agents' intentions (in the sense of Cohen and Levesque's BDI theory) in order to reason about agents' intended actions and their consequences.

This is especially interesting in game-like situations, where agents can consider hypothetical strategies of other agents, and come up with a better analysis of the game. We define a counterfactual operator $(\mathbf{str}_a\sigma)$ to reason about outcomes of strategy σ; in consequence, one can reason explicitly about *how* agents can achieve their goals, besides reasoning about *when* does it happen and *who* can do it, inherited from temporal logic and logic of strategic ability. We discuss the notion of intending *to do* an action, as opposed to of intending *to be* in a state that satisfies a particular property; we analyze the relationship between action- and state-oriented intentions, and point out that our framework allows for a natural interpretation of *collective* intentions and goals. We show how a dynamic-like logic of strategies can be defined on top of the resulting language, and argue that propositional dynamic logic can be embedded in it in a natural way. We present a model checking algorithm that runs in time linear in the size of the model and length of the formula. Finally, we suggest that this operator sits very well in game-like reasoning about rational agents, and show examples of such reasoning. Most concepts that we present here have been discussed only briefly due to space limitations.

C. Bento, A. Cardoso, and G. Dias (Eds.): EPIA 2005, LNAI 3808, pp. 512–523, 2005.

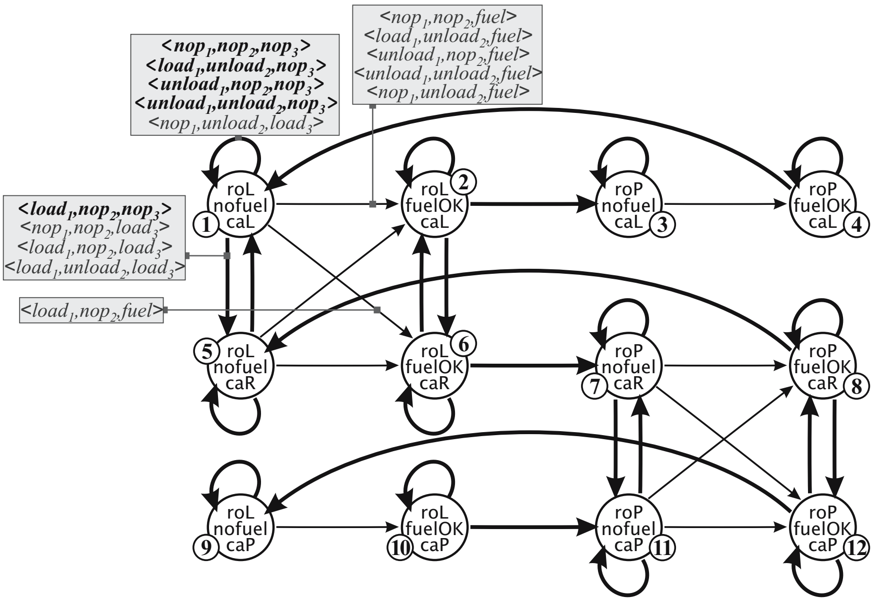

Fig. 1. Simple Rocket Domain. The "bold" transitions are the ones in which agent 3 intends to always choose nop_3.

2 What Agents Can Achieve?

Alternating-time Temporal Logic (ATL) [1] is a generalization of the branching time temporal logic CTL [8], in which path quantifiers are replaced by *cooperation modalities*. Formula $\langle\langle A\rangle\rangle\varphi$, where A is a coalition of agents (i.e., a subset of the "grand" set of agents $\mathbb{A}\mathrm{gt}$), expresses that there exists a collective plan for A such that, by following this plan, A can enforce φ. ATL formulae include temporal operators: "$\bigcirc$" ("in the next state"), $\Box$ ("always") and $\mathcal{U}$ ("until").[1] 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 the extension of ATL, proposed in this paper, makes use of terms that describe strategies, and in this sense is very different from ATL, in which strategies appear only in the semantics and are *not* referred to in the object language. We will introduce the semantic concepts behind ATL formally in Section 3. For now, we give a flavor of it with the following example.

Example 1. Consider a modified version of the Simple Rocket Domain from [3]. There is a rocket that can be moved between London (roL) and Paris (roP), and piece of cargo that can lie in London (caL), Paris (caP), or inside the rocket (caR). Three agents are involved: 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

[1] An additional operator $\Diamond$ ("now or sometime in the future") can be defined as , $\Diamond\varphi \equiv \top\,\mathcal{U}\varphi$.

fuel. Every agent can also stay idle at a particular moment (the nop – "no-operation" actions). The "moving" action has the highest priority. "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). 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. A model for the domain is shown in Figure 1 (we will refer to this model as M_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.

Example ATL formulae that hold in $M_1, 1$ are: $\neg\langle\langle 1\rangle\rangle\Diamond\mathsf{caP}$ (agent 1 cannot deliver the cargo to Paris on his own), $\langle\langle 1,3\rangle\rangle\Diamond\mathsf{caP}$ (1 and 3 can deliver the cargo if they cooperate), and $\langle\langle 2,3\rangle\rangle\Box(\mathsf{roL}\wedge\langle\langle 2,3\rangle\rangle\Diamond\mathsf{roP})$ (2 and 3 can keep the rocket in London forever, and still they retain the ability to change their strategy and move the rocket to Paris). ■

Players' strategies and players' preferences are key concepts in game theory. Preference Game Logic (PGL) [20] has been an attempt to import the concept of preferences into the framework of ATL via formulae $[A:p]\varphi$, meaning that "*if agents A prefer outcome p then φ holds*". We would like to follow the basic idea behind PGL in this paper; however, it models agents' behavior in a rather arbitrary way. Roughly, the idea behind $[A:p]\varphi$ is that, if A prefer outcome p, they will only perform certain strategies, and they all lead to φ. In this paper, we disconnect these two notions, one giving the agents recommended strategies (given their preferences), the other calculating the effects of certain strategies being chosen. Our primary focus is on reasoning about outcomes of strategies, regardless of where the strategies come from (and whether they are rational or not). We formalize this kind of reasoning in section 3. However, having a device for reasoning about outcomes of *all* strategies, and a criterion of rationality, we can combine the two to reason about outcomes of strategies that *rational* agents may or should follow. This issue is discussed in more detail in Section 4.

3 ATL with Intentions

The language of ATL+I (with respect to a set of agents $\mathbb{A}\mathrm{gt}$, atomic propositions Π, and sets of primitive strategic terms $\Upsilon_{a_1},...,\Upsilon_{a_k}$ for agents $a_1,...,a_k$ from $\mathbb{A}\mathrm{gt}$) can be formally defined as the following extension of ATL:

$$\varphi ::= p \mid \neg\varphi \mid \varphi\wedge\varphi \mid \langle\langle A\rangle\rangle\bigcirc\varphi \mid \langle\langle A\rangle\rangle\Box\varphi \mid \langle\langle A\rangle\rangle\varphi\,\mathcal{U}\varphi \mid (\mathbf{str}_a\sigma_a)\varphi$$

where $p \in \Pi$ is a proposition, $a \in \mathbb{A}\mathrm{gt}$ is an agent, $A \subseteq \mathbb{A}\mathrm{gt}$ is a group of agents, and $\sigma_a \in \Upsilon_a \cup \{\,\mathsf{any}\,\}$ is a strategic term for a. Models for ATL+I extend *concurrent game structures* from [1] with intention-accessibility relations, strategic terms and their denotation, and can be defined as:

$$M = \langle \mathbb{A}\mathrm{gt}, Q, \Pi, \pi, Act, d, o, \mathcal{I}_{a_1},...,\mathcal{I}_{a_k}, \Upsilon_{a_1},...,\Upsilon_{a_k}, [\![]\!]_{a_1},...,[\![]\!]_{a_k}\rangle.$$

$\mathbb{A}\mathrm{gt} = \{a_1,...,a_k\}$ is the set of all agents (the "grand coalition"), Q is the set of states of the system, Π the set of atomic propositions, $\pi : \Pi \to \mathcal{P}(Q)$ a valuation of propositions, and Act the set of (atomic) actions; function $d : \mathbb{A}\mathrm{gt}\times Q \to \mathcal{P}(Act)$ defines actions available to an agent in a state, and o is the (deterministic) transition function that assigns the outcome state $q' = o(q, \alpha_1, \dots, \alpha_k)$ to every state q and tuple of actions $\langle \alpha_1, \dots, \alpha_k \rangle$ that can be executed by the grand coalition in q.

$\mathcal{I}_a \subseteq Q \times Act$ is the intention-accessibility relation of agent a ($q\mathcal{I}_a\alpha$ meaning that a possibly intends to do action α when in q). A *strategy* of agent a is a conditional plan that specifies what a is going to do in every possible situation (state). We represent a's strategies as functions of type $s_a : Q \to \mathcal{P}(Act)$ such that, for every $q \in Q$: (1) $s_a(q)$ is non-empty, and (2) $s_a(q) \subseteq d_a(q)$. Thus, strategies can be non-deterministic – we only require that they specify choices of agents, and at least one choice per state. Strategic terms $\sigma \in \Upsilon_a$ are interpreted as strategies according to function $[\![]\!]_a : \Upsilon_a \to (Q \to \mathcal{P}(Act))$ such that $[\![\sigma]\!]_a$ is a valid strategy for a. We also define $[\![\,\mathsf{any}\,]\!]_a$ as the strategy that collects all valid actions of a, i.e. $[\![\,\mathsf{any}\,]\!]_a(q) = d_a(q)$ for every q. A *collective strategy* for a group of agents $A = \{a_1, ..., a_r\}$ is simply a tuple of strategies $S_A = \langle s_{a_1}, ..., s_{a_r} \rangle$, one per agent from A. A *path* $\Lambda = q_0q_1q_2...$ in M is an infinite sequence of states that can be effected by subsequent transitions, and refers to a possible course of action (or a possible computation) that may occur in the system. We define $\Lambda[i]$ to be the ith state in path Λ.

In ATL, agents can choose any legal action at each state. Having added intentions to ATL models, we assume that *agents only do what they intend*. We say that strategy s_a is *consistent with a's intentions* if the choices specified by s_a are never ones that a does *not* intend, i.e. $q\mathcal{I}_a\alpha$ for every q and $\alpha \in s_a(q)$. A collective strategy S_A is consistent with A's intentions if s_a are consistent with a's intentions for all $a \in A$. The set of outcome paths of a (collective) strategy S_A from state q, denoted by $out(q, S_A)$, is defined as the set of paths in M, starting from q, that can result from A executing S_A. Unlike in ATL, we are going to consider only courses of action that are consistent with intentions of all agents:

$out(q, S_A) = \{\Lambda = q_0q_1... \mid q_0 = q$ and for every $i = 1, 2, ...$ there exists a tuple of all agents' actions $\langle \alpha_{a_1}^{i-1}, ..., \alpha_{a_k}^{i-1} \rangle$ such that $\alpha_a^{i-1} \in s_a(q_{i-1})$ for $a \in A$, and $q_{i-1}\mathcal{I}_a\alpha_a^{i-1}$ for $a \in \mathbb{A}\mathrm{gt} \setminus A$, and $o(q_{i-1}, \alpha_{a_1}^{i-1}, ..., \alpha_{a_k}^{i-1}) = q_i\}$.

Semantics of ATL+I can be given via the following clauses:

$M, q \models p$ iff $q \in \pi(p)$, for an atomic proposition p;
$M, q \models \neg\varphi$ iff $M, q \not\models \varphi$;
$M, q \models \varphi \wedge \psi$ iff $M, q \models \varphi$ and $M, q \models \psi$;
$M, q \models \langle\langle A \rangle\rangle \bigcirc \varphi$ iff there is a collective strategy S_A consistent with A's intentions, such that for every $\Lambda \in out(q, S_A)$, we have that $M, \Lambda[1] \models \varphi$;
$M, q \models \langle\langle A \rangle\rangle \Box \varphi$ iff there is S_A consistent with A's intentions, such that for every $\Lambda \in out(q, S_A)$ and $i = 0, 1, 2, ...$, we have $M, \Lambda[i] \models \varphi$;
$M, q \models \langle\langle A \rangle\rangle \varphi \mathcal{U} \psi$ iff there is S_A consistent with A's intentions, such that for every $\Lambda \in out(q, S_A)$ there is $i \geq 0$ such that $M, \Lambda[i] \models \psi$ and for all j such that $0 \leq j < i$, we have $M, \Lambda[j] \models \varphi$;
$M, q \models (\mathbf{str}_a\sigma)\varphi$ iff $revise(M, a, [\![\sigma]\!]_a), q \models \varphi$.

The function $revise(M, a, s)$ updates model M by setting a's intention-accessibility relation $\mathcal{I}_a = \{\langle q, \alpha \rangle \mid \alpha \in s(q)\}$, so that s and $\mathcal{I}_a$ represent the same mapping in the resulting model. In a way, $revise$ implements agents' intention revision (or strategy change) in game structures with intentions.

Example 2. Let us go back to the rocket agents from Example 1. If we have no information about agents' intended actions and strategies, we can model the game with model M_1' which augments M_1 with the least restrictive intention-accessibility relations, so that $q\mathcal{I}_a\alpha$ for every $q \in Q$, $a \in \mathbb{A}\mathrm{gt}$ and $\alpha \in d_a(q)$. Let nop denote the "lazy" strategy for agent 3, i.e. $[\![nop]\!]_3(q) = nop_3$ for every q. Model $M_2 = revise(M_1', 3, [\![nop]\!]_3)$ depicts the situation where 3 intends to play nop and the other players have no specific intentions. Transitions, consistent with the intention-accessibility relations, are indicated with bold face font and thick arrows in Figure 1. Note that, for example, $M_2, 1 \models \langle\!\langle 2 \rangle\!\rangle \Box \neg \mathsf{caR}$ (agent 2 can keep the cargo outside the rocket), and $M_2, 1 \models \langle\!\langle \rangle\!\rangle \Box \mathsf{nofuel}$ (the rocket tank is always empty for all courses of action).[2] Thus, also $M_1', 1 \models (\mathbf{str}_3 nop)\langle\!\langle 2 \rangle\!\rangle \Box \neg \mathsf{caR}$ and $M_1', 1 \models (\mathbf{str}_3 nop)\langle\!\langle \rangle\!\rangle \Box \mathsf{nofuel}$. ■

In ATL+I, agents' current strategies are added to typical ATL models via modal relations $\mathcal{I}_a$. This resembles to some extent the semantics of epistemic temporal strategic logic from [19], where ATL-like formulae are interpreted over models *and* strategies. However, the strategies in [19] are used mostly as a technical device to define the semantics of cooperation modalities: they cannot be referred to in the object language of ETSL, and they change only in a very limited way on the semantic side.

The counterfactual intention operator $(\mathbf{str}_a\sigma)$, on the other hand, is very similar to the commitment operator from [17]. However, committing to a strategy is modeled in [17] through an update operator that *removes* the unintended choices from the system, and hence it refers to *irrevocable* commitments. Here, intended strategies can be freely revised or revoked, which makes our proposal close to Stalnaker's work on hypothetical reasoning about strategies [15], cf. Section 3.3 for more discussion.

Remark 1. Our semantics of cooperation modalities deviates from the original semantics of ATL [1] in two respects. First, we employ "memoryless" strategies in this paper, while in [1] strategies assign agents' choices to *sequences* of states (which suggests that agents can recall the whole history of the game). It should be pointed out that both types of strategies yield equivalent semantics for "vanilla" ATL, although the choice of one or another notion of strategy affects the semantics (and complexity) of the full ATL* and most ATL variants for games with incomplete information. Thus, we use memoryless strategies to increase the simplicity and extendability of our approach.

Second, we allow for non-deterministic strategies here, while only deterministic strategies are used in [1]. One reason is that we need the "all actions possible" strategic term any to express some important properties. Additionally, we consider non-deterministic strategies vital for modeling situations in which some agents may play at random (inherent nondeterminism)[3] or we have only partial information about agents' intentions (underspecification). Note that, if agents A have a non-deterministic strategy

[2] The "empty set" cooperation modality $\langle\!\langle \rangle\!\rangle$ is equivalent to the CTL's "for every path" quantifier A. Similarly, $\langle\!\langle \mathbb{A}\mathrm{gt} \rangle\!\rangle$ is equivalent to the CTL's "there is a path" quantifier E.

[3] This interpretation makes nondeterministic strategies similar to *mixed strategies* from game theory. However, we do not assume any probability distribution for the agents' choices here.

S_A to guarantee φ *for all computations that may result from playing* S_A, then every deterministic sub-strategy of S_A guarantees φ as well. In consequence, non-deterministic strategies do not change the semantics of cooperation modalities (even for ATL*). ■

It might be convenient to add collective strategies to the language of ATL+I. For $A = \{a_1, ..., a_r\}$, we define:

$$(\mathbf{str}_A\langle\sigma_{a_1}, ..., \sigma_{a_r}\rangle)\varphi \equiv (\mathbf{str}_{a_1}\sigma_{a_1})...(\mathbf{str}_{a_r}\sigma_{a_r})\varphi.$$

In what follows, we will sometimes overload the symbol any to denote a tuple of strategies $\langle\, \mathsf{any}\,, ..., \,\mathsf{any}\,\rangle$.

Remark 2. ATL+I semantically subsumes the original "pure" ATL from [1], as ATL models can be treated as a special case of ATL+I models, in which every available choice is possibly intended by agents at each state. ■

Remark 3. ATL+I syntactically subsumes ATL, as the ATL formulae $\langle\langle A\rangle\rangle\bigcirc\varphi$, $\langle\langle A\rangle\rangle\Box\varphi$, and $\langle\langle A\rangle\rangle\varphi\mathcal{U}\psi$ are equivalent to ATL+I formulae $(\mathbf{str}_{\mathbb{A}\mathrm{gt}}\,\mathsf{any}\,)\langle\langle A\rangle\rangle\bigcirc\varphi$, $(\mathbf{str}_{\mathbb{A}\mathrm{gt}}\,\mathsf{any}\,)\langle\langle A\rangle\rangle\Box\varphi$, and $(\mathbf{str}_{\mathbb{A}\mathrm{gt}}\,\mathsf{any}\,)\langle\langle A\rangle\rangle\varphi\mathcal{U}\psi$ respectively. ■

3.1 Intentions *to Do* vs. Intentions *to Be*

In this paper – among other issues – we consider a particular notion of *intentions*. Most models from the classical literature on intentions [6,14] suggest that intentions refer to properties of *situations*, i.e. agents intend to *be* in a state that satisfies a particular property. However, another notion of "intending" seems to be equally common in everyday language (and even in papers that refer to agents from a more practical perspective, e.g. [13]): namely, an agent may intend to *do* a particular action or execute a plan. In fact, "intending to do" was already considered in [7]; however, in that work, intentions were treated as a secondary notion that had to be derived from primitive concepts like beliefs or desires. We propose to model these "dynamically-oriented" intentions as first-class entities instead. Having the intentions "to do" in the models, we can also enable reasoning about them in the object language via another modal operator Int_a with the following semantics:

$$M, q \models \mathsf{Int}_a\sigma \textit{ iff for each } \alpha \in \textit{Act we have } q\mathcal{I}_a\alpha \Leftrightarrow \alpha \in [\![\sigma]\!]_a(q).$$

Note that $\mathsf{Int}_a\sigma$ formalizes a *local* notion of intention, i.e. intention to do an *action* in a particular state, while the counterfactual operator $(\mathbf{str}_a\sigma)$ is global in its scope. Collective intentions can be defined as:

$$\mathsf{Int}_{\{a_1,...,a_r\}}\langle\sigma_{a_1}, ..., \sigma_{a_r}\rangle \equiv \mathsf{Int}_{a_1}\sigma_{a_1} \wedge ... \wedge \mathsf{Int}_{a_r}\sigma_{a_r}.$$

Furthermore, intentions "to be" can be defined as follows. Let us assume that nondeterministic strategies model *genuine non-determinism* of agents, i.e. that, if $q\mathcal{I}_a\{\alpha_1, \alpha_2, ...\}$ then agent a does not know himself whether he is going to execute α_1 or α_2 or ... etc. in state q. Under this interpretation, we propose the following definition of coalition A's intentions "to be" (we call such intentions *goals* after Cohen and Levesque):

$$\mathsf{Goal}_A\varphi \equiv (\mathbf{str}_{\mathbb{A}\mathrm{gt}\setminus A}\ \mathsf{any}\,)\langle\!\langle\rangle\!\rangle\varphi \wedge \neg(\mathbf{str}_{\mathbb{A}\mathrm{gt}}\ \mathsf{any}\,)\langle\!\langle\rangle\!\rangle\varphi.$$

That is, A intend to bring about goal φ iff φ is an inevitable consequence of A's intended strategy, regardless of what other agents do – but φ is not *"physically"* inevitable (i.e. inevitable for all possible intentions, cf. Section 3.3). The definition is somewhat preliminary, since it does not fully address e.g. unwelcome but inevitable consequences of one's intended course of action; we hope to discuss such issues further in future work. Note that, in the above definition, φ is a property of paths (courses of action) rather than states. Thus, Goal_a says which courses of action a intends to take part in (or bring about), rather than which states he intends to be in. This approach allows us to express subtle differences between various types of an agent's intentions "to be": the agent may intend to be in a state that satisfies φ right in the *next* moment ($\mathsf{Goal}_a\bigcirc\varphi$), or he may intend to *eventually* bring about such a state ($\mathsf{Goal}_a\Diamond\varphi$), or be in "safe" states all the time ($\mathsf{Goal}_a\Box\mathsf{safe}$) etc. For instance, the "lazy" strategy of agent 3 in model M_2 (Example 2) implies that the rocket will never get out from London if 1 is the initial state – regardless of what 1 and 2 do. Thus, $M_2, 1 \models \mathsf{Goal}_3\Box\mathsf{roL}$.

3.2 A Dynamic Logic of Strategies

It should be easy to see from previous examples how we can reason about outcomes of agents' strategies with ATL+I. We point out that our $(\mathbf{str}_a\sigma)$ operator can be used to facilitate reasoning about strategies in the style of dynamic logic [10]. In particular, formulae $[A/\sigma]\varphi$ meaning that *"every execution of strategy σ by agents A guarantees property φ"*, or, more precisely, *"for every execution of strategy σ by A, φ inevitably holds (regardless of what other agents do)"* can be defined as:

$$[A/\sigma]\varphi \equiv (\mathbf{str}_A\sigma)(\mathbf{str}_{\mathbb{A}\mathrm{gt}\setminus A}\ \mathsf{any}\,)\langle\!\langle\rangle\!\rangle\varphi.$$

Note that, in that case, φ should be a temporal formula (path formula), as execution of a strategy is a process that happens over time.

Moreover, we observe that strategies in ATL are very similar to the way in which *programs* (or *actions*) are modeled in dynamic logic. In fact, our strategic terms and their denotations can refer to both strategies and actions. The difference lies not in the semantic representation of actions vs. strategies, but in the way their execution is understood: actions are one-step activities, while a strategy is executed indefinitely (or until it is replaced with another strategy). A fragment of propositional dynamic logic can be embedded in ATL+I with the following definitions, where σ is a program executed by the grand coalition of agents (i.e. by the whole *system*):

$$[\sigma]\varphi \equiv [\mathbb{A}\mathrm{gt}/\sigma]\bigcirc\varphi, \text{ and consequently}$$
$$<\!\sigma\!>\varphi \equiv \neg[\mathbb{A}\mathrm{gt}/\sigma]\bigcirc\neg\varphi.$$

A richer language of strategic terms is needed to embed the full syntax of PDL in ATL+I.

3.3 Properties of Intention Revision in ATL+I

Proposition 1. *Let φ be a formula of* ATL+I*, and let* $\mathbf{Ph} \equiv (\mathbf{str}_{\mathbb{A}\mathrm{gt}}\ \mathsf{any}\,)$ *be a shorthand for the counterfactual operator that yields the bare, "physical" system without any specific intentions assumed (i.e. the system with all actions "marked" as possibly intended by respective agents). The following formulae are tautologies of* ATL+I*:*

1. $(\mathbf{str}_a\sigma_1)(\mathbf{str}_a\sigma_2)\varphi \leftrightarrow (\mathbf{str}_a\sigma_2)\varphi$*: a new intention cancels the former intention.*
2. $(\mathbf{Ph}\langle\langle\rangle\rangle\bigcirc\varphi) \rightarrow (\mathbf{str}_a\sigma)\langle\langle\rangle\rangle\bigcirc(\mathbf{Ph}\varphi)$, $\quad(\mathbf{Ph}\langle\langle\rangle\rangle\Box\varphi) \rightarrow (\mathbf{str}_a\sigma)\langle\langle\rangle\rangle\Box(\mathbf{Ph}\varphi)$, *and* $(\mathbf{Ph}\langle\langle\rangle\rangle\varphi\,\mathcal{U}\psi) \rightarrow (\mathbf{str}_a\sigma)\langle\langle\rangle\rangle(\mathbf{Ph}\varphi)\,\mathcal{U}(\mathbf{Ph}\psi)$.
3. $(\mathbf{str}_a\sigma)\langle\langle\mathbb{A}\mathrm{gt}\rangle\rangle\bigcirc(\mathbf{Ph}\varphi) \rightarrow (\mathbf{Ph}\langle\langle\mathbb{A}\mathrm{gt}\rangle\rangle\bigcirc\varphi)$*, and similarly for* $\Box\varphi$ *and* $\varphi\,\mathcal{U}\psi$.

The counterfactual operator $(\mathbf{str}_A\sigma)$ is based on model update, which makes it similar to the preference operator from [20] and the commitment operator from [17]. Unlike in those approaches, however, model updates in ATL+I are *not* cumulative (cf. Proposition 1.1). This is because the choices we assume unintended by a via $(\mathbf{str}_a\sigma)$ are not removed from the model, they are only left "unmarked" by the new intention-accessibility relation $\mathcal{I}_a$. The update specified by $(\mathbf{str}_a\sigma)$ does not change the "hard", temporal structure of the system, it may only change the "soft" modal relations that encode agents' mental attitudes. In a way, it makes it possible to distinguish between the "physical" abilities of agents, and their intentional stance. Two important properties of such non-cumulative model updates are addressed by Propositions 1.2 and 1.3: first, a property that holds in the next moment for all *physical* paths of a system, is also *physically* true in the next moment for the paths consistent with agents' intentions; second, if there is an intentionally possible path along which ϕ holds physically in the next moment, then such a path exists in the system physically as well. Similar results hold for other temporal operators. We note that properties 1.2 and 1.3 are analogues of Lemma 1 from [18] and Proposition 2 from [17], but it is not necessary to restrict their scope to universal (resp. existential) formulae in ATL+I.

An interesting kind of property that can be expressed in ATL+I is: $(\mathbf{str}_A\sigma)\langle\langle\rangle\rangle\Box(\varphi\wedge(\mathbf{str}_A\,\mathsf{any}\,)\langle\langle A\rangle\rangle\Diamond\neg\varphi)$: agents A can use strategy σ to enforce that always φ, and at the same time retain *physical* ability to falsify φ. For instance, for our rocket agents, we have that $M_2, 1 \models (\mathbf{str}_{2,3}\langle nop_2, nop_3\rangle)\langle\langle\rangle\rangle\Box(\mathsf{roL}\wedge(\mathbf{str}_{2,3}\,\mathsf{any}\,)\langle\langle 2,3\rangle\rangle\Diamond\mathsf{roP})$. Note that this kind of property is not even *satisfiable* in logics with models updated by removing transitions, e.g. for the "ATL+commitment" logic introduced in [18].

ATL+I makes it also possible to discuss the dynamics of intentions: we can consider what happens if some agents change their strategies after some time. For example, formula $(\mathbf{str}_b\sigma_1)\langle\langle\rangle\rangle\Diamond(\mathbf{str}_b\sigma_2)\langle\langle\rangle\rangle\Diamond\varphi$, says that φ must be eventually achieved if agent b starts with playing strategy σ_1, but after some time switches to σ_2. Another formula, $(\mathbf{str}_b\sigma_1)\langle\langle a\rangle\rangle\Diamond((\mathbf{str}_b\,\mathsf{any}\,)\langle\langle a\rangle\rangle\Box\varphi)$, states that, if b plays σ_1 initially, then a can secure φ afterwards, even if b changes his strategy. (Example: if b refrains from selling his assets of company a for some time, then a can keep away from bankruptcy, regardless of what b decides to do when a's recovery plan has been executed.)

3.4 Model Checking ATL+I

The *model checking problem* for ATL+I is the problem of determining, for any given ATL+I formula φ, model M, and state q in M, whether or not $M, q \models \varphi$. There are three reasons for the importance of model checking. First, in many real-life situations, it is relatively easy to come up with a "natural" model of the reality. Next, checking if a property holds in a given model is computationally less expensive than checking if it holds in *all* models. Finally, the idea of "planning as model checking" [9] gives it a practical flavor: model checking algorithms can be adapted for generating plans in various domains.

The following algorithm extends the ATL model checking algorithm from [1] to compute the set of states Q_φ in which φ holds in M.

- Cases $\varphi \equiv p$, $\neg\psi$, $\psi_1 \wedge \psi_2$: tackle in the standard way.
- Case $\varphi \equiv (\mathbf{str}_a\sigma)\psi$: compute $M' = revise(M, a, [\![\sigma]\!]_a)$, and check ψ in M'.
- Case $\varphi \equiv \langle\!\langle A\rangle\!\rangle\bigcirc\psi$: compute Q_ψ for the *original* model M, then go through M deleting transitions where any agent a performs an action not dictated by $\mathcal{I}_a$. Finally, use the ATL model checking algorithm for formula $\langle\!\langle A\rangle\!\rangle\bigcirc Q_\psi$ and the resulting ("trimmed") model.
- Cases $\varphi \equiv \langle\!\langle A\rangle\!\rangle\Box\psi$, $\langle\!\langle A\rangle\!\rangle\psi_1\,\mathcal{U}\psi_2$: analogous.

Let us observe that given M, a, and σ, computing $revise(M, a, s_a)$ and the "trimming" procedure can be done in time $O(m)$, where m is the number of transitions in M. As ATL model checking enjoys complexity of $O(ml)$, it gives us the following result.

Proposition 2. *Model checking an* ATL+I *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.*

4 Reasoning About Rational Intentions

Using the counterfactual operator $(\mathbf{str}_A\sigma)$, we do not assume anything about payoffs and/or preferences of players, about their rationality, optimality of their strategies etc. – we simply assume that A intend to play σ (for whatever reasons), and ask what are the consequences. Reasoning about *rational* agents can be done on top of this: we should define what it means for an intention to be rational and then reason about outcomes of such intentions with $(\mathbf{str}_A\sigma)$.

There is a growing literature on using temporal [5], dynamic [11,16], and ATL-style logics [17,4] for reasoning about solution concepts.[4] We note that ATL operators $\langle\!\langle A\rangle\!\rangle$ can be seen as a formalization of reasoning about strategic game forms [17], or, even more naturally, *extensive game forms* – since concurrent game structures generalize[5] game trees with perfect information, except for agents' utilities. In order to "emulate" utilities, we follow the approach of [2]. Let U denote the set of all possible utility values in the game; U will be fixed and finite for any given game. For each value $v \in U$ and agent $a \in \mathbb{A}\mathrm{gt}$, we introduce a proposition $(u_a \geq v)$ into our set Π of primitive propositions, and fix the valuation function π so that $(u_a \geq v)$ is satisfied in state q iff a gets at least v in q. The correspondence between a traditional game tree Γ and a concurrent game structure M can be captured as follows. Let $\Gamma = \langle \Sigma, \mathcal{A}, H, ow, u\rangle$, where Σ is a finite set of players, $\mathcal{A}$ a finite set of actions, H a

[4] Among these, we come perhaps closest to [11] with our approach in this section. In [11], however, the notion of Nash equilibrium is captured via propositional *dynamic* logic, which restricts the discussion to traditional games on finite trees (since only properties of strategies whose execution *terminates* can be addressed). Another consequence of using PDL is that outcomes of strategies are classically defined in terms of properties achieved *eventually* in terminal states, while we propose a more general approach (i.e. temporal properties of *runs*).

[5] Concurrent game structures may include cycles and simultaneous moves of players, which are absent in game trees.

finite set of (finite) action sequences (i.e. legal game histories), and $ow(h)$ defines which player "owns" the next move after history h. We define the set of actions available at h as $\mathcal{A}(h) = \{\alpha \mid h\alpha \in H\}$, and the set of terminal situations as $Term = \{h \mid \mathcal{A}(h) = \varnothing\}$. Function $u : \Sigma \times Term \to U$ assigns agents' utilities to every final position of the game [12]. We say that $M = \langle \mathbb{A}\mathrm{gt}, Q, \Pi, \pi, Act, d, o\rangle$ corresponds to Γ iff: (1) $\mathbb{A}\mathrm{gt} = \Sigma$, (2) $Q = H$, (3) Π and π include propositions $(u_a \geq v)$ to emulate utilities for terminal states in the way described above, (4) $Act = \mathcal{A} \cup \{nop\}$, (5) $d_a(q) = \mathcal{A}(q)$ if $a = ow(q)$ and $\{nop\}$ otherwise, (6) $o(q, nop, ..., \alpha, ..., nop) = q\alpha$, and (7) $o(q, nop, nop, ..., nop) = q$ for $q \in Term$. Additionally, for an ATL+I model M' that adds intentions and strategic terms to M, we define that Γ corresponds to M' iff Γ corresponds to M and $q\mathcal{I}_a\alpha$ for every $q \in Q, a \in \mathbb{A}\mathrm{gt}, \alpha \in d_a(q)$ (all choices are possibly intended). Note that for every extensive form game Γ, there is a corresponding concurrent game structure, but the reverse is not true.

Now we can show how Nash equilibrium can be specified in ATL+I, and how one can reason about outcomes of agents whose rationality is defined in terms of Nash equilibrium. As games specified by concurrent game structures are usually infinite, there are no terminal positions in these games in general. Therefore it seems reasonable to define outcomes of strategies via properties of resulting paths (courses of action) rather than single states.[6] For example, we may be satisfied if a utility value v is achieved eventually: $\Diamond(u_a \geq v)$, preserved until a property p holds: $(u_a \geq v)\,\mathcal{U}\mathsf{p}$ etc. To capture such subtleties, we propose the notion of *T-Nash equilibrium*, parametrized with a unary temporal operator $T = \bigcirc, \Box, \Diamond, \_\mathcal{U}\psi, \psi\mathcal{U}\_$. Thus, we have a family of equilibria now: $\bigcirc$-Nash equilibrium, $\Box$-Nash equilibrium etc. Let σ describe a collective strategy for the grand coalition $\mathbb{A}\mathrm{gt}$, and let $\sigma[a]$ be the strategic term for a's strategy in σ. Similarly, $\sigma[A]$ is the part of σ that refers to the strategy of A. We write $BR_a^T(\sigma)$ to denote the fact that strategy $\sigma[a]$ is a_i's best response to $\mathbb{A}\mathrm{gt} \setminus \{a\}$ playing $\sigma[\mathbb{A}\mathrm{gt} \setminus \{a\}]$. For example, $BR_a^{\Box}(\sigma)$ means that a cannot increase his minimal guaranteed payoff by deviating from $\sigma[a]$ unilaterally. Likewise, $BR_a^{\Diamond}(\sigma)$ says that a cannot increase his maximal guaranteed payoff (i.e. the payoff that can be obtained *eventually* along *every* possible course of action) by a unilateral deviation from $\sigma[a]$. We write $NE^T(\sigma)$ to denote the fact that σ is a T-Nash equilibrium.

$$BR_a^T(\sigma) \equiv (\mathbf{str}_{\mathbb{A}\mathrm{gt}\setminus\{a\}}\sigma[\mathbb{A}\mathrm{gt} \setminus \{a\}])(\bigwedge_{v \in U} (\langle\!\langle a \rangle\!\rangle T(u_a \geq v)) \to (\mathbf{str}_a\sigma[a])\langle\!\langle\rangle\!\rangle T(u_a \geq v))$$

$$NE^T(\sigma) \equiv \bigwedge_{a \in \mathbb{A}\mathrm{gt}} BR_a^T(\sigma).$$

Proposition 3. *Let Γ be a game, and M a concurrent game structure with intentions, corresponding to Γ. Then $M, \varnothing \models NE^{\Diamond}(\sigma)$ iff σ denotes a Nash equilibrium in Γ.*

Thus, Nash equilibrium in traditional games is the special case of our temporal Nash equilibrium, in which we ask about utilities one must get eventually at the end of the game. NE^T extends this notion by focusing on temporal patterns rather than single utility values. Moreover, as concurrent game structures specify interactions that are usually infinite and may include simultaneous moves of players (as well as cycles of transitions), the concept of Nash equilibrium naturally extends to such generalized games in our definition.

[6] The idea of assigning utilities to *runs* rather than states is not entirely new, cf. [21].

1	x, 1, 0	7	x, −3, −1
2	x, 1, 2	8	x, −3, 1
3	x, 1, 0	9	1, −3, 0
4	x, 1, 2	10	1, −3, 2
5	x, 1, −1	11	1, −3, 0
6	x, 1, 1	12	1, −3, 2

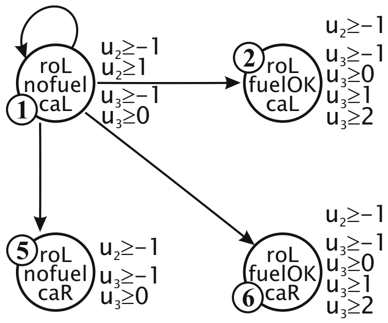

Fig. 2. The rocket game: Utility table and a fragment of the game structure

Example 3. Let us consider an infinite game played by the "rocket agents" from previous examples. Figure 2 shows the table of utilities for the game, as well as a fragment of system M_3, that augments M_1' with propositions encoding agents' utilities. Note that, unlike for game structures corresponding to traditional game trees, there are no final states in the model, and utility values are defined for most states.

Let $carry$ denote the strategy for agent 1, in which the agent loads the cargo in states $1, 2, 5$, moves the rocket in states $4, 6$, unloads the cargo in $7, 8$ and does nothing in $3, 9, 10, 11, 12$. Moreover, $fuel$ denotes the strategy in which 3 executes $fuel_3$ in $1, 3, 5, 7, 9, 11$, and nop_3 elsewhere. Now, $M_3, 1 \models NE^{\Diamond}(\langle carry, nop, fuel\rangle)$ because $BR_1^{\Diamond}(\langle carry, nop, fuel\rangle)$ and $BR_2^{\Diamond}(\langle carry, nop, fuel\rangle)$ and $BR_3^{\Diamond}(\langle carry, nop, fuel\rangle)$. Also, $M_3, 6 \models NE^{\Box}(\langle nop, nop, nop\rangle)$: 2 and 3 are satisfied at state 6, and 1 cannot achieve $\Box\mathsf{caP}$ anyway. Thus, the system is in $\Diamond$-Nash equilibrium in state 1, and in $\Box$-Nash equilibrium in state 6. ■

Properties of rational strategies can be now verified through formulae of form $NE^T(\sigma) \wedge (\mathbf{str}_{\mathbb{A}\mathrm{gt}}\sigma)\varphi$, where φ is the property we would like to check. For example, we have that $M_3, 6 \models NE^{\Box}(\langle nop, nop, nop\rangle) \wedge (\mathbf{str}_{\mathbb{A}\mathrm{gt}}\langle nop, nop, nop\rangle)\langle\langle\rangle\rangle\Box\mathsf{caR}$.

Remark 4. Building upon the concept of Nash equilibrium, we may like to express rationality of strategies as: *"$rational_A^T(\sigma_A)$ iff there is $\sigma'_{\mathbb{A}\mathrm{gt}\setminus A}$ such that $NE^T(\sigma_A, \sigma'_{\mathbb{A}\mathrm{gt}\setminus A})$"*. In a similar way, it seems natural to reason about behavior of rational agents with sentences like *"suppose that A intend to play any strategy σ_A such that $rational_A^T(\sigma_A)$, then φ holds"*, Note that reasoning of this kind is beyond the scope of ATL+I, as the logic does not include explicit quantification over strategies yet. ■

5 Conclusions

What ATL offers, is in fact an abstraction of strategies. ATL modalities quantify over strategies in game theory-like fashion, but the strategies are hidden in the semantics: we can only specify *who* can do *what* and *when* in the object language of ATL, but we cannot tell *how* it can be done. In this paper, we propose to extend ATL with a notion of agents' intentions, and with an operator that enables addressing agents' strategies explicitly. The resulting logic, ATL+I, provides a formal language to express (and reason about) facts concerning strategies of agents in multiagent settings. We believe that the logic offers more than just a sum of its parts: counterfactual reasoning in game-like situations, dynamic logic of strategies, intention revision, rationality criteria, reasoning about rational intentions as well as relationship between intentions and goals are example issues that can be formalized and investigated with ATL+I. Thus, most of all, we see ATL+I as a potent framework for modeling and specifying systems that include multiple agents, and for discussing and verifying their properties.

We thank Peter Novak and the anonymous reviewers for their helpful remarks.

References

1. R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-time Temporal Logic. *Journal of the ACM*, 49:672–713, 2002.
2. A. Baltag. A logic for suspicious players. *Bulletin of Economic Research*, 54(1):1–46, 2002.
3. A. L. Blum and M. L. Furst. Fast planning through graph analysis. *Artificial Intelligence*, 90:281–300, 1997.
4. G. Bonanno. Modal logic and game theory: Two alternative approaches. *Risk Decision and Policy*, 7:309–324, 2002.
5. G. Bonanno. A characterization of von Neumann games in terms of memory. *Synthese*, 139(2):237–256, 2004.
6. M.E. Bratman. *Intentions, Plans, and Practical Reason*. Harvard University Press, 1987.
7. P.R. Cohen and H.J. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42:213–261, 1990.
8. E. A. Emerson. Temporal and modal logic. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, pages 995–1072. Elsevier Science Publishers, 1990.
9. F. Giunchiglia and P. Traverso. Planning as model checking. In *ECP*, pages 1–20, 1999.
10. D. Harel, D. Kozen, and J. Tiuryn. *Dynamic Logic*. MIT Press, 2000.
11. B.P. Harrenstein, W. van der Hoek, J.-J. Meyer, and C. Witteveen. A modal characterization of Nash equilibrium. *Fundamenta Informaticae*, 57(2–4):281–321, 2003.
12. M. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, 1994.
13. M. Pollack. The uses of plans. *Artificial Intelligence*, 57(1):43–68, 1992.
14. A.S. Rao and M.P. Georgeff. Modeling rational agents within a BDI-architecture. In *Proceedings of KR-91*, pages 473–484, 1991.
15. R. Stalnaker. Knowledge, belief and counterfactual reasoning in games. *Economics and Philosophy*, 12:133–163, 1996.
16. J. van Benthem. Games in dynamic epistemic logic. *Bulletin of Economic Research*, 53(4):219–248, 2001. Proceedings of LOFT-4.
17. W. van der Hoek, W. Jamroga, and M. Wooldridge. A logic for strategic reasoning. In *Proceedings of AAMAS'05*. 2005.
18. W. van der Hoek, M. Roberts, and M. Wooldridge. Social laws in alternating time: Effectiveness, feasibility and synthesis. *Synthese*, 2005.
19. S. van Otterloo and G. Jonker. On Epistemic Temporal Strategic Logic. In *Proceedings of LCMAS*, pages 35–45, 2004.
20. S. van Otterloo, W. van der Hoek, and M. Wooldridge. Preferences in game logics. In *Proceedings of AAMAS-04*, 2004.
21. M. Wooldridge. *An Introduction to Multi Agent Systems*. John Wiley & Sons, 2002.