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.

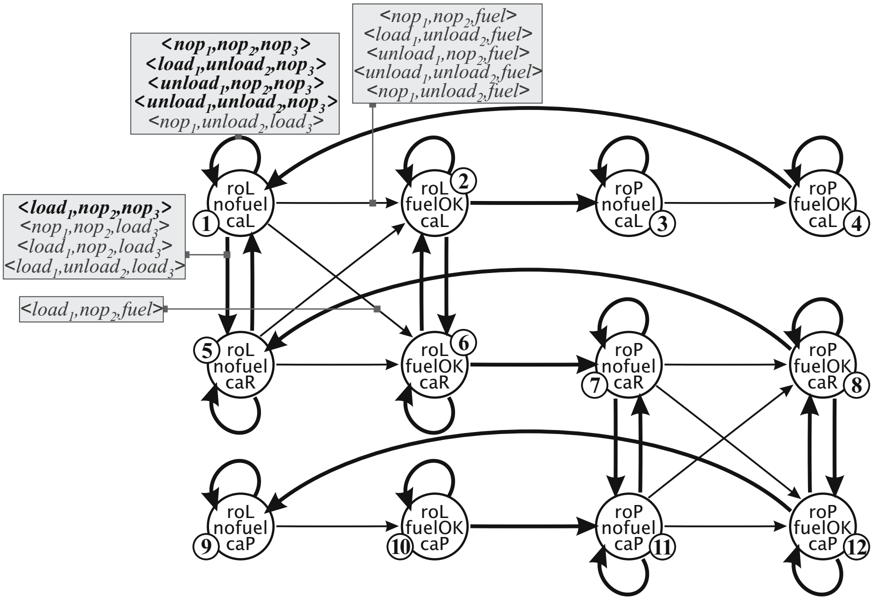

Fig. 1. Simple Rocket Domain. The "bold" transitions are the ones in which agent 3 intends to always choose nop_3.

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 , $\Diamond\varphi \equiv \top\,\mathcal{U}\varphi$.