An Architecture for Multiple Bilateral Negotiation (Extended Abstract)

Angela Fabregues
Artificial Intelligence Research Institute (IIIA-CSIC)
Campus Universitat Autònoma de Barcelona, 08193 Bellaterra, Catalonia, Spain
fabregues@iiia.csic.es

ABSTRACT

Automated negotiation agents that aim to proficiency negotiate in realistic scenarios involving humans in repeated environments have some special requirements; e.g. they must deal with emotions and uncertainty. In this paper we propose an architecture for agents to bilaterally negotiate on plans of action with several other agents (also humans).

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems

General Terms

Theory

Keywords

Automated negotiation, architecture, plan search.

1. INTRODUCTION

We address in this paper the complex problem of simultaneous, repeated and bilateral negotiations in competitive multiagent environments. The agents are either software or human agents that compete but that can occasionally cooperate. The negotiation objects are joint plans of action (that we call negotiation options). We are specially interested in negotiation domains that have a very large set of potential joint action plans as these are those with potential commercial interest (e.g. time tabling, team formation, supply chain management, gaming). In these scenarios, agents (and humans) need to negotiate to improve their outcome. For instance, teachers swapping time slots in their class schedules, or members of a potential team negotiating by pairs the tasks to be performed. The environment is generally observable but the internal state of the other agents and their negotiations are usually private, that is, every agent can see the messages that it sends or receives but not the messages exchanged between any two other agents of the multiagent system.

In open systems, reaching agreements on joint action plans is the way to figure out what our counterparts will do, and even this only to a certain extent as in some domains defection is possible. Negotiations are usually time framed. There is a deadline by which a negotiation process has to be finished. When deadlines are tight negotiators need to search quickly for good enough negotiation proposals (options) instead of looking for optimal proposals. For large solution spaces it is either not possible or too long to find them. If an agent waits too long others may have reached agreements that are incompatible with the plans the agent likes.

The scenarios we are interested in witness repeated negotiations, for instance teachers negotiate every semester, or members of teams negotiate tasks for each problem to solve. These repeated interactions permit agents to build relationships, check whether the agreements are kept and act accordingly. If an agent breaks an agreement, it may become untrustworthy and the other agent involved in the agreement may penalise it. A good way to penalise an agent is ignoring it, rejecting every proposal it makes as it makes little sense to reach agreements with someone that is untrustworthy: it will probably break the deal. In summary, we address the problem of simultaneous bilateral negotiation of joint plans of action in competitive environments with repeated negotiation encounters and repeated rounds of plan execution. In these environments negotiation speed is crucial because as time goes by the number of available joint plans that can be accepted decreases.

In this paper we propose the COncurrent Multi-BIlateral NEgotiation (COMBINE) architecture that supports the development of agents suitable to work in scenarios like the introduced before. Our aim is to facilitate the research on automated negotiation in realistic scenarios. In [3] we already provided a testbed for them. Now we are supporting the development of the agents proposing this architecture.

2. COMBINE ARCHITECTURE

The architecture is graphically represented in Figure 1. It provides an *interface module* that situates COMBINE agents in their environment, that is, it allows them to observe the environment state, observe and execute actions, and exchange messages with other agents. In other words, this module contains the sensors and actuators of the agent. Which actions to execute and which messages to send is decided by the negotiation module.

The design philosophy behind the COMBINE architecture is to provide some means to negotiate as humans do, as the negotiation counterparts could be humans. In particular, there are two capabilities that we think realistic agents should show: dealing with emotions and dealing with uncertainty. The architecture incorporates emotions as this

Appears in: *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, Conitzer, Winikoff, Padgham, and van der Hoek (eds.), June, 4–8, 2012, Valencia, Spain.

is an important part of the non-constructivist rationality approach, we need to understand emotional reactions of the other negotiators. Although the environment is fully observable, the actions to be executed by the other agents can only be guessed analysing the other agents' previous behaviour. To cope with this uncertainty, we decided to represent the world as a graded BDI model, that is with graded *beliefs*, *desires* and *intentions* following the g-BDI model of [2]. Several other models can be incorporated to the world model providing and/or modifying beliefs, desires and intentions. For instance, a normative system can provide internal norms for the agent to verify interaction protocols.

The space of plans and negotiation options that an agent can execute and propose, respectively, is potentially huge. Thus, we assume that the space is large enough and the negotiation time short enough to preclude obtaining the optimal. That means that any architecture for this type of negotiation needs to give the means to look for good enough solutions. Moreover, the longer it takes to decide what to propose the less probable it is the proposal to be accepted. As time goes by, the agents reach agreements increasing the amount of commitments and reducing the set of options compatible with the commitments. This increases, as time goes by, the probability that our desired plans will not be compatible with the acquired commitments. Consequently, the architecture must allow to start negotiating from the very beginning of a negotiation round.

Dealing with huge solution spaces is not an inconvenient for human agents, e.g. in playing Chess or Go. Humans do work with good enough solutions in their everyday lives. Time constraints, boredom, or tiredness make humans accept good enough solutions. We use an any time algorithm to do the plan search. The generation of plans and options is done in two separate components. The *plan generator* basically searches for a subset of the feasible plans for the current environment state. These plans could possibly include actions to be performed by other agents. They are ranked by the *plan evaluator* that also drives the search of good enough plans. The options are generated from the *plan ranking* by the *option generator* that computes a subset of all the negotiating options resulting from the combination of actions from the ranked plans. This subset of combinations is also ranked according to an *option evaluator*. Both evaluators, for plans and options, base their evaluations on the current world model of the agent.

The elements perceived by the interface module (environment state, actions, messages or current time) update the beliefs of the agent. With new beliefs, the whole world model is updated (see the arrows in Figure 1) as well as the evaluators and the negotiation strategy. In this way, perception of new events can for instance change the ranking of the most appropriate options to send in a proposal. If another agent has quickly rejected a proposal we made, perhaps the similar options involving that agent should be demoted in the ranking as we may belief that the other agent is not keen on reaching an agreement like this. The update involves perhaps a change in the intentions of the agent and thus impacts on what the negotiation strategy should focus on. Similarly, when a proposal is received or when the environment changes, the data flow impacts on what decisions the negotiation strategy will take. Our own actions and messages also update the world model and consequently the rest of the elements of the architecture.

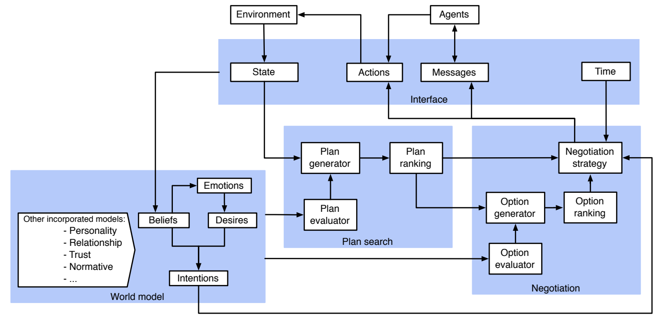

Figure 1: COMBINE architecture. Arrows represent data flows. Coloured boxes represent modules and white boxes are components of those modules.

Agents have to interact from the very beginning, and make proposals on options while they keep on searching for even better plans. The execution of an agent consists of several concurrent processes for: the *interface* (to receive messages and observe the results of actions and the environment state), the *world model* (to update the world model given the perceived changes), the *plan generator* (to continuously update the ranking of plans) and the *negotiation strategy* (to generate options from plans and determine what to do next).

As a summary of the architecture description we outstress a number of characteristics that differentiates this architecture from others [1, 4] and makes it suitable for negotiations involving humans in realistic scenarios:

- It is capable to deal with uncertainty, using a world model based on a graded BDI [2].
- It allows to include emotions and other cognitive aspects like trust.
- It is reactive to changes in the environment.
- It is proactive, from intentions and a reduced set of plans and options is capable of proposing deals.
- Negotiation goes hand in hand with plan search. New commitments reduce the search space and new found plans change the negotiation process.

Acknowledgments

Research supported by the Agreement Technologies CONSOLIDER project under contract CSD2007-0022 and INGENIO 2010.

3. REFERENCES

[1] R. Ashri, I. Rahwan, and M. Luck. Architectures for negotiating agents. In *Proc. of the 3rd Central and Eastern European conf. on Multi-agent systems*, CEEMAS'03, pages 136–146, Berlin, Heidelberg, 2003.

[2] A. Casali, L. Godo, and C. Sierra. A graded bdi agent model to represent and reason about preferences. *Artificial Intelligence*, 175:1468–1478, 2011.

[3] A. Fabregues and C. Sierra. Dipgame: a challenging negotiation testbed. *Journal of Engineering Applications of Artificial Intelligence*, 24:1137–1146, 10/2011 2011.

[4] K. Hindriks, C. Jonker, and D. Tykhonov. Towards an open negotiation architecture for heterogeneous agents. In *Cooperative Information Agents XII*, volume 5180 of *Lecture Notes in Computer Science*, pages 264–279. 2008.