Interac-DEC-MDP: Towards the use of interactions in DEC-MDP

Vincent Thomas, Christine Bourjot, Vincent Chevrier
Campus Scientifique - BP 239
54506 Vandoeuvre-les-Nancy, FRANCE
{vthomas, bourjot, chevrier}@loria.fr

1. Introduction

Extensions of Markov Decision Processes like DEC-POMDP [BZI00] and MMDP [Bou99] have been proposed as formalisms for automatic computation of collective behavior. However, in order to do so, decentralized approaches have to take into account the simultaneous evolution of the agents, and to face the issue of credit assignment. The usual mono-agent methods where egoistic agents are driven by a single reward cannot be applied anymore ([SPG03]). New mechanisms must be added to consider the global reward and the actions of other agents in the system, like empathy ([CSC02]) or elaborated communication ([PT02]).

The approach we propose is to get inspiration from biological systems to find new ways of coordinating independent learners at low costs and on the sole basis of local rewards. The Interac-DEC-MDP model is based on the reintroduction of an interaction module between agents observed in natural system in order to profit from simple individual leanings.

2. Interac-DEC-MDP

The Interac-DEC-MDP formalism is an extension of the DEC-MDP model where each agent has total observability and where the global reward is only partially perceived by each agent. The originality of Interac-DEC-MDP relies in the adding of an interaction module. Our interactions are defined as reactive mutual influences exerted by two agents. Thus, the resolution of interactions does not consider only egoistic interests but is based on an assessment, made by the two involved agents, of more global interests and can thus produce altruistic behaviors. The idea we have followed is that, if the interactions are well chosen, they can reduce the conflicts between agents at low costs.

3. Formalism

An Interac-DEC-MDP is constituted by two modules.

The first one is the *action* module. It corresponds to a classic DEC-MDP. During *action* phase, each agent decides autonomously its action and the system evolves according to the set of actions performed by all the agents. Each agent i receives then an individual local reward r_i. The aim of the system is to maximize the total reward earned by all the agents (as the sum of individual rewards). This task is difficult because each agent has only access to its local reward and cannot know what other agents have earned. The second module is an *interaction* module. Considered interactions only involve two agents. During the *interaction* phase, each agent sequentially decides which interaction using and with which agent interacting. Once an interaction is triggered, the two involved agents exchange signals related to their expected individual reward and accordingly, decide collectively the future evolution of the system. The way interactions between agents are represented is general enough to represent various kinds of interaction (for each interaction, there are various results, each one is characterized by a transition matrix).

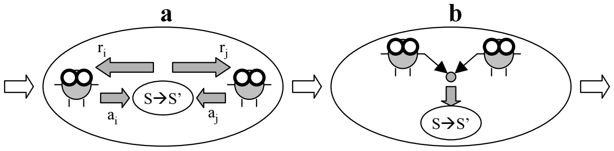

Figure 1. a) action and b) interaction modules

It must be noticed that an Interac-DEC-MDP without any interaction corresponds exactly to the associated DEC-MDP. The main asset of Interac-DEC-MDP formalism is that actions and interactions are represented in the same mathematical framework and can be simultaneously taken into account to compute the agent behaviors: action module focus on individual interests, interaction module on more collective ones.

4. Solving an Interac-DEC-MDP

An Interac-DEC-MDP system is thus characterized by the policies of all the agents in the action phase, the policy related to the triggering of an interaction (which interaction and with which agent interacting) and collective decision making during interaction. For each couple of agents, an interaction policy is defined and characterizes this collective decision process. Solving an interac-DEC-MDP consists in determining all these action and interaction policies.

A first very simple algorithm based on Reinforcement learning has been tested. It can be divided into two phases. During the first phase, each ”independent learner” agent learns its action policy or in other words, how to react in the system without the presence of interaction. This phase corresponds to a simple Q-learning where each agent tries to maximize its egoistic interests.

During the second phase, the action policy is frozen and agents learn how to interact in order to take advantages of the previous individual learnings. The result of interaction between agents is decided according to the sum of the previous estimated individual utilities of the involved agents (for greedy interaction policy, the chosen result corresponding to the argmax of this sum). Interaction policies can thus be reconstructed whenever agents interact thanks to local numerical signal exchanges and do not need elaborated representations. The underlying assumption of this approach is that the sole individual utility of other agents (and not their whole policies) can be sufficient to make relevant collective decision.

5. Results

For a diving-for-food toy problem, agents can dive into water to fetch food, stay in the cage or eat food during action phase. They receive individual positive reward whenever they consume food and negative reward each time they dive. The interaction that has been added consists for the involved agents to exchange food. In the action phase, agents learn to dive into water in order to fetch food and to eat it and in interaction phase, they learn to exchange food on the basis of the previous individual learnings.

Several scenario have been tested. In these scenario, two agents are put together in this situation. Agents may have different swimming abilities (related to the negative reward $-\alpha$ received each time they dive). This simple learning give good results. When a good swimmer (for which $|\alpha|$ is low) is in the presence of a bad swimmer (for which $|\alpha|$ is high), exchange of food occurs. It has for consequence the decrease of the utility of the good swimmer agent (which has to dive more often in order to eat) but also the increase of the global utility of the system (because the bad swimmer do not have to dive, which has a high cost for the system). Moreover, the learning of triggering policies manage to limit the interactions only to the useful ones.

Results have been compared to optimal ones (computed with a centralized approach). It turns out that the collective behavior generated with this simple reinforcement learning is not the optimal one but is close to it. This is due to the fact that action policies are not questioned when interactions are introduced in the system. Now that interactions have been learned and can be considered by agents, it must interesting to re-learn the action policies.

6. Discussion

To conclude, the Interac-DEC-MDP has been applied to a toy problem, but, interaction module is general enough to represent various kinds of interactions. Moreover, even if it has only been tested with two agents, this formalism gave interesting results. We think it can be generalized for a more important number of agents if the interactions are only local interactions. The next step of our approach will be to test a more complex learning in a more complex task (like bucket-brigade task for example) in order to verify the spontaneous appearance of relevant organizational structure due to interactions (like, formation of chains between agents to carry buckets on the basis of an interaction consisting of taking a bucket from another agent). Endly, up to now, only total observability has been considered, in the long run, we plan to extend this model to partial observability in order to build scalable collective behaviour.

References

[Bou99] C. Boutilier. Sequential optimality and coordination in multiagent systems. In *Proceedings of IJCAI '99, Stockholm, Sweden*, 1999.

[BZI00] D.S. Bernstein, S. Zilberstein, and N. Immerman. The complexity of decentralized control of markov decision processes. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI'00), Stanford, California*, 2000.

[CSC02] I. Chades, B. Scherrer, and F. Charpillet. A heurisitic approach for solving decentralized-pomdp: Assessment on the pursuit problem. In *17th ACM Symposium on Applied Computing (SAC 2002)*, 2002.

[PT02] D.V. Pynadath and M. Tambe. The communicative multiagent team decision problem : analyzing teamwork theories and models. *Journal of intelligence research*, 16:389–423, 2002.

[SPG03] Y. Shoham, R. Powers, and T. Grenager. Multi-agent reinforcement learning: a critical survey. Technical report, 2003.