Combining Individual and Cooperative Learning for Multi-agent Negotiations

Leen-kiat Soh
Department of Computer Science and Engineering
University of Nebraska-Lincoln, 115 Ferguson Hall
Lincoln, NE 68588-0115 USA
001-402-472-6738
lksoh@cse.unl.edu

Juan Luo
Department of Computer Science and Engineering
University of Nebraska-Lincoln, 115 Ferguson Hall
Lincoln, NE 68588-0115 USA
001-402-472-6738
luo@cse.unl.edu

ABSTRACT
In this paper, we propose a distributed multi-strategy learning methodology based on case-based reasoning in which an agent conducts both individual learning by observing its environment and cooperative learning by interacting with its neighbors. Cooperative learning is generally more expensive than individual learning due to the communication and processing overhead. Thus, our methodology employs a cautious utility-based adaptive mechanism to combine the two, an interaction protocol for soliciting and exchanging information, and the idea of a chronological casebase. Here we report on experimental results on the roles and effects of the methodology in a multiagent environment.

Categories and Subject Descriptors
I.2.11 [Artificial Intelligence]: Distributed Artificial Intelligence – intelligent agents, multiagent systems.

General Terms
Design.

Keywords
Distributed learning, cooperative learning, case-based reasoning, multiagent systems

1. INTRODUCTION
We propose a distributed multi-strategy learning methodology based on case-based reasoning (CBR) in a multiagent environment. Each agent is capable of individual and cooperative learning. Individual learning refers to learning based on an agent’s perceptions and actions, without communicating directly with other agents in the environment. Cooperative learning refers to learning through interaction among agents. The objective here is to learn something better to solve a problem that an agent has failed to solve or solve satisfactorily.

Our methodology employs a cautious utility-based adaptive mechanism to combine the two, an interaction protocol for soliciting and exchanging information, and the idea of a chronological casebase. In our multiagent environment, agents negotiate to collaborate on real-time tasks such as multi-sensor target tracking and CPU resource allocation. When agents negotiate, each follows a dynamically generated negotiation strategy. Each agent derives its negotiation strategy for each negotiation using CBR.

2. METHODOLOGY
Our agents learn how to negotiate better. The learned knowledge is encapsulated in cases: the negotiation task is the case problem; the negotiation strategy is the case solution; and the outcome of the negotiation is the case result. Our negotiation approach is argumentative [1], in which every agent may assume two different roles in its negotiation tasks. As an initiator, an agent tries to convince the responder to agree to give up certain resources or to help perform a task. As a responder, an agent evaluates the request against its own constraints. If the arguments supporting the request are convincing enough, then the responder will agree to a deal. Therefore, every agent maintains two casebases, one for each role. Under normal CBR operations, when an agent confronts a negotiation task, it retrieves the best case that matches the current problem, and adapts the solution of the best case as the negotiation strategy. The agent then uses the negotiation strategy for its negotiation. When the negotiation ends, it records the outcome of the negotiation. Then the agent determines whether to learn the new case—whether to store it in its casebases.

2.1 Chronological Casebases & Usage History
We utilize the notion of a chronological casebase in which each case is stamped with a time-of-birth (when it was created) and a time-of-membership (when it joined the casebase). All initial cases are given the same time-of-birth and time-of-membership. A foreign case, however, may have a much earlier time-of-birth than a time-of-membership when imported to a local casebase. In addition, we profile each case’s usage history (Table 1). An agent evaluates the performance of a case based on its usage history. If the case is poor in its performance, it may be replaced (or forgotten). If the case is deemed to be problematic, then a cooperative learning will be triggered and the case will be replaced. Table 2 lists the heuristics we use in tandem with the chronological casebase.

2.2 Individual Learning
When a negotiation completes, if the new case is useful and adds to the casebase’s diversity, the agent learns it. This constitutes the basis of the individual learning strategy of our methodology. If
the casebase’s size has reached a preset limit, then the agent considers replacing one of the existing cases with the new case. This feature enables our agents to perform both incremental and refinement learning. For the refinement learning, we use heuristics \textit{H1}, \textit{H2}, and \textit{H3}.

Table 1. The usage history that an agent profiles of each case

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>timesOfUsed</em></td>
<td>the number of times the case has been used</td>
</tr>
<tr>
<td><em>timesOfSuccessUsed</em></td>
<td>the number of times the case has been used in a successful negotiation</td>
</tr>
<tr>
<td><em>timesOfIncurNewCase</em></td>
<td>the number of times the usage of the case has led to a new case getting added to the casebase</td>
</tr>
<tr>
<td><em>timesOfRequest</em></td>
<td>the number of times the case has been designated as a problematic case, i.e., with very low utility</td>
</tr>
<tr>
<td><em>timeStamp</em></td>
<td>the last time that the case was used or the time when the case was added</td>
</tr>
</tbody>
</table>

Table 2. Heuristics and Cooperative Learning

<table>
<thead>
<tr>
<th>Heuristics</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>\textit{H1} Currency</td>
<td>If a case has not been used in a long time, then this case is more likely to be replaced.</td>
</tr>
<tr>
<td>\textit{H2} Evolution</td>
<td>With everything else equal, an old case is more likely to be replaced than a young case.</td>
</tr>
<tr>
<td>\textit{H3} Usefulness</td>
<td>If a case’s <em>timesOfSuccessUsed</em> is significantly small, then the case is more likely to be replaced.</td>
</tr>
<tr>
<td>\textit{H4} Solution Quality I</td>
<td>If a case has a high <em>timesOfUsed</em> but a low <em>timesOfSuccessUsed</em>, then it is a problematic case.</td>
</tr>
<tr>
<td>\textit{H5} Solution Quality II</td>
<td>If a case has a low <em>timesOfSuccessUsed</em> and a high <em>timesOfIncurNewCase</em>, then the solution of this case is probably not suitable for the problems encountered by the agent and it is a problematic case.</td>
</tr>
</tbody>
</table>

2.3 Cooperative Learning

However, if an agent consistently fails to negotiate successfully given a particular problem description, it needs to look for a better solution (i.e., a better negotiation strategy). This motivates the cooperative learning strategy of our methodology. Note that this cooperative learning is performed separately from the actual problem solving due to real-time constraints—a negotiation task needs immediate attention and cannot afford meddling with cooperative learning.

As previously mentioned, we have adhered to a cautious approach to cooperative learning: First, the agent evaluates the case to determine whether it is problematic. To designate a case as problematic, we use heuristics \textit{H4} and \textit{H5}: a (frequently used) case is problematic if it has a low success rate and a high incurrence rate. That means the case addresses an appropriate problem but does \textit{not} provide a satisfactory solution. Second, the agent only requests help from another agent that it thinks is good at a particular problem. The idea here is that we want to approach neighbors who have initiated successful negotiations with the current agent, with the hope that the agent may be able to learn how those neighbors have been able to be successful. Each agent keeps a profile of every neighbor that documents the negotiation relationships between the two agents [1]. Among the profiled parameters is \textit{helpRate}. This is the percentage of times that the agent has agreed to a request by a particular neighbor. The agent selects the neighbor with the highest \textit{helpRate} to ask for help, for example. This increases the chance that the neighbor may have a better solution than the agent’s. Finally, the agent adapts the foreign case before adopting it into its casebase. At the same time, the usage history parameters of the new case are reset.

3. EXPERIMENTS AND RESULTS

We conducted several sets of experiments and here are some key observations on a real-time multisensor target-tracking, noisy and dynamic, environment.

(1) Cooperative learning brings more utility and diversity per learning occurrence than individual learning,

(2) A small casebase learns more effectively in terms of utility and diversity, but not faster since our learning is problem-driven. A large casebase learns in a similar manner as an average casebase except when it is greater than the preset limit that triggers case replacement,

(3) The initial casebase affects the effectiveness of learning. Both types of learning bring more utility and diversity to an initial casebase previously grown within an agent than one previously grown with the other agents’ influence,

(4) Different environments affect agents’ learning behavior. Depending on the frequency of a task and its characteristics, an agent may rely more on individual learning or cooperative learning. For example, if a type of tasks (tracking) is time consuming and directional, then increasing its frequency actually weakens the potential benefit of individual learning and encourages the agent to perform more cooperative learning. The environments also impact the two initiating and responding roles differently, especially for negotiations associated with tough requirements (such as at least three members of a tracking coalition). Since an initiating agent has to shoulder the coalition management and decision making, it is able to learn more and more diverse and useful cases than a responding agent.

4. CONCLUSIONS

We have presented a multi-strategy learning methodology in a dynamic multiagent environment, where agents learn to negotiate better. For our experiments, we investigated the roles of individual and cooperative learning in various types of casebases, in terms of the initial size and the initial make-up of the cases, and the effects that different environments had on the learning behavior of agents.

5. ACKNOWLEDGMENTS

This work is partially supported by a grant from the DARPA ANTS project, subcontracted from the University of Kansas (Contract number: 26-0511-0026-001).

6. REFERENCES