Collaborative RPD Agents Assisting Decision Making in Active Decision Spaces

Guruprasad Airy, Po-Chun Chen, Xiaocong Fan, John Yen, David Hall
College of Information Sciences and Technology, The Pennsylvania State University, USA
{gma139, pzc123}@psu.edu, {zfan, jyen, dhall}@ist.psu.edu

Michael Brogan, Tim Huynh
Solers Inc, Arlington, Virginia, USA
{Michael.Brogan, Tim.Huynh}@solers.com

Abstract

One of the challenging issues in a distributed time sensitive information rich environment is how to assist the decision makers to make decisions quickly and effectively. This paper describes a decision model that has been developed for such situations and how using the R-CAST agent architecture can assist the humans in dealing with the information challenges. The example demonstrates how the system can be used in a military scenario.

1. Introduction

In information rich fast paced distributed environments, such as military scenarios, there are challenges on how to effectively make decisions. For instance, to enable successful processing of potential enemies, team members must effectively work together to quickly gather and make sense of information from multiple sources. However, teamwork in this area is often threatened by the fact that team members need to process large amounts of dynamically changing information under time pressure. Moreover, the information resident in such situations are typically distributed across people, objects, tools, and environments due to security concerns often associated with their roles and responsibilities. The quality and the timeliness of decision making can be significantly hampered by these unique and complex challenges. (Fan, et al 2005)

The focus of this research is to develop a generic decision model using collaborative decision making agents to support enhanced situational awareness and highly-focused, accelerated decision making. It has been found that the ability to anticipate upcoming decisions allows the decision makers to prepare for and execute those decisions much more effectively.

The system is being developed for use in military scenarios and the example describes a situation where a commander has to reach a decision on what asset to deploy to deal with an enemy target. Such a decision requires effective team collaborations to establish shared situation awareness and to rapidly link dynamic information from multiple sources.

To assist user in dealing with this, we use R-CAST, a collaborative agent architecture that is extended from the CAST architecture (Collaborative Agents for Simulating Teamwork) (Yen, et al, 2005). The remainder is organized as follows. Section 2 introduces the R-CAST architecture; Section 3 describes the decision model; Section 4 talks about how the decision model maps to the R-CAST agent architecture; Section 5 describes an example scenario and Section 6 summarizes the paper.

2. R-CAST Agent Technology

R-CAST is a framework for developing a team of intelligent agents that have the capability to make decisions based on the Recognition-Primed Decision (RPD) model. RPD is human-like decision making model developed by Klein. (Klein, 1993)

The R-CAST agent architecture, shown in Figure 1, comprises a knowledge base, a communication manager, a collaborative RPD module, a process manager and a recommendation module. The architecture is domain independent and can be configured for use in any domain as long as the relevant knowledge and domain-specific actions are described appropriately.
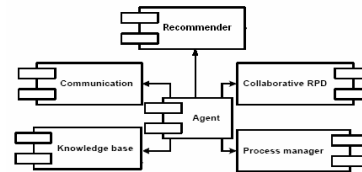

Figure 1. R-CAST Architecture

The *knowledge base (KB)* contains information regarding the external world and the other agents. It has a forward-chaining rule based system that can reason about missing information and proactively find ways to satisfy the information requirements. The *Communication Manager* handles inter-agent communication and supports different protocols. The *Process Manager* handles the execution of various plans. Multiple plans can be concurrently run in an agent. The *Collaborative RPD module* implements the decision making process and

allows team members (humans and other agents) to collaborate during the decision making. The *Recommendation module* handles the cost benefit analysis of a set of options (usually course of actions) by using criteria for evaluating each option. A score is generated for each option based on the criteria specified and it can be presented to a user for choosing an option or can be automatically chosen based on the score.

3. ADS Vision

Active Decision Spaces is a technology being developed for supporting decision making in information rich fast paced environments. The idea is to enable the users to anticipate, support, make and execute operational decisions.

An Active Decision Space consists of a template for describing how a decision is made. This template is called a Decision Process Model (DPM). It is described as a set of inter-connected decision processes where each individual decision process describes a set of decision nodes, information processes and their relationships. A decision node can be another decision process (DP) or a decision element (DE). The output of a decision process is a high level decision that is generated based on the outputs from its children nodes. A decision element is the primary component that models an individual decision which cannot be decomposed further. A decision element can either be used to assist a human in making a decision or can be automated to make a decision on its own. The output of each DE is a decision that affects the decision of the subsequent Decision Elements. The information processes (IPs) basically pass on the information gathered from sensors to the decision nodes. Their description is beyond the scope of this paper.
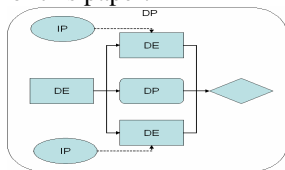

Figure 2. Example decision process

Figure 2 illustrates a decision process consisting of three Decision Elements, a sub Decision Process and two Information Processes. The diamond represents the final decision recommendation for the decision process.

4. Realizing ADS Agents Using R-CAST

In this section, we describe how the Decision Process Model (DPM) is realized by a team of collaborative R-CAST agents and the three key design issues of our implementation in detail followed by a summary.

4.1. From DPM to R-CAST Agents

Each decision node in the DPM is handled by a unique R-CAST agent. The agents representing decision elements are called Decision Element Agents (DEAs) and the agents which represent decision processes are called Decision Process Agents (DPAs). Each DPA controls and coordinates its children DEA/DPAs. Note that a sub DPA has its own children. The mapping preserves the hierarchical structure of a DPM.

Additionally, in a DPM there are information needs and information flow specified which can be realized by defining information sources in the knowledge bases of R-CAST agents, such that an agent can figure out where to gather its missing information.

In dynamic environments when we may have multiple predefined DPMs, we use the RPD module with necessary experiences to take care of situation recognition and trigger a corresponding DPM.

In the next three subsections, we will show more details regarding three key design issues of our implementation, which are decision process modeling, the subscription-based information exchange, and RPD-based DPM triggering.

4.2. Process modeling

A decision process in a DPM consists of a set of decision nodes as well as their relationships to construct the flow of the decision procedure. To realize a process model, firstly we build a Decision Process Agent (DPA) with knowledge of the process flow to manage a decision process. The process knowledge is represented in R-CAST's process definition language by mapping each of its children decision nodes to a "plan" and constructing the flow relationships by setting their preconditions as the accomplishment of their precedent nodes. Figure 3 shows an example of a decision process with four children nodes.
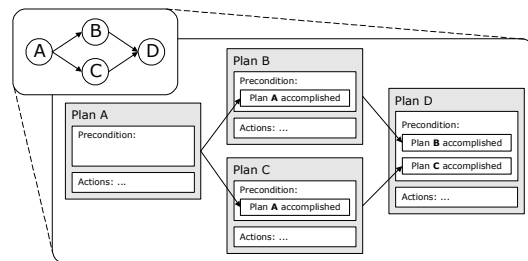

Figure 3. Execution of a decision process

Secondly, we create a DEA or DPA for each of these plans. When a plan is launched, it will assign a task to its corresponding DEA/DPA. Lastly, to monitor all its children's states, we let the parent DPA subscribe to the status reports for automatic updates for all of its children.

4.3. Information exchange using subscription

For better handling of information requirements, we implemented a subscription mechanism for information exchange. It can be applied on two different aspects. One is for information acquisition and delivery, the other is the assistance of managerial operations. As for the first aspect, when making decisions, we may have information needs for a specific type of information; if we can make use of subscription, the knowledge can be kept up to date automatically. As for the managerial aspect, as mentioned previously, a DPA can use subscription to monitor the states of its children DEA/DPAs. Using subscription can reduce the cost of requesting since we only need to make requests once at the very beginning and then the updates will be delivered automatically.
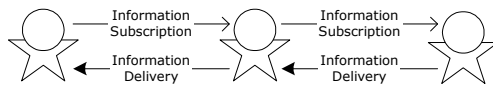

Figure 4. Chained information subscriptions

Besides, information subscription can be chained to provide extra benefits (Sun, 2005). By constructing an information supply chain, an agent can gather inaccessible information indirectly through another agent. This also helps reducing the communication traffic. Figure 4 illustrates the concept of chained information subscription.

4.4. RPD-based DPM triggering

In a dynamic environment, before running a specific DPM, we need to recognize the current situation and pick up an appropriate one to launch. Our approach is to use RPD to take care of this kind of triggering.
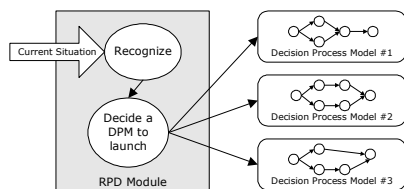

Figure 5. RPD-based DPM triggering

We can deploy multiple groups of agents which implement different kinds of DPMs as described before. Then we deploy one additional agent with RPD module as well as relevant experiences to decide which group of agents to launch. Figure 5 demonstrates the idea of RPD-based DPM triggering.

4.5. Summary

R-CAST provides a domain-independent decision making model by its RPD module. By implementing a

DPM, which explicitly defines the decision process and information needs, we can organize a group of R-CAST agents as a team to perform domain-specific decision making.

Domain-independent decision making using RPD is based on experiences, which is flexible but hard to capture the details of a complex decision process.

Domain-specific decision making using DPM is based on user-predefined models, which explicitly describes everything needed for the decision in a specific domain. Moreover, we can embed various elements in the decision process to assist our decision making, such as human interventions, cost-benefit analysis, or even RPD instances. However, once a DPM is determined, it is hard to be modified at run-time.

In summary, the domain-specific decision making using DPM is more flexible in design but could be less adaptable in run-time.

5. Example

This section describes an example scenario in which the system was tested. It illustrates only one decision process out of an entire military Decision Process Model that was developed and tested.

5.1. Description of the scenario

The system was primarily developed for use in military scenarios where information is distributed and decisions need to be made under time pressure. In this example we consider a scenario in which a commander has to make a decision on which asset to assign to deal with an enemy target that has been spotted.
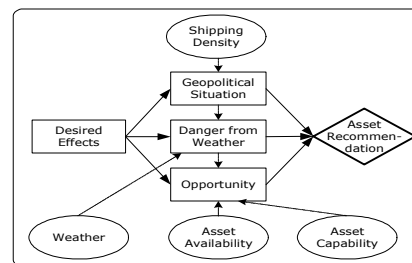

Figure 6. The example scenario

The decision process consists of four decision elements and four information processes as illustrated in Figure 6. A human user selects the desired effect which could be either to chase away the enemy or attack the enemy. The output of this decision element is passed onto the subsequent decision elements which are automated to make decisions based on their respective inputs from information processes. The geopolitical cost agent gathers information about friendly assets or forces in the area and determines how they may be affected in this situation. The

weather agent determines how the weather might play a role in choosing an asset. The opportunity agent gathers information about availability of assets and their capabilities and determines which assets are more suitable in the current situation. The decision information from these three agents is passed onto the parent decision process that generates a ranked list of assets that can be used for achieving the desired effect and lets the human user make a decision.
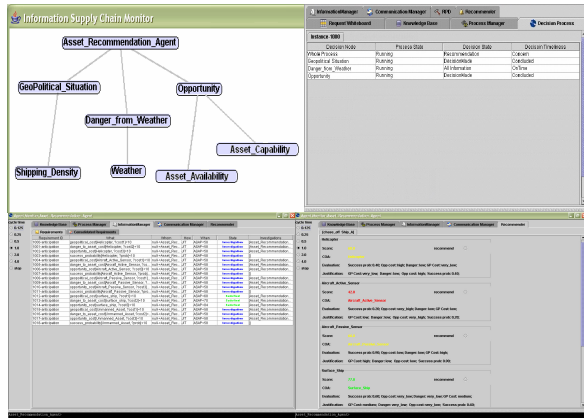

Figure 7-a, b, c, d (clockwise). Screenshots

Figure 7(a) shows the dependency links between the various agents. It can be seen that the dependency graph is the same as the decision process specified in the example scenario in Figure 6. Figure 7(b) shows the decision process monitor that maintains the status and controls and co-ordinates the various decision nodes under a decision process. In addition to maintaining the Process State (whether the agent is running or not) the system also maintains the Decision State (how much information has been gathered) and the Decision Timeliness (whether the decision is on time or late). Figure 7(c) shows the recommendation output of an agent which indicates the various options along with a score. The user can select one or more options to make a decision. Figure 7(d) shows the information requirements of an agent, specifying not only what information is missing but also how this requirement can be satisfied.

6. Conclusion

This paper describes an initial version of the system in which a team of collaborative decision making agents that support proactive information exchange are used to assist humans in making decisions in time critical information rich scenarios. Using R-CAST, ADS allows military commanders to increasingly and accurately process large number of targets and to make timely actionable

decisions. From the experiments, it is clear that the system performs as expected.

In future versions, we plan to add learning capabilities to the agents and to conduct detailed experiments to evaluate the performance gains of using such a system. The final system is aimed to allow experimentation and demonstration of advanced tactical information exchange, reduced cognitive load, enhanced situation awareness, and positive human-agent collaboration.

References

[1] X. Fan, S. Sun, and J. Yen. On Shared Situation Awareness for Supporting Human Decision-Making Teams, In *Proceedings of 2005 AAAI Spring Symposium on AI Technologies for Homeland Security*, pp. 17-24, Stanford, CA, Mar. 2005.

[2] X. Fan, S. Sun, M. McNeese, and J. Yen. Extending recognition-primed decision model for human-agent collaboration. In *Proceedings of the Fourth International Joint Conference on AAMAS*, pages 945-952, The Netherlands, July 2005.

[3] X. Fan, J. Yen, and R. A. Volz. A theoretical framework on proactive information exchange in agent teamwork. *Artificial Intelligence Journal*, 169(1):23-97, 2005.

[4] X. Fan, B. Sun, S. Sun, M. McNeese, and J. Yen. RPD-enabled agents teaming with humans for multi-context decision making. In *Proceedings of the Fifth International Joint Conference on AAMAS*, pages 34-41, Hakodate, Hokkaido, Japan, May 2006.

[5] G.A. Klein, A recognition-primed decision making model of rapid decision making. In G. Klien, J. Orasanu, R. Calderwood and C., Zsambok Eds., *Decision Making In Action: Models and Methods*, pages 138-147, 1993

[6] G. A. Klein. The recognition-primed decision model. In *Sources of Power: How People Make Decisions*, chapter 3, pages 15-30. The MIT Press, 1998.

[7] E. Norling, L. Sonenberg, and R. Ronnquist, Enhancing multi-agent based simulation with humanlike decision making strategies. In *Proceedings of the second International workshop on Multi-Agent Based Simulation* (Moss, S. and Davidsson, P. Eds.), page 214-228, 2000

[8] S. Sun and J. Yen, Information Supply Chain: A Unified Framework for Information-Sharing. In *Proceedings of IEEE International Conference on Intelligence and Security Informatics* (IEEE ISI-2005). Atlanta, GA, May 19-20, 2005.

[9] J. Yen, X. Fan, S. Sun, T. Hanratty, and J. Dumer. Agents with shared mental models for enhancing team decision-makings. *Journal of Decision Support Systems*, 41(3):634-653, 2006