Negotiating Agents for Supply Chain Management

Ye Chen, Yun Peng, Tim Finin, Yannis Labrou, and Scott Cost
Computer Science and Electrical Engineering
University of Maryland Baltimore County
Baltimore, MD 21250
{yechen, ypeng, finin, jklabrou, rcost1}@cs.umbc.edu

1. Introduction

A supply chain is a network of suppliers, factories, warehouses, distribution centers and retailers, through which raw materials are acquired, transformed, produced and delivered to the customer. A supply chain management system (SCMS) manages the cooperation of these system components. In the computational world, roles of individual entities in a supply chain can be implemented as distinct agents. The functions and procedures of a company in the real market are complicated and include information collection, policy making and actions. Therefor, it is impossible to describe software agent behaviors for an uncertain e-commerce environment such as supply chain management in the traditional single threaded model. To solve the problem, we introduce the concept of negotiation into software agent design for supply chain management and present our ideas to solve the problem of communication and decision-making for negotiating agents.

When the software agents enter into the market, a SCMS is thereby transformed into a multi-agent system. Since software agents might belong to different companies and are self-interested, a pure scheduling scheme can not help. In addition, software agents tend to cooperate in a relatively dynamic way. To address these problems, we propose a MAS framework of negotiating agents for supply chain management. In our framework, there is no preset relationship between agents. When an order comes, a virtual supply chain may emerge through negotiation processes. The components of the chain may change according to the external situation even after the order has been accepted.

2. Negotiating agents and system framework

In this section we give the definition of negotiating agents and describe the framework of the Multi-agent System (MAS) for supply chain management [1]. Negotiation is a very broad term that as employed generally covers a range of problem solving activities involving two or more parties, who interact to arrange for joint activities or the settlement of a mutual concern, e.g., to exchange goods and services. We identify four general issues, which need to be addressed for software agents in a multi-agents system for supply chain management.

(1) **Communication.** In order to support meaningful communication among the negotiating parties we need to have a common language for expressing primitive communicative acts that make up a negotiation (e.g., a call for proposals, a rejection of a proposal, etc.) as well as a way to specify different protocols that can be used (e.g., English auction, contract net, bargaining, etc.).

(2) **Representation.** Most negotiation is about complex objects (physical or abstract) that may require the support of a sophisticated representation scheme. Examples can range from orders for goods to contracts for services

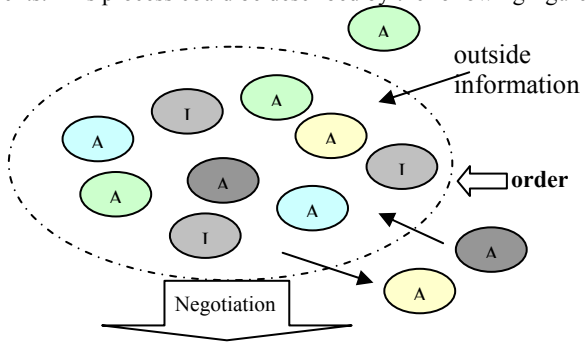
(3) **Problem solving.** Many aspects of negotiation can be modeled as an exercise in distributed constraint solving. There is a large body of work on algorithms and techniques for constraint solving that can be applied to negotiation problems.

(4) **Human interaction.** Negotiation has to be carried out in the context of existing human organizations. Whatever automated negotiation processes have to be coupled with humans in appropriate ways, either for authorization and modification or as part of a larger workflow environment.

The behaviors of a software agent, which correspond to the above four aspects of negotiation, are called *negotiating actions*. A software agent is a *negotiating agent* if it can at least take communicating and problem solving actions in a specific domain.

In our framework, the MAS for supply chain management consists of two types of negotiating agents – functional agents and information agents. *Functional agents* implement some supply chain management functionality. These agents are usually owned by different companies and are therefore assumed to be self-interested and thus free to join, remain in or leave the supply chain system. *Information agents* are predefined in the system and help functional agents to find potential negotiation partners or provides other altruistic service such as accepting the registration from a functional agent. All of the negotiating agents have some understanding of system ontology and use a certain Agent Communication Language (ACL) to make conversation. The interaction between them, the *negotiation process*, is modeled as a process of cooperatively assigning values to a set of variables.

In the framework, There are no centralized super-agents or distributed mediators to handle the agent cooperation. All these activities occur through negotiation processes, regardless of whether two sides are involved in bargaining for some goods intentionally or de-committing a contract caused by the outside events. This process could be described by the following figure:



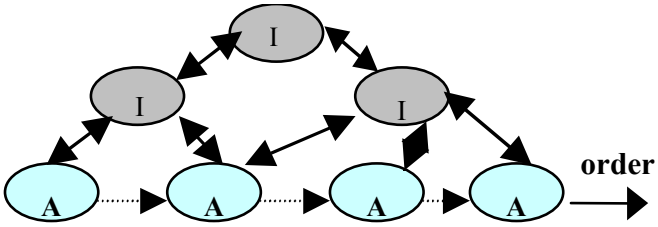


Fig 1 A negotiation-based MAS for supply chain management (A: functional agents; I: information agents)

3. Negotiating agents communication

In a supply chain negotiation process, negotiating agents use an Agent Communication Language (ACL) [5] to bargain with each other. The table below presents the peformatives designed for the negotiating agents based on FIPA ACL [4]. A negotiation protocol, formally described using Color Petri Net (CPN) is also given.

Accept-proposal	the action of accepting a previously submitted proposal to perform an action
Acknowledge	the action of acknowledging the withdraw of a proposal
CFP	the action of calling for proposals to perform a given action
CFMP	the action of calling of a modified proposal according to the one the agent previously receives
Reject-proposal	the action of rejecting a proposal to perform some action during a negotiation (with reasons).
Proposal	the action of submitting a proposal to perform a certain action, given certain preconditions
Terminate	the action to terminate the negotiation process
Withdraw	the action of withdrawing an outstanding proposal

Table 1 Pair-wise negotiation performatives

The above performatives can be used to construct negotiation protocol between negotiating agents. The following figure gives an example of describing a pair-wise agent negotiation protocol for two negotiating agents in Color Petri Net (CPN) [1,2].

```
Color PR = product AGENT* AGENT;
fun Color AGENT = index a with 1..2;
diff (x <> y);
Color MES = subset PR by diff declare ms;
Color PERFORMATIVE = accept-proposal | CFP | proposal | reject-
   proposal | terminate;
Color BARGAIN = String;
Color CONTENT = product PERFORMATIVE * BARGAIN;
Color E with e;
fun (s,r,c) = mult' PR * CONTENT(1`s, AGENT-1`s, c);
var s,r : AGENT;
var c : CONTENT;
```

4. Agent negotiation as DCS

A distributed constraint satisfaction (DCS) problem is a constraint satisfaction problem in which variables and constraints are distributed among multiple automated agents. In our model,

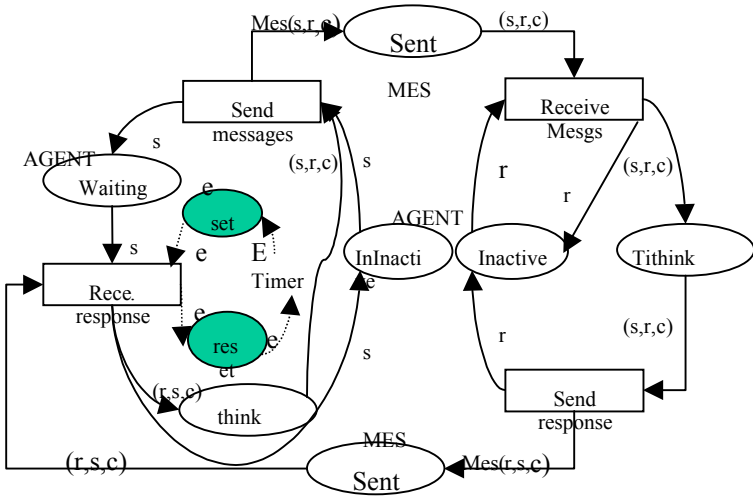


Fig 2 Pair-wise negotiation in a MAS with two functional agents

the order the system receives is considered as the solution needed to be provided. Each negotiating agent contributes its own partial solution. Many techniques and algorithms could be used to attack this problem. We prefer to introduce probability theories into the negotiating agent problem solving process. The detailed research is in the progress.

5. Conclusion

In this abstract we have described an approach to modeling the supply chain management problem in the real business environment using software agents. We use the concept of negotiating agent to model the self-interested entities in the market place. The system framework we designed allows negotiating agents join, stay or leave the system freely. The basic ideas and methods to attack the aspects of negotiating agent negotiation behaviors including the communication and problem solving parts have been given and studied. We are currently working with a prototype system implemented with the Jackal [3] toolkit for building multi-agent systems.

References

[1] Y. Chen Y. Peng, T. Finin, Y. Labrou, R. Cost, B. Chu, R. Sun and R. Willhelm, "A Negotiation-based Multi-agent System for Supply Chain Management'', *ACM Autonomous Agents workshop on Agent-based Decision-support for Managing the Internet-enabled Supply-chain*, Seattle, May, 1999.

[2] R. Cost, Y. Chen, T. Finin, Y. Labrou and Y. Peng, "Modeling Agent Conversations with Colored Petri Net," *ACM Autonomous Agents workshop on Specifying and Implementing Conversation Policies*, Seattle, May, 1999.

[3] R. Cost, T. Finin, Y. Labrou, X. Luan, Y. Peng, I. Soboroff, and J. Mayfield, ``Jackal: a Java-based Tool for Agent Development", poster paper, ACM Autonomous Agents Conference, 1999.

[4] *FIPA97 Specification, Part 2: Agent Communication Language*, Foundation for Intelligent Physical Agents, October, 1997.

[5] Y. Labrou, T. Finin and Y. Peng, "The current landscape of Agent Communication Languages", *Intelligent Systems*, volume 14, number 2, March/April 1999, IEEE Computer Society.

[6] Y. Peng, T. Finin, Y. Labrou, B. Chu, J. Long, W. Tolone, and A. Boughannam, ``A Multi-Agent System for Enterprise Integration'', *Journal of Applied Artificial Intelligence*, 1998.