IDEA GROUP PUBLISHING
701 E. Chocolate Avenue, Hershey PA 17033-1117, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com
ITB8016

Chapter VIII

Design Agents with Negotiation Capabilities

Jana Dospisil and Liz Kendall
Monash University, Australia

INTRODUCTION

Agents are viewed as the next significant software abstraction, and it is expected they will become as ubiquitous as graphical user interfaces are today. Multi-agent systems have a key capability to reallocate tasks among their members, and this may result in significant savings and improvements in many domains, such as resource allocation, scheduling, e-commerce, etc. In the near future, agents will roam the Internet, selling and buying information and services. These agents will evolve from their present-day form—simple carriers of transactions—to efficient decision makers. It is envisaged that the decision-making processes and interactions between agents will be very fast (Kephart, 1998).

The importance of automated negotiation systems is increasing with the emergence of new technologies supporting faster reasoning engines and mobile code. A central part of agent systems is a sophisticated *reasoning engine* that enables the agents to reallocate their tasks, optimize outcomes, and negotiate with other agents. The *negotiation strategy* used by the reasoning engine also requires high-level interagent communication protocols and suitable collaboration strategies. Both of these subsystems—a *reasoning engine* and a *collaboration strategy*—typically result in complicated agent designs and implementations that are difficult to maintain.

Diversity in reasoning and strategies has led to "homegrown" agent development done independently from scratch. This has led to the following problems (Bradshaw, 1997):

- **Lack of an agreed definition:** Agents built by different teams have different capabilities.
- **Duplication of effort:** There has been little reuse of agent architectures, designs, or components.
- **Inability to satisfy industrial strength requirements:** Agents must integrate with existing software and computer infrastructure. They must also address security and scaling concerns.

This chapter explores current concepts and directions in negotiation and collaboration strategies in Multi-agent systems. The focus is on coupling the theoretical underpinning of negotiation processes with trends and directions in software engineering and design. In particular, the implications of the theoretical work on design are discussed.

In this chapter we provide an overview of different reasoning and negotiation strategies

among agents and then we move to issues relevant to the architecture, design, and implementation of Multi-agent systems based on constraint technology and software patterns. In the second section we discuss the fundamentals of negotiation strategies and protocols based on the following theories:

- Game theoretical models of bargaining and negotiation (analytical strategies),
- Evolutionary (machine learning) computational approaches,
- Constraint technology.

Particular attention is paid to constraint technology, scheduling, and resource-allocation strategies. Finally, we present two software engineering approaches for building agent systems:

1) Plug-in components for agents, and
2) Software patterns.

BACKGROUND

Selected Negotiation and Reasoning Techniques

It is necessary to coordinate the activities of a set of autonomous agents, and some of them could be mobile agents, while others are static intelligent agents. We usually aim at decentralized coordination which produces the desired outcomes with minimal communication. Many different types of *contract protocols* (cluster, swaps, and Multi-agent, as examples) and *negotiation strategies* are used. The evaluation of outcomes is often based on marginal cost (Sandholm, 1993) or game theory payoffs (Mass-Colell, 1995). Agents based on constraint technology use complex search algorithms to solve optimization problems arising from the agents' interactions. In particular, coordination and negotiation strategies in the presence of incomplete knowledge are good candidates for constraint-based implementations.

The focus of any negotiation strategy is to maximize outcomes within the rational boundaries of the environment. The classification of negotiation strategies is not an easy task since a negotiation strategy can be realized by any algorithm which evaluates outcomes, computes appropriate actions, and follows the information exchange protocol.

Negotiation is a search process. The participants jointly search a multidimensional space (e.g., quantity, price, and delivery) in an attempt to find a single point in the space at which they reach mutual agreement and meet their objectives. For many-to-many coupling or interaction between participants, the market mechanism is used, and for one-to-many negotiation, auctions are more appropriate. The market mechanism often suffers from an inability to scale down efficiently (Osborne, 1990) to smaller numbers of participants. On the other hand, one-to-many interactions are influenced by strategic considerations and involve interactive bargaining where agents search for *Pareto efficient* agreement.

Three distinct techniques are reviewed in the following subsections:

i) Analytical (game theory-based approach),
ii) Evolutionary approaches (genetic algorithms approach), and
iii) Constraint technology and constraint agents.

Analytical Approach (Game Theory)

The principles of bargaining and negotiation strategies in Multi-agent systems have