Designing Agents with Negotiation Capabilities

Jana Polgar
Monash University, Australia

SOFTWARE AGENTS TODAY

Agents are viewed as the next significant software abstraction, and it is expected they will become as ubiquitous as graphical user interfaces are today. Agents are specialized programs designed to provide services to their users. Multiagent systems have a key capability to reallocate tasks among the members, which may result in significant savings and improvements in many domains, such as resource allocation, scheduling, e-commerce, and so forth. 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 inter-agent communication protocols, and suitable collaboration strategies. Both of these sub-systems – a *reasoning engine* and a *negotiation strategy* - typically result in complicated agent designs and implementations that are difficult to maintain.

Activities of a set of *autonomous agents have to be coordinated*. Some 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 multiagent, 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' interaction. In particular, coordination and negotiation strategies in the presence of incomplete knowledge are good candidates for constraint-based implementations.

SELECTED NEGOTIATION AND REASONING TECHNIQUES

Negotiation space is determined by two components: *negotiation protocol* and *negotiation strategy*. The *negotiation protocol* defines the rules of behavior between the participants in terms of interactions, deals, bidding rules, temporal constraints and offers, as components of the protocol. Two agents must first agree on the negotiation protocol before any interaction starts.

The *negotiation strategy* is a specification of the sequence of actions the agent intends to make during the negotiation. Strategies should be compatible with the negotiation protocol. 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 the negotiation strategy can be realized by any algorithm capable of evaluating outcomes, computing appropriate actions, and following the information exchange protocol.

The *negotiation mechanism* is the actual implementation of negotiation strategy and negotiation protocol. This field is evolving fast, with emergence of new agent platforms, wireless encounters and extended mobility.

Negotiation is a search process. The participants jointly search a multi-dimensional 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. The *market mechanism* is used for many-to-many coupling or interactions between participants. *Auctions* are more appropriate for one-to-many negotiations. The market mechanism often suffers from inability to efficiently scale down (Osborne, 1990) to smaller numbers of participants. On the other hand, one-to-many interactions are influenced by strategic considerations and involve integrative bargaining, where agents search for *Paretto efficient* agreements (tradeoffs).

NEGOTIATION STRATEGIES

Analytical Approach (Game Theory)

The principles of bargaining and negotiation strategies in multiagent systems have attracted economists. Early foundations and mathematical models were investigated by Nash (1950), and the field is still very active. The *game theory* is a collection of analytical tools designed to understand and describe bargaining and interaction between decision makers. Game theory uses mathematical models to formally express real-life strategies (Fudenberg, 1991; Osborne, 1994).

The high-level abstraction allows the model to be applied to a variety of situations. The model places no restrictions on the set of actions available to the player. With regard to mathematical models, there already exist many sophisticated and elaborated strategies for specific negotiation problems. The Contract Net Protocol (CNP) (Sandholm, 1993; Smith, 1980) represents the model of decentralized task allocation where agents locally calculate their marginal costs for performing sets of tasks. The pricing mechanism in Sandholm (1993) generalizes the CNP to work for both cooperative and competitive agents. In Zeng (1996), bilateral negotiation based on the Bayesian method is presented. It demonstrates the static nature of the model. The learning effect is achieved by using dynamic updates of a knowledge base, which is consulted during the negotiation process.

Most of the studies assume perfect rationality (flawless deduction, marginal costs are computed exactly, immediately and without computational cost), and the infinite horizon of strategic bargaining. These are not realistic assumptions. More advanced studies deal with coalition formation and negotiation strategies in the environment of multiple self-interested or cooperative agents with bounded rationality (Sandholm, 1993) and bargaining with deadlines.

Analytical approach has the advantage of stable and reliable behavior. The main disadvantage is the static nature of the model, resulting in potential predictability of the outcomes. The other problems are associated with the notion of perfect rationality.

Contracts in automated negotiations consisting of self-interested agents are typically designed as binding (impossible to breach). In cooperative distributed problem solving, commitments are often allowed to be broken based on some local reasoning. Frequently, the protocols use continuous levels of commitment based on a monetary penalty method (Sandholm, 1993). Unfortunately, the inflexible nature of these protocols restricts an agent's actions when the situation becomes unfavorable. The models that incorporate the possibility of decommitting from a contract with or without reprisals (Sen, 1994; Smith, 1980) can accommodate some changes in the environment and improve an agent's status. However, all of these protocols are somewhat restricting with respect to evolving, dynamic situations.

Evolutionary Strategies

With *evolutionary strategies*, the data used as the basis for negotiation, as well as the algorithm operating on the data, evolve. This approach provides more efficient learning, supports the dynamics of the environment, and is adaptable. However, only a few implementations have been attempted, and these have been of only simple negotiation strategies (Aridor, 1998). *Genetic algorithms* are probably the most common techniques inspired by evolution, in particular by the concepts of natural selection and variation. The basic genetic algorithm is derived from the hypothesis that the candidate solutions to the problem are encoded into "chromosomes". Chromosomes represent a solution or instance of the problem hand encoded into a binary string. The algorithm then operates on this binary string. It begins with a randomly generated set of candidate solutions. The set of candidate solutions is generated as a random string of ones and zeroes. Each chromosome is evaluated and the fitness of the chromosome could be the value of the objective function (or the utility if we want to maximize the outcome). A new population is created by selecting individuals to become parents. A thorough description of the genetic algorithm approach can be found in Goldberg (1989).

A very large amount of research has been carried out in the application of evolutionary algorithms to situations that require decisions. Examples include coalition games, exchange economies, and double auctions. This approach was inspired by the concept of variation and natural selection. The intelligent agents are modeled using classifier systems to select decisions. Although the recent research shows that multiagent systems of classifiers are capable of learning how to play *Nash-Markov equilibrium*, the current limitations of computational resources and the instability of "home-grown" implementations significantly constrain the nature of the strategies. The important question is what design and implementation techniques should be used to ease this conflict and to provide the resources required for genetic learning to operate in an unrestricted way. It is believed that the ability of agents to learn simple games would be beneficial to electronic commerce.

Constraint Agents

The potential of constraint-based agents is still to be fully realized and appreciated. One of the possible frameworks for constraint-based agents is outlined in Nareyek (1998). This framework considers agents as a means for simplifying distributed problem solving. An agent's behavior and the quality of solutions depend on the underlying action-task planning system. The recent results with some constraint planners and constraint satisfaction problems (CSP) indicate the potential advantages of this approach.

Related Content