Uncertainty Management Framework for Multi-Agent System

Wei Wu, Edidiong Ekaette, Behrouz Homayoun Far
Department of Electrical and Computer Engineering
University of Calgary
2500 University Drive, NW, Calgary, Alberta, Canada T2N 1N4
Email: {wuwe, ekaette, far @enel.ucalgary.ca}

Abstract

Software Agents interaction involves uncertainty. In order to model and formalize agents' interaction, three properties: public, private and protected are adopted to precisely represent the *belief-desire-intension* (BDI) agent, and agent's relationships have been interpreted in terms of cooperation, coordination and competition based on BDI agent model. An agent's uncertainty management framework is then proposed with respect to some decision support tools, this framework could ease the analysis of multiagent systems in a more formal way and help design efficient reasoning process for software agents.

Key words:

Software Agent, Multi-Agent System, Agent Interaction, Uncertainty Management.

1. Introduction

Multi-agent system (MAS) paradigm provides a new and powerful approach for analyzing, designing and implementing complex, heterogeneous, open, distributed, and networked software systems. A central concern in MAS design is to specify how software agents interact with each other. That is, how agents can communicate, compete with each other, or share the burden of tasks efficiently, without which, software agents will unintentionally waste their efforts and resources, fail to accomplish their individual and collective objectives.

However, agent-based development, especially in the area of agent-based decision making process appears to be rather disordered and incoherent. In this paper we concentrate on interpretation of software agent's decision making by analyzing the agents' interaction with respect to the *belief-desire-intension (BDI)* agent model [3] and application of uncertainty management techniques in agents' decision making process in MAS. In Section 2, BDI agent model [3] will be reviewed and a new attributes will be added, based on which the interactions among agents will be semi-formally modeled and represented in Section 3. Some decision support techniques that can potentially be adopted in agents' uncertainty management are reviewed in Section 4 and an uncertainty management framework for agent based systems is presented in Section 5 followed by conclusions in Section 6.

2. Extended BDI Agent

One of the popular approaches for developing agent-based systems, views the constituent agents as rational agents with the *Belief*, *Desire* and *Intention* (BDI) mental attitudes, representing the information, motivational and deliberate states of the agent, respectively [4]. In order to adopt BDI agent to represent the relationships between agents, we include a new mental attitude – *Knowledge Base*. This mental attitude that could be considered as the fourth attribute of an agent reflects "how to do" something. It completes the BDI cycle. The *Knowledge Base* is necessary because it determines the behavior and is crucial for a software agent to achieve its goals.

The states of a BDI agent at any given moment is a quadruple (B, D, I, KB), where B $\subseteq$ Belief, D $\subseteq$ Desire, I $\subseteq$ Intension [3] and KB$\subseteq$ Knowledge Base.

- *Belief* represents an agent's information of the state of nature of the environment, which is a recursively updating attributes determined by the following function,

  $$\wp(Bel) \times P \rightarrow \wp(Bel)$$

  An agent's new set of beliefs is a mapping on the basis of the current percept and current beliefs [3].
- *Desire* is a set of options that are available for an agent to achieve its intention, it is generated from a set of beliefs and a set of intentions, this generation process is depicted as follows,

  $$\wp(Bel) \times \wp(Int) \rightarrow \wp(Des)$$
- *Intention* is actually the goal that an agent wants to achieve, an agent's intention will give a guideline of what to do.
- *Knowledge Base* is a centralized repository for an agent's information, it is a database of related information about a particular subject and actually represents the concept of "how to do" of an agent.

These four attributes *Belief*, *Desire*, *Intention* and *Knowledge Base* could represent an agent's internal reasoning process of deciding which action to perform in the furtherance of their goals, but they are not sufficient if we want to model an agent's relationship with its counterparts or opponents. When an agent engages with the others, the status of the four attributes of self and the counterparts determine the way the interaction should take place. We suggest that each attribute can have three states: *public*, *private* and *protected*. *Public* means that this attribute is visible from the others, *private* means that this attribute is masked from others, while the meaning of *protected*, which is defined differently from Object Oriented viewpoint, is that the probability distribution of the attribute is revealed to others.

3. Modeling Agents' Interaction and Decision Making

Three types of agents' interactions, i.e. *cooperation*, *coordination* and *competition* have already been defined in [2]. We try to interpret them based on the extended BDI agent model which could be represented by four attributes and three properties and is depicted in Figure 1.

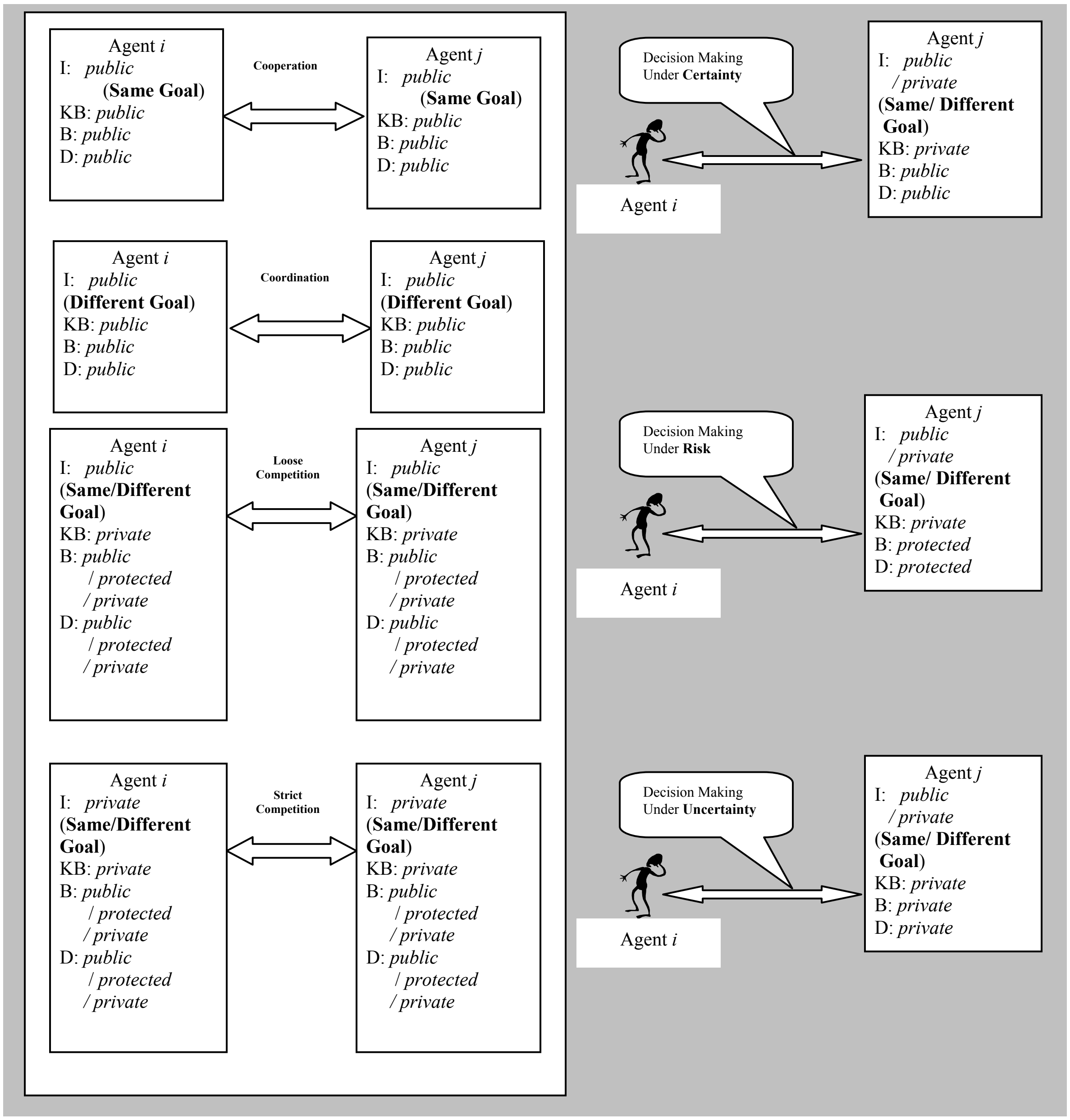

Fig. 1. Modeling agents' interaction using BDI & three types of decision making

Cooperation: Cooperation is revealing an agent's goal and knowledge behind it to the other parties [2]. According to the definition of extended BDI agent in Section 2, all four attributes of an agent are revealing to the others in its environment.

Coordination: Coordination is revealing an agent's goal and the knowledge behind it to the other parties [2]. In coordination, agents have separate goals. In other words, an agent's Intention, Desire, Belief and Knowledge Base are all public.

Competition:

Loose Competition: Loose competition is revealing only an agent's goal but masking the knowledge behind it to the other parties [2]. Alternatively speaking, only an agent's Intention is public, its Knowledge Base is private because it want to hind its own knowledge, Belief and Desire could be any of these three properties: public, private or protected depends on how much uncertainty exists in its interaction, no matter this agent has the same or different goal with its opponents.

Strict Competition: Strict competition is neither revealing an agent's goal nor the knowledge behind it to the other parties [2]. Thus, both an agent's Intention and Knowledge Base are private, and its Belief and Desire could be public, private or protected, it may have the same or different goal with its opponent.

Identification of these three types of relationship is necessary for an agent to perform well and interact with others to solve problem efficiently. In a MAS environment, when an agent is going to take an action, the first thing it needs to be sure of is its relationship with every other agent in the same system: it may be at cooperation or ordination relationship with some agents, in which, knowledge sharing and symbol level communications are possible [2]. In such cases, the generation of an agent's plan of action is straightforward and uncertain decision making techniques are not required. On the other hand, the agent may be at competition relationship with some others, either loose competition or strict competition, in which an agent may try to get as much amount of its opponents' information as possible and hide its own, different amount of information an agent possesses about its opponent reflects different levels of uncertainty when it makes its decision which has been identified in [2]. The interpretation of these three cases based on extended BDI agent model is illustrated in Fig. 1, we assume that agent *i* is competing with agent *j*,

- *Decision Making under Certainty*
  Agent *i* knows exactly the state of agent *j*. Both the *Belief* and *Desire* of agent *j*'s must be declared as public, means that they are accessible or readable by agent *i*.
- *Decision Making under Risk*
  Agent *i* knows only the probability distribution not the exact state of agent *j*. In this case, both the *Belief* and *Desire* of agent *j*'s are declared as protected, in other words, agent *i* can only access the probability distribution of agent *j*.
- *Decision Making under Uncertainty*
  Agent *i* knows nothing about agent *j*, neither the probability distribution nor its exact state. Agent *j* tries to mask its own information, makes both its *Belief* and *Desire* private.

The data an agent gathers from its opponents is mostly uncertain data. Making decisions based on uncertain data needs decision support techniques and theories that handle uncertainty. We review some of those techniques in the next section and try to develop a decision making framework for agent ineraction.

4. Review of Decision Support Techniques

There are a number of decision making theories and techniques that could deal with uncertainties in agents' interactions. They can be divided in to two groups: *decision support* and *computational techniques* according to their different approaches to handle uncertainty problems.

4.1 Decision Support Theories and Techniques

The decision support theories and techniques we talk about in this paper is the class of theories and techniques that help the decision maker implement the whole decision making support process, in particular, they are used to model the controlled objects and the environments, formalize the situation, generate several alternative, suggest solutions to the decision making or give the most rational one among these alternative solutions according to each specific theory. We list the most commonly used decision making theories and techniques below.

Decision theory
Decision theory is a means of analyzing which of series of options should be taken when it is uncertain exactly what the result of taking the option will be [6]. Decision theory helps the decision maker to choose the "best" decision option. Best here, which is similar to the meaning of rational in game theory, refers to one that could maximize the expected utilities. Consequences, payoffs and probabilities are concepts used in decision theory.

Game theory
Game theory relates closely to decision theory, it is the study of ways in which strategic interactions among rational players produce outcomes with respect to the preferences (or utilities) of those players [5]. In game theory, each player's actions affect the outcomes of every other player.

In game theory, a game could be categorized from two viewpoints, time sequence of each player's action and the knowledge a player gets from other players, therefore, four types of games: strategic game with complete information, strategic game with incomplete information, extensive game with complete information and extensive game of incomplete information and their associated equilibriums are the basis of game theory (see Table 1). Game theory provides a powerful tool with which to analyze scenarios in which an agent must make decisions in an unpredictable environment [6]. It has been applied to the MAS research in many ways.

Table 1. Four Categories of Game Theory and Equilibriums

	Static Game		Dynamic Game	
Game with perfect information	Strategic game with complete information	Nash equilibrium	Extensive game with complete information	Subgame perfect equilibrium
Game with imperfect information	Strategic game with incomplete information	Bayesian game	Extensive game with incomplete information	Perfect Bayesian equilibrium

Decision Network & Bayesian Network

In a decision making environment there exist many elements: alternatives (decisions), states (factors or random variables representing the world), preferences (indicating the values associated with the different alternatives) and the relationships between the alternatives, states and preferences. This collection of elements form what is known as the decision basis. A decision network (or influence graph) is a graphical representation of the decision basis. The alternatives are represented as decision nodes (rectangles), states are represented as chance nodes (ovals) and preferences as utility nodes (diamonds). The result of inference on a decision network is a ranking of alternatives based on their calculated utility values, as a result of which, the alternative with the highest utility is chosen by the decision maker.

A Bayesian network [14] is a specialization of decision networks that contain only chance nodes. It is used to calculate a probability distribution over a set of hidden/unobserved events, which are either impossible to directly measure or too expensive to measure directly. Inputs to a Bayesian network are readings/states of observed events – these are called evidence. Causal relationships between unobserved and observed events are captured and represented using directed arcs that run from cause to effect. The result of a Bayesian network is a probability distribution over the set of hidden/unobserved states that affect a potential decision. In a situation where one has only one possible course of action, which is predetermine, the Bayesian network structure is all that is required for decision making.

Theory of Evidence

Dempster-Shafer theory of evidence is introduced in the late seventies by Glenn Shafer and Arthur Dempster. The representation of chance in this theory is denoted by a belief function rather than a Bayesian mass distribution [8]. It assigns probability values to *sets* of possibilities rather than single events: their appeal rests on the fact they naturally encode evidence in favor to propositions. The degree of support is indicated by numbers between 0 and 1, and it concerns the combination of degrees of belief rather than focusing on how this numbers are determined [8]. It is the only theory that provides a method to combine two sets of probability data.

Ordered Weighted Averaging (OWA)

Ordered Weighted Averaging (OWA) operator is a kind of operator for aggregation which is introduced by Ronald. R. Yager in 1988. This class of operator could be used for aggregating criteria functions to form overall decision functions with the consideration of the importance of those criteria in many disciplines [12]. OWA offers an ideal bridge between *and*-like operators (t-norms) which express the idea that all the criteria are satisfied and *or*-like operators (s-norms) used to represent the case in which the satisfaction of any of the criteria is the decision maker's desire [12]. Apparently, OWA is a very useful tool for solving software agent's decision making problem, especially for the case that several criteria are involved and uncertainty and risk have to be minimized after the aggregation.

4.2 Computational Techniques

The computational techniques here refer to theories and techniques that serve as the mathematical basis for decision making and to make the design of artificial intelligent systems to a quantitative level.

Genetic Algorithm
Genetic algorithm is a family of computational models inspired by evolution which encodes a potential solution to a certain problem on a simple chromosome-like data structure and applies recombination and mutation operators to these structures in order to preserve critical information [9]. In these algorithms, the search space of a problem is represented by a collection of individuals which are represented by and referred to as chromosomes.

Uncertainty Programming Techniques
Uncertainty programming, including stochastic programming techniques, gives a strong foundation for optimization theory and solving problems in uncertain environments where extensive processing power is available. It is good for complex decision making system which has the characteristic of multi-dimensional, multi-functional and multi-criteria with stochastic and fuzzy parameters [13].

Fuzzy Logic
Fuzzy set theory, originally introduced by Lotfi Zadeh in the 1960's, it is a mathematical representation of uncertainty and vagueness and provide formalized tools for dealing with the imprecision intrinsic to many problems. This theory follows the idea that knowledge could be expressed in a more natural way by using imprecise method called fuzzy sets by which many decision-making and problem-solving tasks that are too complex to be understood quantitatively becomes much simpler [10]. Fuzzy set theory has been adopted in the application of information technology and now more widely used in the artificial intelligence field to deal with uncertain problem domain.

Neural Network
Neural network (also referred to neural nets, artificial neural nets and neural computing) is a method of solving complex, non-determinstic problem by simulating the learning process through massively parallel, highly-interconnected processing systems. It can

make contributions for training and learning software agent to approach reasoning and sensory ability of humans [11].

5. Decision making Framework

All the theories and techniques mentioned in Section 4 have their advantages and disadvantages to address problems arisen in MAS decision making problem. Therefore, when design and/or implement practical MAS, which technique could be applied or how several theories/techniques could be combined together to handle specific decision making problem depends on the characteristic of each type of agents' interactions as we defined in Section 3. A very simple decision making framework is illustrated in Table 2. It is developed to help select appropriate technique to take care of an agent's decision making in MAS.

Table 2. Agent's Decision Making Theories & Techniques

	Cooperation & coordination	**Loose & Strict Competition**		
		Certainty	**Risk**	**Uncertainty**
Decision Support Techniques	Decision Network & Bayesian Network Theory of Evidence	Decision Theory Game Theory (Strategic/Extensive Game with Complete Information) Theory of Evidence	Decision Theory Game Theory (Strategic/Extensive Game with Incomplete Information) Theory of Evidence	Decision Theory Game Theory (Strategic/Extensive Game with Incomplete Information) Theory of Evidence OWA
Computational Techniques	Fuzzy Logic Neural Network	Uncertainty Programming Techniques	Genetic Algorithm Uncertainty Programming Techniques	Genetic Algorithm Uncertainty Programming Techniques

5.1 Cooperation & Coordination

In cooperation and coordination environment, an agent is aware of the goals, intentions, desires and knowledge of other agents in the environment. In making its individual decision, it need not to worry about incorrect information or hidden goals, intentions and capabilities of other agents, therefore, no uncertainty and risk is involved. The process of deciding what action is going to be taken can be handled by Decision and Bayesian networks, fuzzy set theory and Dempster Shafer's theory of evidence when it has several alternative possible solutions.

5.2 Loose & Strict Competition

In competition environment, as defined in Section 3, two cases that are loose competition and strict competition exist because of different properties of agent's *Intention*. And when we try to find what theories or techniques are appropriate for decision making, we have to consider different levels of uncertainty involved in agent's decision making process,

- *Decision Making under Certainty*
  In this case, Decision theory, Game theory and Dempster Shafer theory could work as the decision support theories: utility function, goals, and outcomes in decision theory could serve as the basis; two of the four classes of game which is strategic and extensive game with complete information defined in Game theory are good to select rational solutions among several alternatives. Uncertainty programming techniques could be used as the computational tools to solve optimization problem.

- *Decision Making under Risk*
  In this case, Decision theory, Game theory and Dempster Shafer theory could also work as the decision support theory, in Game theory, strategic and extensive game with incomplete information could be adopted. Genetic algorithm could also be used to calculate the Bayesian Nash equilibrium and again uncertainty programming techniques could deal with optimization problem.

- *Decision Making under Uncertainty*
  The decision support theories and computational techniques that are appropriate for this case is the same as those for the case of risk, in addition, OWA could be adopted as an aggregation technique or computational technique, for example, it could help assign some weighting values according to specific parameter.

6. Conclusions

With wide adoption of the Internet as open environment and the increasingly popularity of machine-independent programming language, software agent and MAS are valuable software engineering abstractions for the development of software systems. In particular, agents' interaction and how an agent could make rational decision making when it is in different situation might pay off to invest some time to understand the basic concepts, to identify how those theories and techniques in the literature could be adopted. This paper has reviewed and extended a popular used software agent model – BDI agent model; represented three types of relationships among agents, distinguished different levels of uncertainty when it is in competition relationship with its opponents; reviewed some decision making support theories and computational techniques; accordingly, a MAS decision making framework was developed. It could be adopted as a guidance when building practical software agent and thus facilitate the design and implementation of software agent and MAS.

References

[1] Behrouz Homayoun Far, et al. Formalization of Organizational Intelligence for Multiagent System Design. IEICE Trans. Inf. & Syst., Vol. E83-D. No. 4, 2000.

[2] Hiroyuki Onjo, Behrouz Homayoun Far. A Unified View of Heterogeneous Agents' Interaction, IEICE Trans. Inf. & Syst., Vol. E84-C, No. 8, 2001.

[3] Michael Wooldgidge. Intelligent Agents. In Multiagent Systems: A Mordern Approach to Distributed Artificial Intelligence, ed. Gerhard Weiss. pp. 27-73. MIT Press. 1999.

[4] Anand S. Rao, Michael P. Georgeff. BDI Agents: From Theory to Practice. Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95). San Francisco, USA, 1995.

[5] Stanford Encyclopedia of Philosophy: Game theory. http://plato.stanford.edu/entries/game-theory/

[6] Simon Persons, Michael Wooldridge. Game Theory and Decision Theory in Multi-Agent Systems. Autonomous Agents and Multi-Agent Systems, 5, 243-254, 2002.

[7] Pedro Larrafiaga, Cindy M. H. Kuipers, Roberto H. Murga, and Yosu Yurramcndi. Learning Bayesian Network Structures by Searching for the Best Ordering with Genetic Algorithms. IEEE Transactions on Systems, Man and Cybernetics, Vol. 26. No. 4. July 1996.

[8] Glenn Shafer. A Mathematical Theory of Evidence. Princeton University Press. Princeton, New Jersey, 1976.

[9] Darrell Whitley. A Genetic Algorithm Tutorial. Computer Science Department. Colorado State University.

[10] Fuzzy Logic: http://www.emsl.pnl.gov:2080/proj/neuron/fuzzy/what.html

[11] Mohammed H. A. Tafti. Neural Network: A New Dimension in Expert System Application. Proceedings of the 1990 ACM SIGBDP conference on Trends and directions in expert systems. New York. 1990.

[12] Ronald. R. Yager. On Ordered Weighted Averaging Aggregation Operators in Multicriteria Decision Making. IEEE Transactions on Systems, Man and Cybernetics, Vol.18. No. 1. Jan/Feb. 1988.

[13] Baoding Liu, Riqing Zhao. Stochastic Programming and Fuzzy Programming. Tsinghua Express. Beijing. 2001.

[14] Judea Pearl. Reasoning under Uncertainty. Annual Review of Computer Science, 4:37-72, 1990.