

A Cognitive Model for Multi-Agent Collaboration

Zhongzhi Shi¹, Jianhua Zhang^{1,2}, Jinpeng Yue^{1,2}, Xi Yang¹

¹Key Laboratory of Intelligent Information Processing, Institute of Computing Technology,
Chinese Academy of Sciences, Beijing, China

²University of Chinese Academy of Sciences, Beijing, China

Email: shizz@ics.ict.ac.cn, zhangjh@ics.ict.ac.cn, yuejp@ics.ict.ac.cn, yangx@ics.ict.ac.cn

Received October 27, 2013; revised November 23, 2013; accepted November 30, 2013

Copyright © 2014 Zhongzhi Shi *et al.* This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. In accordance of the Creative Commons Attribution License all Copyrights © 2014 are reserved for SCIRP and the owner of the intellectual property Zhongzhi Shi *et al.* All Copyright © 2014 are guarded by law and by SCIRP as a guardian.

ABSTRACT

In multi-agent system, agents work together for solving complex tasks and reaching common goals. In this paper, we propose a cognitive model for multi-agent collaboration. Based on the cognitive model, an agent architecture will also be presented. This agent has BDI, awareness and policy driven mechanism concurrently. These approaches are integrated in one agent that will make multi-agent collaboration more practical in the real world.

KEYWORDS

Cognitive Model; Multi-Agent Collaboration; Awareness; ABGP Model; Policy Driven Strategy

1. Introduction

Collaboration occurs over time as organizations interact formally and informally through repetitive sequences of negotiation, development of commitments, and execution of those commitments. Both cooperation and coordination may occur as part of the early process of collaboration, and collaboration represents a longer-term integrated process. Gray describes collaboration as “a process through which parties who see different aspects of a problem can constructively explore their differences and search for solutions that go beyond their own limited vision of what is possible” [1].

Agent is an active entity and chooses a decision process dynamically through interactive with environment [2]. The agent makes a choice from a large set of possibilities which can determine the transformation of environmental information, the internal information and decision process. Thus the result is the mode of operation that an agent choose. In multi-agent system, the agent performs one or multiple tasks. In other words, the internal information from an agent is also a function of the information that it receives from other agents. This represents the problem in a new way *i.e.* the multi-agent takes decision collaboratively for which it optimizes the team behavior. Thus individual decision is avoided for the global interest irrespective of environmental input.

Hence a better action is chosen in a collaboration manner according to the environment information.

In a multi-agent system, no single agent owns all knowledge required for solving complex tasks. Agents work together to achieve common goals, which are beyond the capabilities of individual agent. Each agent perceives information from the environment with sensors and finds out the number of cognitive tasks, selects the particular combination for execution in an interval of time, and finally outputs the required effective actions to the environment. There is one significant prior work formalizing joint activities. Paper [3] provides analyses for Bratman’s three characteristic functional roles played by intentions, and shows how agents can avoid intending all the foreseen side-effects of what they actually intend. Paper [4] describes collaborative problem-solving and mixed initiative planning. However, these models focus more on formal aspects of belief states and reasoning rather than how agents behave. Other works, such as COLLAGEN [5] and RavenClaw [6] focus on task execution but lack explicit models of planning or communication. The PLOW system [7] defines an agent that can learn and execute new tasks, but the PLOW agent is defined in procedural terms making it difficult to generalize to other forms of problem-solving behavior. Paper [8] builds cognitive model on the theories, accounts for col-

laborative behavior including planning and communication, and in which tasks are represented declaratively to support introspection and the learning of new behaviors, but only focuses on internal mental state of agent and does not consider environment situation.

As an internal mental model of agent, BDI model has been well recognized in philosophical and artificial intelligence area. Bratman's philosophical theory was formalized by Cohen and Levesque [3]. In their formalism, intentions are defined in terms of temporal sequences of agent's beliefs and goals. Rao and Georgeff have proposed a possible-world's formalism for BDI architecture [9]. The abstract architecture they proposed comprises three dynamic data structures representing the agent's beliefs, desires, and intentions, together with an input queue of events. The architecture allows update and query operations on the three data structures [10]. The update operations on beliefs, desires, and intentions are subject to respective compatibility requirements. These functions are critical in enforcing the formalized constraints upon the agent's mental attitudes. The events that the system can recognize include both external events and internal events.

Wooldridge and Lomuscio proposed a multi-agent VSK logic which allows us to represent what is objectively true of an environment, what is visible, or knowable about the environment to individual agents within it, what agents perceive of their environment, and finally, what agents actually know about their environment [11]. In multi-agent system, group awareness is an understanding of the activities of others and provides a context for own activity. Group awareness can be divided into basic questions about who is collaborating, what they are doing, and where they are working [12]. When collaborators can easily gather information to answer these questions, they are able to simplify their verbal communication, able to better organize their actions and anticipate one another's actions, and better able to assist one another.

Policy-Based Management is a management paradigm that separates the rules governing the behavior of a system from its functionality. It promises to reduce maintenance costs of information and communication systems while improving flexibility and runtime adaptability. A rational agent is any entity that perceives and acts upon its environment, and selects actions based on the information receiving from sensors and built-in knowledge which can maximize the agent's objective. Paper [13] points out following notions: A reflex agent uses if-then action rules that specify exactly what to do under the current condition. In this case, rational behavior is essentially compiled in by the designer, or somehow pre-computed. A goal-based agent exhibits rationality to the degree to which it can effectively determine which actions to take to achieve specified goals, allowing it greater

flexibility than a reflex agent. A utility-based agent is rational to the extent that it chooses the actions to maximize its utility function, which allows a finer distinction among the desirabilities of different states than do goals. These three notions of agent hood can fruitfully be codified into three policy types for multi-agent system.

In this paper, a cognitive model for multi-agent collaboration will be proposed in terms of external perception and internal mental state of agents. The agents thus are in a never-ending cycle of perception, goal selection, planning and execution.

2. Cognitive Model ABGP

Agent can be viewed as perceiving its environment information through sensors and acting environment through effectors [14]. A cognitive model for multi-agent collaboration should consider external perception and internal mental state of agents. Awareness is knowledge created through interaction between an agent and its environment. Endsley pointed out awareness has four basic characteristics [15]:

Awareness is knowledge about the state of a particular environment.

Environments change over time, so awareness must be kept up to date.

People maintain their awareness by interacting with the environment.

Awareness is usually a secondary goal—that is, the overall goal is not simply

Gutwin *et al.* proposed a conceptual framework of workspace awareness that structures thinking about groupware interface support. They list elements for the conceptual framework [16]. Workspace awareness in a particular situation is made up of some combination of these elements.

For internal mental state of agents we can consult BDI model which was conceived by Bratman as a theory of human practical reasoning. Its success is based on its simplicity reducing the explanation framework for complex human behavior to the motivational stance [17]. This means that the causes for actions are always related to human desires ignoring other facets of human cognition such as emotions. Another strength of the BDI model is the consistent usage of folk psychological notions that closely correspond to the way people talk about human behavior.

In terms of above considerations, we propose a cognitive model for multi-agent collaboration through 4-tuple <Awareness, Belief, Goal, Plan>. And the cognitive model can be called ABGP model.

2.1. Awareness

Multi-agent awareness should consider basic elements and relationship in multi-agent system. Multi-agent

awareness model is defined as 2-tuple $MA = \{\text{Element}, \text{Relation}\}$, where Elements of awareness as follows:

- Identity (Role): Who is participating?
- Location: Where are they?
- Intentions: What are they going to do?
- Actions: What are they doing?
- Abilities: What can they do?
- Objects: What objects are they using?
- Time point: When do the action execute?

Basic relationships contain task relationship, role relationship, operation relationship, activity relationship and cooperation relationships.

Task relationships define task decomposition and composition relationships. Task involves activities with a clear and unique role attribute

Role relationships describe the role relationship of agents in the multi-agent activities.

Operation relationships describe the operation set of agents.

Activity relationships describe activity of the role at a time.

Cooperation relationships describe the interactions between agents. A partnership can be investigated through cooperation activities relevance among agents to ensure the transmission of information between different perception of the role and tasks for maintenance of the entire multi-agent perception

2.2. Belief

Belief represents the information, an agent has the world it inhabits and its own internal state. This introduces a personal world view inside the agent. Belief can often be seen as knowledge base of an agent, which contains abundant contents, including basic axioms, objective facts, data and so on.

Belief knowledge base is a 3-triple $K = \langle T, S, B \rangle$, where, T describes the basic concepts in the field and their definitions, axioms form domain concepts, namely domain ontology; S is the domain between facts and formulas there is a causal relationship between constraints, called causality constraint axiom, which ensures consistency and integrity of the knowledge base; B is the set of beliefs in current state, containing facts and data. The contents of B is changed dynamically.

2.3. Goal

Goals represent the agent's wishes and drive the course of its actions. A goal deliberation process has the task to select a subset of consistent desires. In a goal-oriented design, different goal types such as achieve or maintain goals can be used to explicitly represent the states to be achieved or maintained, and therefore the reasons, why actions are executed. Moreover, the goal concept allows

modeling agents which are not purely reactive *i.e.*, only acting after the occurrence of some event. Agents that pursue their own goals exhibit pro-active behavior.

There are three ways to generate goals:

- System designer sets the goal or the goal will be selected when the system initializing.
- Generate the goal according to observing the dynamic of the environment.
- Goals generate mainly stem from internal state of agent.

2.4. Plan

When a certain goal is selected, agent must looking for an effective way to achieve the goal, and sometimes even need to modify the existing goal, this reasoning process is called planning. In order to accomplish this plan reasoning, agent can adopt two ways: one is using already prepared plan library, which includes some of the planning for the actual inference rules which also called static planning; another way is instantly planning, namely dynamic programming.

Static plan is aimed at some specific goal, pre-established goals needed to achieve these basic processes and methods, thus forming the corresponding goals and planning some of the rules, that planning regulations. Since these rules in the system design has been written, the planning process actually becomes a search in the library in the planning process of matching.

Dynamic plan finds an effective way to achieve the goals and means for a certain goal which based on the beliefs of current status, combined with the main areas of the axioms and their action description. In the dynamic situation, the environment domain is dynamically changing, the beliefs are also constantly changing. Even for the same goal, in a different state, the planning and implementation of processes may be different. Therefore, the dynamic plan is very important for multi-agent system, especially in a complex environment.

In dynamic mode, we use top-down methods. Typically a goal might be composed by a number of sub-goals, so we must first achieve all sub-goals of the goal.

3. Policy-Driven Strategy

There are many definitions about the policy, and different area has different standard. For example, IETF/DMTF policy is defined as a series of management to a set of rules [18]. Many people regard the policy simply equated with defined rules, which is obviously too narrow. We use relatively broad, loosely definition: policy is used to guide the behavior of the system means. In the multi-agent system, policy describes agent behavior that must be followed, which reflects the human judgment. Policy

tells agent what should do (objectives), how to do (action), what extent (utility) which guide the agent behavior.

Paper [13] defines a unified framework for autonomic computing policies which are based upon the notions of states and actions. In general, one can characterize a system, or a system component, as being in a state S at a given moment in time. Typically, the state S can be described as a vector of attributes, each of which is either measured directly by a sensor, or perhaps inferred or synthesized from lower-level sensor measurements. A policy will directly or indirectly cause an action “a” to be taken, and the result of which is that the system or component will make a deterministic or probabilistic transition to a new state. This unified framework also fits to multi-agent system.

A multi-agent system at a time t in state s_0 , s_0 usually consists of a series of attributes and values to characterize. Policy actions directly or indirectly caused by the execution of action, causing the system to shift to a new state s , as shown in **Figure 1**. Policy can be seen as a state transition function, and multi-agent system changes their state based on policies.

A policy P is defined as four-tuples $P = \{S_t, A, S_g, U\}$, where S_t is the trigger state set, that is, the state of implementation of the policy is triggered collection; A is action set; S_g is the set of goal states; U is the goal state utility function set, to assess the merits of the goal state level. At least three types of policy will be useful for multi-agent system.

3.1. Action Policies

Action policy describes the action that should be taken whenever the system is in a given current state. Action policy does not take appropriate action after describing the system achieved state, that does not give, nor given goal state utility function. Typically action policy $P = \{S_t, A, _, _ \}$, where “_” indicates empty. Action takes the form of IF (Condition) THEN (Action). Condition speci-

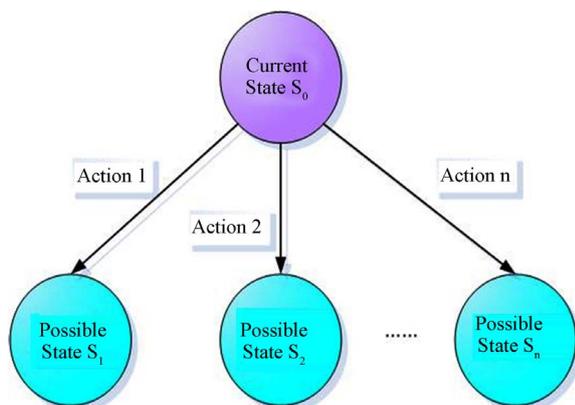


Figure 1. Multi-agent policy model based on the state transition.

fies either a specific state or a set of possible states that all satisfy the given Condition. Note that the state that will be reached by taking the given action is not specified explicitly.

3.2. Goal Policies

Goal policy does not give the system a state should take action, but to describe the system required to achieve the goal state. Goal policies specify either a single desired state, or one or more criteria that characterize an entire set of desired states. Unlike action strategies as dependent people come clear that the action taken, but goal policy according to the goal to generate a reasonable action. Usually goal policy $P = \{S_t, _, S_g, _ \}$.

3.3. Utility Function Policies

A Utility Function policy is an objective function that expresses the value of each possible state. Utility function policies generalize goal policies. Utility function policies provide a more detailed and flexible mechanism than the goal policies and action policies, but the utility policies need policy makers on the system model which has a more in-depth and detailed understanding of the need for more modeling, optimization and algorithms. Typically utility function policy $P = \{S_t, _, _, U \}$.

4. Multi-Agent Collaboration

Multi-agent systems are computational systems in which a collection of loosely autonomous agents interact to solve a given problem. As the given problem is usually beyond the agent’s individual capabilities, agent need to exploit its ability to collaboration, communicate with its neighbors. For multi-agent collaboration there are two general approaches, one approach is through awareness facility to specify the cooperation relationship between agents; another approach is through joint intention.

As above mentioned cooperation relationships of awareness in ABGP model, we describe the interactions between agents. Agent collaboration is defined explicitly when multi-agent is designed. A partnership can be investigated through cooperation activities relevance between agents to ensure the transmission of information between different perception of the role and tasks for maintenance of the entire multi-agent perception.

In the joint intention theory, a team is defined as a set of agents having a shared objective and a shared mental state. Joint intentions are held by the team as a whole, and require each team member to informing other one whenever it detects the goal state change, such as goal achieved, or it is impossible to archive or as the dynamic of the environment, the goal is no relevant. For more detail you can refer to paper [19]

5. Agent Architecture

In terms of the cognitive model for multi-agent collaboration an agent architecture has been proposed shown in **Figure 2**. The abstract architecture we propose comprises four dynamic data structures representing the agent's awareness, belief, desire, and plan, together with an input queue of events. We allow update and query operations on the four data structures. The update operations on awareness, beliefs, desires, and intentions are subject to respective compatibility requirements. These functions are critical in enforcing the formalized constraints upon the agent's mental attitudes, also observing the environment situation. The events of the system includes include both external events and internal events. We assume that the events are atomic and are recognized after they have occurred. Similarly, the outputs of the agent actions are also assumed to be atomic.

The main interpreter loop is given below. We assume that the event queue, awareness, belief, desire, and intention structures are global.

```

ABGP-interpreter
  initialize-state();
  repeat
    get-new-events-from-awareness();
    get-new-events-from-ACL-communication();
    options:= option-generator(event-queue);
    selected-options:= deliberate(options);
    update-plans(selected-options);
    execute();;
    drop-successful-attitudes();
    drop-impossible-attitudes();
  end repeat

```

This agent architecture has three outstanding features:

Agent has BDI reasoning which is a successful approach for multi-agent systems.

Agent makes deliberation not only depending on in

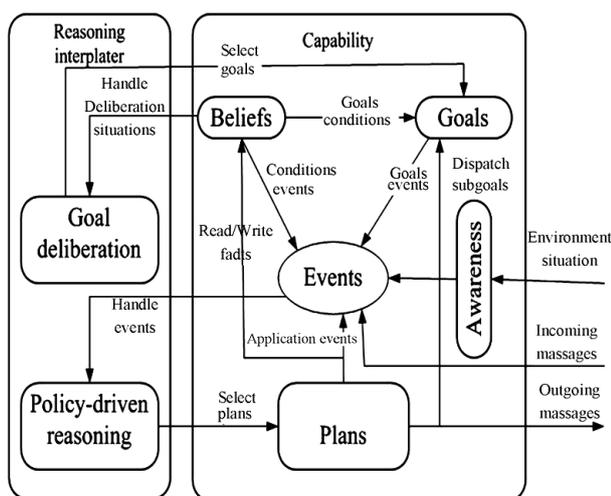


Figure 2. Agent architecture.

ternal mental state, it also concerns awareness information from outside environment. This feature makes agent collaborative more reality.

Policy-driven reasoning can improve the performance and enhance flexibility.

6. Conclusions and Future Works

In multi-agent system, agents work together for solving complex tasks and reaching common goals. A cognitive model for multi-agent collaboration is proposed in this paper. Based on the cognitive model, we develop an agent architecture which has BDI, awareness and policy driven mechanism concurrently. These approaches are integrated in one agent that will cause multi-agent collaboration in a more practical world.

We are going to develop a multi-agent collaboration software system, and apply it to animal robot collaboration work in the future.

Acknowledgements

This work is supported by the National Program on Key Basic Research Project (973) (No. 2013CB329502), National Natural Science Foundation of China (No. 61035003, 60933004, 61202212, 61072085), National High-tech R&D Program of China (863 Program) (No. 2012AA011003), National Science and Technology Support Program (2012BA107B02).

REFERENCES

- [1] B. Gray, "Collaborating: Finding Common Ground for Multiparty Problems," Jossey-Bass, San Francisco, 1989.
- [2] Z. Z. Shi, "Advanced Artificial Intelligence," World Scientific Publishing Co., Singapore City, 2011. <http://dx.doi.org/10.1142/7547>
- [3] P. R. Cohen and H. J. Levesque, "Intention Is Choice with Commitment," *Artificial Intelligence*, Vol. 42, No. 2-3, 1990, pp. 213-361. [http://dx.doi.org/10.1016/0004-3702\(90\)90055-5](http://dx.doi.org/10.1016/0004-3702(90)90055-5)
- [4] G. Ferguson and J. F. Allen, "Mixed-Initiative Dialogue Systems for Collaborative Problem-Solving," *AI Magazine*, Vol. 28, No. 2, 2006, pp. 23-32.
- [5] C. Rich and C. L. Sidner, "COLLAGEN: When Agents Collaborate with People," *1st International Conference on Autonomous Agents*, Marina del Rey, 5-8 February 1997, pp. 284-291. <http://dx.doi.org/10.1145/267658.267730>
- [6] D. Bohus and A. Rudnicky, "The RavenClaw Dialog Management Framework: Architecture and Systems," *Computer Speech & Language*, Vol. 23, No. 3, 2009, pp. 332-361. <http://dx.doi.org/10.1016/j.csl.2008.10.001>
- [7] J. Allen, N. Chambers, G. Ferguson, L. Galescu, H. Jung, M. Swift and W. Taysom, "PLOW: A Collaborative Task Learning Agent," *Proceedings of the 22nd Conference on Artificial Intelligence (AAAI-07)*, Vancouver, 22-216 July

- 2007, pp. 1514-1519.
- [8] G. Ferguson and J. Allen, "A Cognitive Model for Collaborative Agents," *Proceedings of the AAAI 2011 Fall Symposium on Advances in Cognitive Systems*, Washington DC, 4-6 November 2011, pp. 112-120.
- [9] A. S. Rao and M. P. Georgeff, "Modeling Rational Agents within a BDI-Architecture," In: J. Allen, R. Fikes and E. Sandewall, Eds., *Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning*, Morgan Kaufmann Publishers, San Mateo, 1991.
- [10] L. Braubach and A. Pokahr, "Developing Distributed Systems with Active Components and Jadex," *Scalable Computing: Practice and Experience*, Vol. 13, No. 2, 2012, p. 100.
- [11] M. Wooldridge and A. Lomuscio, "Multi-Agent VSK Logic," *Proceedings of the 17th European Workshop on Logics in Artificial Intelligence (JELIAI-2000)*, Springer-Verlag, Berlin, 2000.
- [12] C. Gutwin and S. Greenberg, "A Descriptive Framework of Workspace Awareness for Real-Time Groupware," *JCSCW*, Vol. 11, 2002, pp. 411-446.
- [13] J. O. Kephart and W. E. Walsh, "An Artificial Intelligence Perspective on Autonomic Computing Policies," *Proceedings of the 5th IEEE International Workshop on Policies for Distributed Systems and Networks*, Yorktown Heights, 7-9 June 2004, pp. 3-12.
- [14] Z. Z. Shi, X. F. Wang and J. P. Yue, "Cognitive Cycle in Mind Model CAM," *International Journal of Intelligence Science*, Vol. 1, No. 2, 2011, pp. 25-34. <http://dx.doi.org/10.4236/ijis.2011.12004>
- [15] M. Endsley, "Toward a Theory of Situation Awareness in Dynamic Systems," *Human Factors*, Vol. 37, No. 1, 1995, pp. 32-36. <http://dx.doi.org/10.1518/001872095779049543>
- [16] C. Gutwin and S. Greenberg, "The Importance of Awareness for Team Cognition in Distributed Collaboration," In: E. Salas and S. Fiore, Eds., *Team Cognition: Understanding the Factors That Drive Process and Performance*, American Psychological Association, Washington DC, 2004, pp. 177-201. <http://dx.doi.org/10.1037/10690-009>
- [17] A. Pokahr and L. Braubach, "The Active Components Approach for Distributed Systems Development," *IJ-PEDS*, Vol. 28, No. 4, 2013, pp. 321-369.
- [18] Distributed Management Task Force, Inc. (DMTF), "Common Information Model (CIM) Specification, Version 2.2," 1999. <http://www.dmtf.org/spec/cims.html>
- [19] J. H. Zhang, Z. Z. Shi, J. P. Yue, *et al.*, "Joint Intention Based Collaboration," *IJCAI-13 Workshop WIS2013*, 2013.