J Syst Sci Syst Eng (Jun 2007) 16(2): 222-234 ISSN: 1004-3756 (Paper) 1861-9576 (Online)
DOI: 10.1007/s11518-007-5043-9 CN11-2983/N

EXPERT DISCOVERY AND KNOWLEDGE MINING IN COMPLEX MULTI-AGENT SYSTEMS*

Minjie ZHANG[1] Xijin TANG[2] Quan BAI[1] Jifa GU[2]

[1]*School of Computer Science and Software Engineering, University of Wollongong,*
Wollongong, NSW 2522, Australia
minjie@uow.edu.au (✉), *quan@uow.edu.au*
[2]*Institute of System Science, Chinese Academy of Sciences,*
Beijing, 100080, China
xjtang@iss.ac.cn, jfgu@amss.ac.cn

Abstract

Complex problem solving requires diverse expertise and multiple techniques. In order to solve such problems, complex multi-agent systems that include both of human experts and autonomous agents are required in many application domains. Most complex multi-agent systems work in open domains and include various heterogeneous agents. Due to the heterogeneity of agents and dynamic features of working environments, expertise and capabilities of agents might not be well estimated and presented in these systems. Therefore, how to discover useful knowledge from human and autonomous experts, make more accurate estimation for experts' capabilities and find out suitable expert(s) to solve incoming problems ("Expert Mining") are important research issues in the area of multi-agent system. In this paper, we introduce an ontology-based approach for knowledge and expert mining in hybrid multi-agent systems. In this research, ontologies are hired to describe knowledge of the system. Knowledge and expert mining processes are executed as the system handles incoming problems. In this approach, we embed more self-learning and self-adjusting abilities in multi-agent systems, so as to help in discovering knowledge of heterogeneous experts of multi-agent systems.

Keywords: Knowledge discovery, knowledge mining, expert mining, multi-agent system

1. Introduction

Nowadays, many complex problems require diverse expertise and multiple techniques. In order to solve such problems, numbers of heterogeneous agents (Lesser 1998, 1999), which may include both of human experts and autonomous agents, are sometimes required to work together in some open domains (Artikis & Pitt 2001, Huhns & Stephens 1999). In these complex domains, agents' number, experiences

* This work was supported by a Scientific Cooperation Program between Chinese Academy of Sciences and Australian Academy of Science in 2005 - 2006, and National Natural Sciences Foundation of Science (Grant No. 70571078 & 70221001). The original version was presented at the conference of KSSS06.

and expertise may not be stable. Also, due to heterogeneities, expertise and capabilities of agents might not be well estimated by the system. In this case, how to discover useful knowledge from human and autonomous experts, make more accurate estimation for experts' capabilities and find out suitable expert(s) to solve incoming problems are important research issues in multiple areas, which include multi-agent systems (MASs), distributed information retrieval, distributed problem solving, data-mining, etc.

For complex multi-agent systems (CMASs), cooperative problem solving can normally be achieved through the following three major approaches: (1) cooperation protocols (Lesser 1999), (2) coordination mechanisms (Bai & Zhang 2006), and (3) communication language or protocols (Finin et al 1994, Labrou et al 1999). Most cooperation and coordination protocols or mechanisms are managed by centralized control. When an agent-world becomes more complex and dynamic, centralized approaches show the strong weakness in effective coordination among agent systems. Traditional communication languages are message-based communication and have been adopted by many multi-agent applications and agent platforms, such as JACK (Busetta et al 2007), ZEUS (Nwana et al 1999) and FIPA-OS (2007). However, the message-based communication approaches also have some weaknesses. Agents that use the message-based language to communicate with each other must fully understand all vocabularies of the language. This shortcoming restricts agents to limited vocabularies, and increases the risk of the communication. In order to overcome the limitations of message-based communication, some researchers in MASs proposed on the knowledge-based communication. Generally, knowledge-based communication languages enable agents to express properties of objects and relationships between objects within their information. One typical knowledge-based ACL is the Knowledge Interchange Format (KIF) (Genesereth & Fikes 1992). Knowledge-based communication languages are widely applied in web-based applications. However, most knowledge-based communication languages were not originally developed for agent applications in open environments. Therefore many agent actions cannot be described in these languages. This feature greatly restricts the use of these languages.

In order to overcome some limitations mentioned above, we have proposed different decentralized approaches for complexity of problem solving through different methodologies, such as dynamic team forming strategies (Bai & Zhang 2005), a coloured Petri Net approach (Bai et al 2007) for modeling of interactions among heterogeneous agents in both self-interested agent systems and cooperative agent systems. These approaches are designed for traditional MASs but might not work properly for complex MASs in which external knowledge sources of a multi-agent system (MAS) could be developed by different organizations for different purposes. These knowledge sources update their data independently, and may use different formats to represent knowledge. To work together in the same MAS to achieve some common goals, these heterogeneous agents need to interact with each other to exchange their knowledge or acquire domain knowledge from the system. The motivation of this research is to help agents obtain the latest domain knowledge from various knowledge sources through knowledge management mechanisms in MASs to form a knowledge level communication and interaction for complex problem solving in open domains.

In this paper, we introduce an ontology-based approach for knowledge and expert mining (Kovalerchuk et al 2001), which is to discover specialised knowledge and expertise of experts, in hybrid MASs. In this approach, ontologies are used to describe knowledge of a system. The knowledge and expert mining, which is a life-time long process, will be performed through update the ontology of the system. The knowledge and expert mining are processed as the experts of the system solve incoming problems.

The rest of the paper is organised as follows. In Section 2, we present some basic concepts of MASs and CMASs. In Section 3, the concept of ontology and ontologies in MASs are introduced. We propose the ontology-based approach in Section 4. Finally, the conclusion is given in Section 5.

2. Experts and Knowledge in Complex Multi-agent Systems

Knowledge discovery is a preliminary and an important process for MAS applications. It mainly contains two processes:

1. Initial understanding the problem domain and describing incoming problems in an agent-readable format;
2. Understanding agents' capabilities and expertises.

Knowledge discovery for MASs with heterogenous agents is more difficult then that in common MASs. Here, we define a MAS with hybrid agents (human and autonomous agents) as a complex multi-agent system (CMAS). As a CMAS, agents of the system could be from various originations and have various expertises, knowledge and capabilities. This makes it hard to describe agents' knowledge clearly in a particular formal way. Agents of a CMAS are experts in some particular area(s). They not only possess special knowledge but also expertise and experience to solve some particular problems. Here, we take medical diagnosis as an example. Various human experts or/and diagnose-agents from different branches are grouped together to give proper diagnosis for patients. Each agent has its own knowledge, experience and expertise of own branch area. However, some of their knowledge (especially for human experts) is hard to be formally described or even discovered. Through several preliminary knowledge mining steps, it is impossible to find out and describe all knowledge of every agent in the system. Furthermore, some kinds of information, such as experience, even might not be well realised by the agent itself.

Definition 1: An expert is a major member (either a software agent or a human expert) in a complex multi-agent, with the knowledge of the agent world and the expertise for a particular problem solving in a special field.

Definition 2: Knowledge of an expert can be classified as domain knowledge (understanding of the agent world) and special knowledge (expertise).

Comparing with common agents, an expert is more sophistic. It has ability to check, modify and update knowledge according to its own expertise and experience. It has strong learning ability to improve its capability (accumulate experience). On the other hand, experts also bring dynamics to the system. As an expert improves its knowledge and ability, the overall capability and domain knowledge of the system should also be modified (see Figure 1).

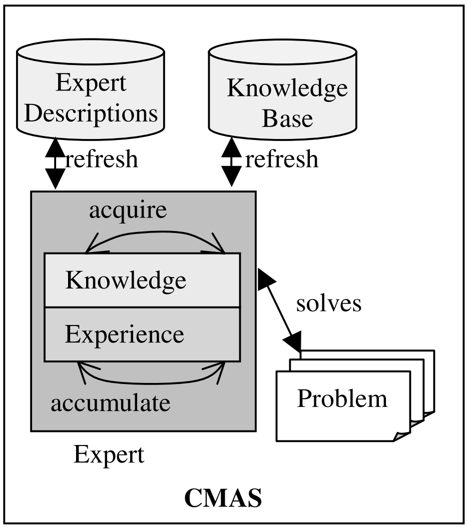

Figure 1 Expert in a CMAS

For a CMAS, expert discovery is another important process. This process is to estimate expertise of experts, category experts into proper areas and find out suitable experts for incoming tasks.

In this research, we use ontology to describe knowledge of agents, experts, and domain knowledge of complex systems. Ontologies are formal knowledge specifications that can be understood by agents. In our research, we take OIL languages as the ontology specification language. OIL (Fensel et al 2001) is an advanced ontology language that has rich representation formalism and ontological modelling primitives. An ontology in OIL is represented via a set of ontology definitions. Four key words, which are *class-def*, *subclass-of*, *slot-def*, and *slot-constraint*, are used in ontology definitions to describe relationships of different concepts. The meanings of these four key words are described as follows:

class-def: a class definition associated with a class name and a class description;

subclass-of: a statement of a class's parent class(es);

slot-def: a slot definition associated with a slot name; and

slot-constraint: a list of global constraint(s) applied to a slot (a slot can also be called as a role or an attribute).

3. Ontology

To facilitate knowledge and expert discovery in CMASs, in this research, we establish ontology to describe conceptually concise basis of system knowledge.

3.1 Ontologies in MASs

In the area of MASs, an ontology is a description of the concepts, relationships and constraints that can exist for an agent/expert or a community of agents/experts (Bai & Zhang 2004, Gruber 1991, Zhang & Li 2000). It can provide not only a description for knowledge contents but also relationships between different knowledge. Ontologies in MASs normally specify conceptualization of a domain in terms of concepts. Each concept represents a class for a specific set of entities. In an ontology, the concepts are typically organized into a taxonomy tree, and each node of the tree represents a concept. Concepts are linked together by means of their semantic relationships. In Figure 2, we give a simple example of ontology that describes “University Department”. In this example, nodes represent concepts of the domain and arrows show the relationships between these concepts.

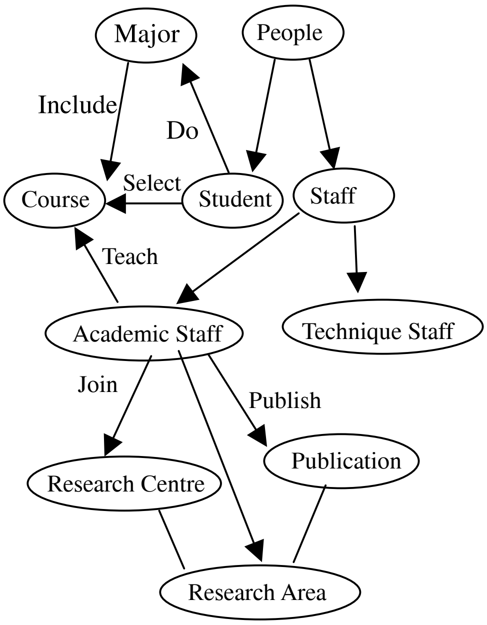

Figure 2 Example of Ontology

Ontologies of a MAS can be classified as common ontology and special ontology. Common ontology is used to describe domain knowledge of the system. It is written in a formal language which can be understood by all agents of the system. Especially in CMAS, some sophistic agents (experts) may have unshared ontologies for its specialized area. This kind ontologies are defined as special ontologies. A special ontology is not published in the system and could be written in some specific format that only can be understood by the expert. An expert can also publish its special ontology to the system. However, the special ontology must be translated into the common language and map into common ontology of the system before it is published. This process can be executed by facilitator agents of the system (see Figure 3). The formal definitions of these two types of ontologies are defined by Definition 3 and Definition 4, respectively.

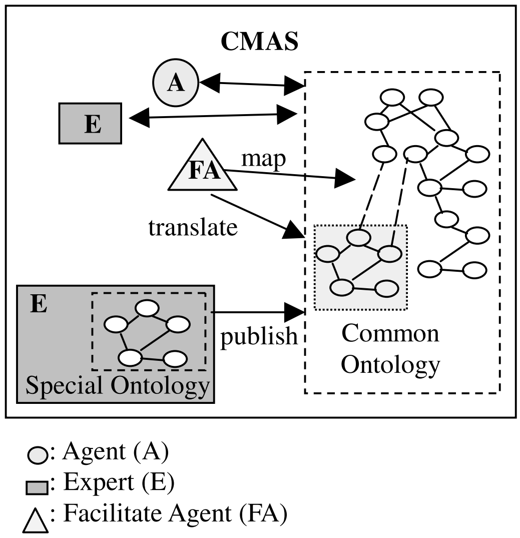

Figure 3 Common and special Ontology in a CMAS

Definition 3: A common ontology is a description of common knowledge that is related to a MAS's working domain or the multi-agent society. Common ontologies are reachable for all agents of a MAS.

Definition 4: A special ontology is a description of some specific knowledge that is related to some particular agents or a single agent of a MAS. A special ontology is generated by an individual agent from the local view point of the agent. Special ontologies are not accessible for all agents of a MAS.

3.2 Inclusion of Experts in Common Ontology

Experts bring dynamics and difficulties for knowledge discovery in CMASs. On the other hand, they are also the most important part of a CMAS. Whether the knowledge of experts are discovered and applied properly is the key benchmark to evaluate a CMAS. To explore experts' knowledge and expertise, in this research, we include experts into the common ontology of the CMAS. Through this way, experts can exist in the common ontology as a class of knowledge and have links with other domain knowledge of the system (see Figure 4). This brings conveniences for task allocation and knowledge discovery. Through an expert's category and linked knowledge, we can find more effective and pertinent way to achieve expert mining.

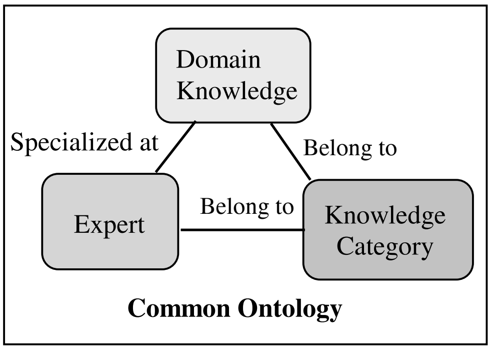

Figure 4 Including experts in system common ontology

Same as other common ontologies, an expert is also described in a unified/formal format. In this research, we define three compulsory properties for concept expert: specialized area, expertise knowledge and experience value. Figure 5 shows a simple example of expert ontology that describes a gynecologist. The experience of an expert represents how good the expert is. Experience value must be estimated by a unified rule among the system.

Expert name: Dr. Munez;
Specialist Area: gynecologist;
Expertise Knowledge: gynaecology;
Experience: 58;

Figure 5 Expert ontology example

Ontology provides common understanding in knowledge sharing channels between agents and experts and supports expert mining, i.e. to discover suitable team members for problem solving through ontology (knowledge) management. In our knowledge management mechanism, the knowledge and ontology searching services are accomplished by the *Knowledge Searcher* of a MAS. Generally, the *Knowledge Searcher* takes two procedures to process a searching query of an agent, i.e. (1) to search the related knowledge category from the ontology-base; and (2) to retrieve the related knowledge from the knowledge-base. When an agent wants to acquire some domain knowledge, it will send a query to the *Knowledge Searcher*. Receiving a query, the *Knowledge Searcher* searches the ontology-base to obtain related knowledge categories of the query from the *Domain Knowledge Ontology*. Then, the *Knowledge Searcher* retrieves related knowledge from the knowledge-base according to the knowledge category, which is obtained from the ontology-base. Finally, the *Knowledge Searcher* sends the searching results to the agent.

4. Approach for Knowledge and Expert Discovery

Many current CMAS applications focus on

how to extract experts' special rules and convert these rules as system expertise. However, most current CMAS applications have some limitations. Firstly, if knowledge extraction is performed without some particular purpose, it is very hard to say whether the extracted knowledge is useful for the system. Secondly, most of these kind approaches will meet the difficulty to extract knowledge from heterogenous experts. Even if the CMAS can perform knowledge translation between several knowledge representation formats (languages), it is still hard to predict knowledge representation format of the incoming expert especially in an open environment. Finally, most current CMAS applications take expert mining as a one-time process. However, since experts have high learning ability and may work in open environments, knowledge, expertises and experiences of experts are updated frequently. Hence, the (one-time) mining result might be not accurate and complete. Considering these limitations, in this section, we present an ontology-based approach for knowledge and expert discovery in CMASs.

4.1 Expert Estimation and Description

When a new expert joins a CMAS, the fist step to include it into the system is to estimate and describe it in an expert ontology.

There are different ways to estimate machine and human experts. For an agent expert, the estimation is achieved through some data mining or AI methods (Fu 1999, Kovalerchunk et al 1997). Estimating a human expert could be achieved through interviews or surveys. Some CMAS applications take expert estimation as the only step to discover experts' knowledge. However, in this approach, expert estimation is the preliminary step to include a new expect into the system. This process is to draw an overall image of the new expert and find some related knowledge and category for it.

4.2 Task Processing and Ontology Update

In this approach, expert mining is a "life time" process. It will be processed as incoming tasks are solved by experts of the system (see Figure 6). Here, we assume that all incoming tasks/problems can be described as the required knowledge representation format of the system. Then, the CMAS will be able to map it with the common ontology to the system. The ontology mapping is to put the task in proper categories and explore related knowledge in the ontology. It can have three possible results:

a. The task can be mapped and the solution for the task can be found in the current common ontology. In this case, the incoming task might be achieved in the system before, and current common ontology is sufficient to provide solution. The common ontology and knowledge will not be refreshed.

b. The task can be mapped but the solution cannot be found in the current common ontology. Normally, this situation occurs when same kind of tasks have been solved in the CMAS, but the task is different with previous (solved) tasks. Another possibility is that experts in the system have solved this kind of problems before, but they did not publish their solutions in the system. In this case, the CMAS will allocate experts that are in the same category of the task to solve the task. After the task has been solved, the system will modify the experience value of the expert in the common ontology.

c. The task cannot be mapped in current common ontology, which means that the task has never been done by the system, and there is no suitable category can be found. In this case, the system will broadcast the task description to all experts of the system and see whether there exist any experts who can solve this task. If no expert gives response, the task will be rejected by the system. If there is an expert who can do the task, the system will allocate the task to it. After the task is solved, the system will set up links between the task and the expert in the common ontology. Also, the expert can publish its expertise and knowledge in the system.

4.3 Experts' Experience

In a CMAS, there may have more than one experts in the same area. These experts may have conflicts on their special knowledge, expertises and solutions. In this case, the system will choose the expert with higher experience value. The experience value can be considered as the reputation of experts. It is calculated according to the previous task execution result of the expert. An expert's experience value could be increased or decreased depending on the performance of a task.

Another case that the system may need to estimate the reputation of an expert is when the expert wants to refresh the common knowledge of the system. In this case, the system needs to evaluate whether the expert is specialised in that area through estimating its experience value. If the value is greater than the specialist threshold, the system will allow it to publish the knowledge. If the value is medium, the system will find out other experts in the area and collect suggestion from them. The knowledge refresh request will be rejected if its experience value is too low.

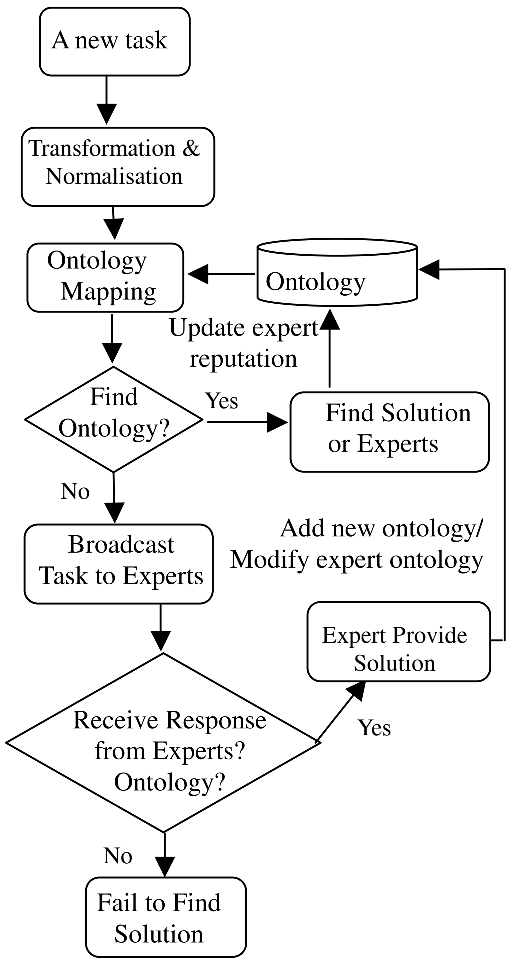

Figure 6 Ontology-based task processing

4.4 Example

In this subsection, we give a simple example to illustrate the expert mining processes in this approach. Suppose that there is a CMAS with four experts, i.e. *ExpA, ExpB ExpC* and *ExpD*.

Expert ontologies of the four experts are shown in Figure 7. From Figure 7, it can be seen that the specialist areas of *ExpA, ExpB, ExpC* and *ExpD* are gynecologist, paediatrician, gynecologist and neurologist, respectively. We also suppose that there are two new tasks *tA* and *tB* come into the CMAS. The required knowledge and expertise area of *tA* is gynaecology, but the required knowledge and expertise area *tB* is unknown.

Since the specialist area of *ExpA* and *ExpC* is matched with *tA*, *ExpA* and *ExpC* are potential experts to solve *tA*. According to the experience value of *ExpA* and *ExpB* (See Figure 7), *ExpA* has a higher experience than *ExpC*. Hence, *ExpA* is selected to handle *tA*. Then, if *tA* is solved by *ExpA* successfully, the experience value of *ExpA* will be increased. Otherwise, the experience value of *ExpA* will be decreased.

Since the required knowledge area of *tB* is unknown, we cannot find a suitable expert to solve *tB* directly. In this situation, *tB* will be broadcasted to all experts of the system. Then, all experts of the CMAS will evaluate whether they can solve *tB*, and give response to indicate whether they can accept it. Here we suppose that only *ExpB* give a positive response to accept *tB*. In this case, *tB* will be allocated to *ExpB*. After *ExpB* solves *tB*, the expert ontology of *ExpB* will be modified and *tB* will be added to its specialist area with an initial experience value (i.e. experience = 1).

From this example, it can be seen that by using expert ontologies to describe experts of a CMAS, we can easily find suitable experts to handle incoming tasks. With an incoming task, we can know the potential experts that can achieve the task, according to the specialist areas of experts (which are described in expert ontologies). Then, according to the experience values of these experts, we can assign the task to the most suitable expert. In addition, expert ontologies are updated as each task is processed in a CMAS. This feature makes CMASs more suitable for dynamic working environments. In this approach, the task handling process is also the expert mining process. As processing incoming tasks, unknown knowledge of experts can be discovered.

Expert name: ExpA;
Specialist Area: gynecologist;
Expertise Knowledge: gynaecology;
Experience: 58;

Expert name: ExpB;
Specialist Area: paediatrician;
Expertise Knowledge: pedology;
Experience: 60;

Expert name: ExpC;
Specialist Area: gynecologist;
Expertise Knowledge: gynaecology;
Experience: 56;

Expert name: ExpD;
Specialist Area: neurologist;
Expertise Knowledge: neurology;
Experience: 43;

Figure 7 Ontology-based task processing

5. Discussion and Conclusion

A CMAS is a system that contains hybrid agents including human agents and autonomous software agents. Agents in a CMAS have strong learning abilities and possess expertises that may be described in different formats. These features bring challenges to knowledge management and discovery in CMASs. Some current CMAS applications hire ontologies to facilitate knowledge management in the system. However, most current CMASs lack of a mechanism to discover expert knowledge and expertise from expert agents. This limitation makes expert agents in a CMAS not been well utilised.

In this paper, we introduce an ontology based approach for expert discovery and knowledge mining in CMASs. This approach uses ontology to manage and structure the domain knowledge of a CMAS and embeds expert ontologies into common ontologies of the system. The contribution of this approach can improve knowledge mining and expert discovery in a CMAS from the following aspects:

1. The including of expert ontologies brings conveniences for task allocation and knowledge discovery. Through an expert's position in the common ontology, we can find a more effective and pertinent way to achieve expert mining.
2. In this approach, knowledge mining is a "life time" process that is executed as incoming tasks which are solved by experts. Related ontologies of experts of a system will be updated after a task is processed. The "life time" knowledge mining process keeps the descriptions of experts accurate. This improves the adaptability and flexibility of a CMAS in dynamic working environments.
3. In this approach, "expert experience" is included in expert ontologies. This brings conveniences for handling solution conflicts and can facilitate a CMAS to find suitable experts to execute incoming tasks.

In summary, the ontology-based approach promises more self-learning and self-adjusting abilities to CMASs, and is more suitable for CMASs with high-ability heterogeneous experts and open environments.

Acknowledgement

Authors would like to greatly appreciate anonymous referees for their valuable comments and suggestions in helping to improve the quality of this paper.

References

[1] Artikis, A. & Pitt, J. (2001). A formal model of open agent societies. In: Proceedings of the 5th International Conference on Autonomous Agents, pp. 192-193, Montreal, May 28 – June 01, 2001, ACM Press

[2] Bai, Q. & Zhang, M. (2004). Agent coordination through knowledge management. International Journal of Knowledge and Systems Sciences, 1(1): 45-52

[3] Bai, Q. & Zhang, M. (2005). Dynamic team forming in self-interested multi-agent systems. In: Proceedings of the 18th ACS Australian Joint Conference on Artificial Intelligence, pp. 674-683, Sydney, December 5-9, 2005, Springer-Verlag

[4] Bai, Q., Zhang, M. & Ren, F. (2007). A Coloured Petri Net based approach for flexible agent interactions. In: Proceedings of the 4th International Conference in IT and Application, pp. 186-192, Harbin, January 15-18, 2007

[5] Bai, Q. & Zhang, M. (2006). Coordinating agent interactions under open environments. In: Fulcher, J. (eds), Advances in Applied Artificial Intelligence, pp. 52-68, Idea Group Publishing

[6] Busetta, P., Ronnquist, R., Hodgson, A. & Lucas, A. (2007). JACK intelligent agent – components for intelligent agents in Java. Technical Report TR9901m Agent Oriented Software Pty. Ltd. http://www.jackagents.com/pdf/tr9901.pdf. Cited May 03, 2007

[7] Fensel, D., van Harmelen F., Horrocks, I., McGuinness, D. & Patel-Schneider, P. (2001). OIL: An Ontology Infrastructure for the Semantic Web. IEEE Intelligent Systems, 16(2): 38-44

[8] FIPA-OS. (2007). Available via SourceForge, http://sourceforge.net/projects/fipa-os/. Cited May 03, 2007

[9] Finin, T., McKay, D., Fritzson, R. & McEntire, R. (1994). KQML: an information and knowledge exchange protocol. Knowledge Building and Knowledge Sharing, Ohmsha and IOS Press

[10] Fu, L. (1999). Knowledge discovery based on neural networks. Communications of ACM, 42(11): 47-50

[11] Genesereth, M. & Fikes, R. (1992). Knowledge interchange format, version 3.0 reference manual. Computer Science Department, Stanford University, logic-92-1, Available via http://ksl.stanford.edu/knowledge-sharing/kif/. Cited May 03, 2007

[12] Gruber, T. R. (1991). The role of common ontology in achieving sharable, reusable knowledge bases. In: Principles of Knowledge Pepresentation and Reasoning - Proceedings of the Second International Conference, pp. 601-602, Cambridge, 1991

[13] Huhns, M. & Stephens, L. (1999). Multiagent systems and societies of agents.In Weiss G. (eds.), Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence, pp. 80-120, MIT Press

[14] Kovalerchuk, B., Vityaev, E. & Ruiz, J. (2001). Consistent and complete data and “Expert” mining in medicine. Studies in Fuzziness and Soft Computing, 60: 238-281

[15] Kovalerchunk, B., Triantaphyllou, E., Ruiz, J. & Clayton, J. (1997). Fuzzy logic in computer-aided breast cancer diagnosis: analysis of lobulation. Artificial Intelligence in Medicine, 11: 75-85

[16] Labrou, Y., Finin, T. & Peng, Y. (1999). Agent communication languages: the current landscape. IEEE Intelligent Systems, 14(2): 45-52

[17] Lesser, V. (1999). Cooperative multiagent systems: a personal view of the state of the art. IEEE Transactions on Knowledge and Data Engineering, 11(1): 133-142

[18] Lesser, V. (1998). Reflections on the nature of multi-agent coordination and its implications for an agent architecture. Autonomous Agents and Multi-agent Systems, 1(1): 89-111

[19] Nwana, H., Ndumu, D. Lee, L. & Collis, J. (1999). ZEUS: a toolkit and approach for building distributed multi-agent systems. In: Proceedings of the 3rd International Conference on Autonomous Agents, pp. 360-361, Seattle, May 01-05, 1999

[20] Zhang, M. & Li, W. (2000). DynaInteg: meta-ontology supporting dynamic knowledge sharing and acquiring for multi-agent cooperation. In: Proceedings of 9th International Conference on Intelligent Systems, pp. 47-51, Louisville, January 15-16, 2000

Minjie Zhang received her Bachelor of Science (in Computer Science) from Fudan University, China in 1982, and her PhD degree from the University of New England, Australia in 1996. Currently she is an associate professor in the School of Computer Science and Software Engineering at University of Wollongong, Australia. She was a co-chair for the First Pacific Rim International Workshop on Electronic Commerce in 2006, and an organization chair, for the International Workshop on Rational, Robust, and Secure Negotiation Mechanisms in Multi-Agent Systems in 2005, 2006 and 2007 and a tutorial chair of the International Conference on Complex Open Distributed Systems in 2007. She is a member of editorial board of System and Information Sciences Notes and the International Journal of Knowledge and Systems Sciences. Dr Minjie Zhang is an active researcher and published over ninety research papers. She is the chief investigator for more than 10 different research grants including an ARC (Australia Research Council) Discovery grant and an ARC International Linkage Grant. She is a member of IEEE and the International Association of Computer and Their Applications. Her research interests include multi-agent systems, distributed information retrieval, agent-based modeling and simulation, and data mining and knowledge discovery.

Xijin Tang acquired her bachelor, master and doctoral degrees in 1989, 1992, and 1995 respectively. Now she is an associate professor in Key Laboratory of Management, Decision-making and Information Systems, Chinese Academy of Sciences. She has been visited Georgia Institute of Technology as research associate during 1998-1999 and Japan Advanced Institute of Science and Technology as a visiting associate professor of "Fujitsu Chair for Science of Complex Systems" during 2000-2001. She has engaged in research on decision support systems and system methodologies, and developed a variety of computerized support systems for water resources management, comprehensive evaluation on commercial information systems, naval weapon system evaluation, etc. during 1992-1998 and devoted to a major NSFC project on meta-synthesis research during 1999-2004. Her current interests include knowledge science, complex system modeling, social network analysis, creativity and decision support systems, etc. which aim to provide effective and intensive knowledge support for unstructured problem solving. Recently she has published two books on meta-synthesis system and the oriental Wu-li Shi-li Ren-li system approach. She has organized 7 international workshops related to meta-synthesis and knowledge science since 2001. She is a council member of Systems Engineering Society of China and of the International Society for Knowledge and Systems Sciences.

Quan Bai received his bachelor degree in 2001 from Tianjin University, China, and acquired his Ph.D degree from the University of Wollongong, Australia in 2007. Currently, he is a research fellow in the School of Computer Science and Software Engineering at the University of Wollongong. He has involved in three research projects and published more than fifteen research papers during the period of his PhD study. Dr Quan Bai's research interests include multi-agent coordination, agent based simulation and modeling, data mining and social network analysis.

Jifa Gu, Ph.D., a Professor in Institute of Systems Science, Academy of Mathematics and Systems Sciences, CAS, Beijing, China, His fields of study includes Operations Research and Systems Science. He is a Vice- president of International Federation for Systems Research, and a Vice- president of International Society of Knowledge and Systems Science. He obtained ten awards from Chinese Academy of Sciences and other state organizations.