B-021

Repository-based Design Method of Cooperation Protocol for Multiagent System

Wenpeng Wei[†] Uchiya Takahiro[‡] Tetsuo Kinoshita[†,††]

1. Introduction

Agent technology has been proved very effective for the development of distributed and complicated information systems. To support the development of Multi-Agent System (MAS), a number of methodologies and frameworks have been proposed. However, the development of MAS remains difficult due to the complexity of the system and the differences between agents and objects. To improve that situation, repository-based agent framework (ADIPS/DASH)[1] has been proposed. DASH is designed for agent with the essence of a software agent to keep itself simple. Further more, the repository mechanism supports the reuse of software agent which makes it easier to develop MAS. A case study[2] of comparison of MAS development in DASH and JADE[3] has been done to show that for some certain system developing in DASH is much more easier and faster than in JADE. The existing software reuse mechanism in DASH called Rule Set could reuse agent behavior knowledge (rules) at code level but requiring high skills and experiences of DASH agent programming which is difficult for the non-expert developers due to the lack of easy-to-use, effective development methodology for DASH. Address to that problem, this paper proposes the repository-based design method of cooperation protocol for multiagent system to support the MAS development in DASH.

2. Related Works

A number of MAS methodologies and frameworks has been proposed. But there are few methodology covered the implementation phase of the development lifecycle. And the researches of comparing agent-oriented methodologies[4][5][6] have been done. One of the problems is a lot of the methodologies are limited in the agent implementation platform which means they could not be used in DASH.

In the other hand, common MAS development in DASH could be divided in three different parts which are knowledge design, protocol design and organization design. The existing MAS development support functions could support the software reuse at code level. We focuses on the frequently performed protocol design to provide support function for software reuse at design level.

3. Proposal

To support the protocol design and to improve the reuse of software component, we introduce a repository-based design method of cooperation protocol for multiagent system. The proposed methodology could help the non-expert designer to finish protocol design based on the protocol library in the repository. And for the expert designer who wish to design brand new protocols, the proposed method provide a 3-step guide to support the design and development of the new protocols. Furthermore, we also implemented supporting tools for the proposed design method that could effectively reduce the coding work of the implementation. Figure 1 shows the overview of the proposal which would be explained in details later.

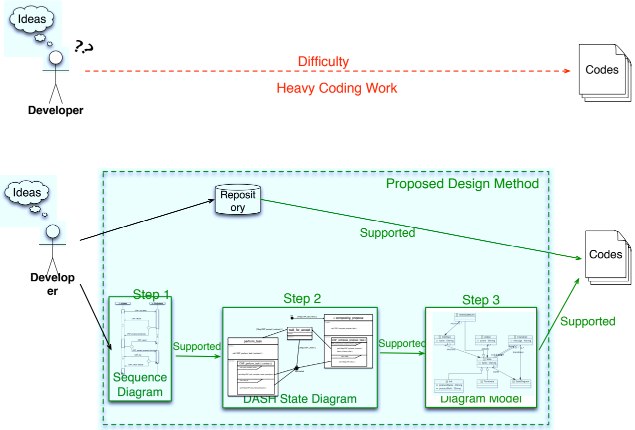

Figure 1 Overview of the Proposed Method

For the non-expert designer, the proposed method would guide the designer to find a appropriate protocol template in the repository and help the designer to customize the protocol template to meet the requirement. Supporting tools for the non-expert designer could give a list of protocol template to support the designer to choose, and furthermore, could automatically generate the necessary codes and instructions to help the designer to finish the protocol design. By taking the advantage of the repository mechanism, even the non-expert designer could design agent cooperation protocol effectively based on the software reuse principle.

In the other hand, for the expert designer, the proposed method provides three steps workflow to guide the protocol design.

The first step is the Role and Sequence Design in which the roles and message sequences of the protocol template would be decided by developer using common UML sequence diagrams with specific notations for the template characters. We introduce a specific nations for DASH protocol templates called interface. The interface is just like the definition of functions, which are decided by the protocol designers in ways of the names and the parameters. The body of the interfaces would be left blank to the protocol users to finish the inner actions to meet their application domain specific requirements.

In the second step which called State Design, the DASH State Diagram should be designed for each role of the

[†] Graduate School of Information Sciences, Tohoku University

[‡] Nagoya Institute of Technology

[††] Research Institute of Electrical Communication, Tohoku University

protocol template depend on the sequence diagram which has been designed in the last step. The DASH State Diagram is formed by six different basic components as shown in Figure 2.

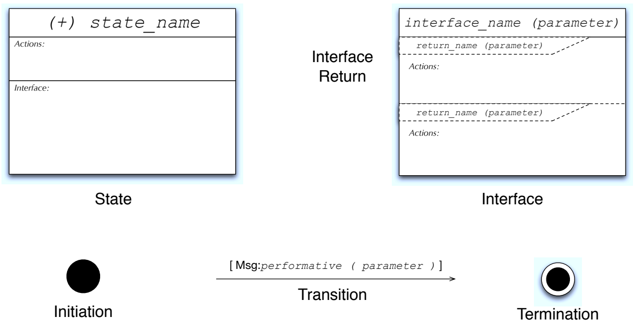

Figure 2 DASH State Diagram Componments

The developers could use the components to construct any kind of DASH State Diagram to describe the state transitions of the roles of the protocol templates. To support the DASH State Diagram design, we developed a graphical editing tool which shown in Figure 3. Once this step has been finished, the developer could implement the designed transitions in the next step of Implementation.

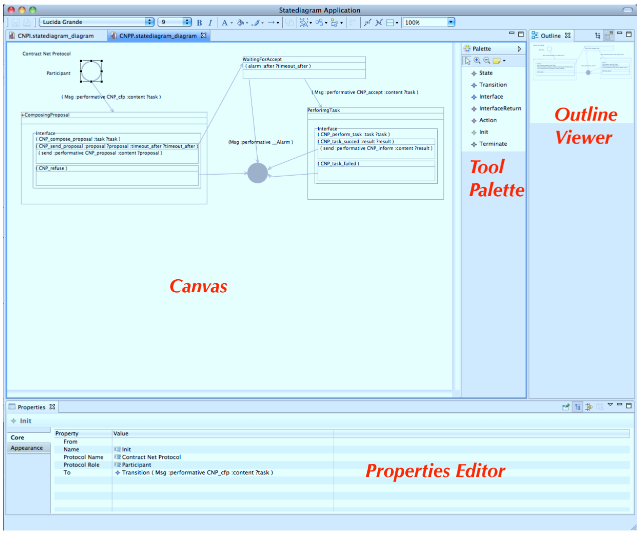

Figure 3 DASH State Diagram Editor

In the last step of Implementation, the graphcial editing tool could automatic generate the DASH rule-set code of the designed protocol template according to the DASH State Diagram. The designer could use the template directly after the auto-generation.

By following the workflow above, the expert designer could use the supporting tools to design cooperation protocol easily and effectively.

4. Experiments and Evaluation

To evaluate the proposed method, we have applied it to the development of multi-agent microgrid control system[7][8]. Firstly, the Contract Net Protocol (CNP) template had be designed and implemented using the proposed design method and supporting tool. Then the system which had been designed and implemented before this research had been redesign and reimplemented using the generated Contract Net Protocol template. Since the system had been developed before this research, to evaluate the coding work reducing effect of the proposed design method, the comparison of lines of code between the old implementation and proposed design method has been performed. The comparison results of different type of agent(Ag.) in the system such as Micro Grid Operation Control Center(MGOCC) and Distributed Generator(DG) are shown in Table 1.

Table 1 The result of the comparison

	MGOCC Ag.	LOAD Ag.	DG Ag.
Old implementation	298 lines	114 lines	136 lines
Proposed method	129 lines	58 lines	62 lines
Coding work reduction	56.7%	49.1%	54.4%

According to the result table, about half coding work has been reduced using the proposed design method comparing to the old implementation.

5. Conclusion

To support the MAS development in DASH, a repository-based design method of cooperation protocol for multi-agent system has been proposed. Both the non-expert and expert designer would be supported to design and develop agent cooperation protocol easily and effectively by the proposed method and supporting tools. An evaluation has been done by apply the proposed method to the development of a MAS application and the result shown the proposed method effectively reduced almost half of the coding work of MAS application development.

参考文献

[1] T. Uchiya, T. Katoh, T. Suganuma, T. Kinoshita, N. Shiratori. An architecture of agent repository for adaptive multiagent system. ICOIN 2002, LNCS 2344, pp. 802—813, 2002.

[2] R. Miura, W. Wei, A. Takahashi, T. Kinoshita. Using Language Grid services based on Active Information Resource. IEICE technical report. Artificial intelligence and knowledge-based processing 108(441), 25-30, 2009-02-20.

[3] Fabio Bellifemine, Agostino Poggi, and Giovanni Rimassa. JADE: a FIPA2000 compliant agent development environment. In Proceedings of the fifth international conference on Autonomous agents (AGENTS '01). ACM, New York, NY, USA, 216-217. 2001.

[4] A. Ghose, G. Governatori, and R. Sadananda (Eds.): Towards Method Engineering for Multi-Agent Systems: A Validation of a Generic MAS Metamodel, PRIMA 2007, LNAI 5044, pp. 255—267, 2009.

[5] B. Apolloni et al. (Eds.): A Comparison of Three Agent-Oriented Software Development Methodologies: ROADMAP, Prometheus, and MaSE, 2007/ WIRN 2007, Part III, LNAI 4694, pp. 909—916, 2007.

[6] P. Giorgini et al. (Eds.):Comparing Agent-Oriented Methodologies, AOIS 2003, LNAI 3030, pp. 78—93, 2004.

[7] Hak-Man Kim and Tetsuo Kinoshita. A New Challenge of Microgrid Operation. Communications in Computer and Information Science, 2010, Volume 78, 250-260.

[8] Hak-Man Kim, Wenpeng Wei and Tetsuo Kinoshita. A New Modified CNP for Autonomous Microgrid Operation Based on Multiagent System. Journal of Electrical Engineering and Technology Vol. 6, No. 1, pp. 139 146, 2011