Study on the Multiagent Based Model System and Model Integration*

Daiping Hu[1], Ruiming Wu[1], Mengyu Li[2], Qi Wang[3]

[1] Antai College of Economics & Management, Shanghai Jiaotong University, 200052 Shanghai, China
{dphu, dwurm }@sjtu.edu.cn
[2]Shanghai Xiandai Vocational & Technical School
Huayan Road 112, 200042 Shanghai, China
dmyli@163.com
[3] Graduate School, Lanzhou University, Lanzhou, 730000 Gansu, China
qiwanglzu@126.com

Abstract. Model integration is to combine some simple models togather for decision making of complex problems. But the model integration is difficult to implement in the traditional DSS. In this paper, we propose a multiagent based model system and study the method of model integration using agents. The multiagernt based model system includes a management agent, many interface agents and a lot of model agents. Every model is encapsulated by an agent, which can be invoked by users utilizing interface agents or other agents. The model integration method is implemented by using the cooperation of model agents. We also study a case to demonstrate the applicaton of model integration using the multiagent based model system we developed.

Keywords: Model system, model integration, multiagent system

1 Introduction

A model is a simplified representation or abstraction of reality. As reality is often too complex to copy exactly or much of the complexity is actually irrelevant in solving the specific problem, simplification is necessary. According to the degree of abstraction, models are classified into three groups: iconic, analog, and mathematical [1]. An iconic model is a physical replica of a system, usually on a different scale from the original. An analog model is a symbolic representation of reality. Representation of the complexity of relationships in many organizational systems can not be done by using icon or analog models. The most abstract model called mathematical model is often used to describe them mathematically.

We build mathematical models to represent for decision making of complex problems. Computers with the capability of speed calculation are applied in modeling. The decision support system (DSS) as a kind of computer based information system

* Supported by National Nature Science Foundation of China (No.70671067) and Youth Research Fund of Antai College of Economics & Management

© A. Gelbukh, A. Kuri (Eds.)
Advances in Artificial Intelligence and Applications
Research in Computer Science 32, 2007, pp. 141–149

Received 18/06/0"
Accepted 31/08/07
Final version 01/10/07

(CBIS) has been studied and developed to support decision making for solving complex problems. A DSS uses data and models, provides easy user interface, and can incorporate the decision maker's own insights. Architecture of a DSS typically includes: data management subsystem, model management subsystem, knowledge management subsystem and user interface subsystem. The model management subsystem is the most important part, which is composed of model base, model base management system, modeling language, model directory, model execution, integration, and command processor.

Models are most important components of a DSS. Model integration provides the method of using two or more models or creates a new model by combining them. Integrated modeling environment should be provided in model subsystem. Some progresses have been made on model integration and some modeling integration platforms have been developed. [2].

Recent years, as achievements of research on the theories, methodologies, architectures, languages of multiagent system (MAS) accumulated, the applications of agents emerge out more and more. Multiagent based modeling applied in complex system is mainly the multiagent based simulation, a bottom-up modeling method. MAS based system as tools for dealing with complex problems is applied in the design paradigm of building agent based DSS, hall of workshop for meta-synthetic engineering (HWME) [3,4,5].

In this paper, we study the multiagent based model system and the methods of model integration. This paper is organized as follows: In section 2, methods of model integration are discussed. Section 3, multiagent based integrated modeling platform is presented. In section 4, we introduce an application about integration of forecasting models and a linear programming model. In section 5, conclusions and discussion are given.

2 Methods of Model Integration

Many researchers have studied on model integration and have made some achievements [2]. Based of their study, model integration can be classified in three ways.

(1) Scheme integration and process integration [6]. Schema integration is the task of merging internal structure of two models to create a new model. Process integration is about solver control.

(2) Definitional integration and procedural integration [6]. Definitional and procedural are in accord with the classical dichotomy of programming languages. Definitional integration is about mode representation and corresponds to scheme integration. Procedural integration is about model manipulation and corresponds to process integration.

(3) Deep integration and functional integration [7]. Deep integration is combining two or more given models together to produce a single new model, subjects to the important qualification that the new model must be represented in same definitional formalism as the given models. Functional integration does not yield a new model. It superimposes a computational agenda for coordinating calculations over the original

models by directing certain models' outputs to other models' inputs while specifying the order of computations.

To implement functional model integration by using agents, we analyze the linkage of models. Two basic and an extended modes of connections are summarized. Two basic modes are pipeline connection and parallel connections.

In pipeline connections of models, the outputs of one model are the inputs to another. The later cannot begin modeling until the previous model gives the outputs. See Fig.1.

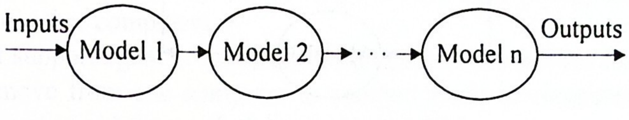

Fig. 1. Pipeline connections of models

In parallel connections of models, inputs of all the models are from the same information sources. Their inputs may be the same or may be different. Generally, if all the models can model the problem individually, their inputs may be the same. If each of the models can model a different part of the given problem, its inputs may be different. See Fig.2.

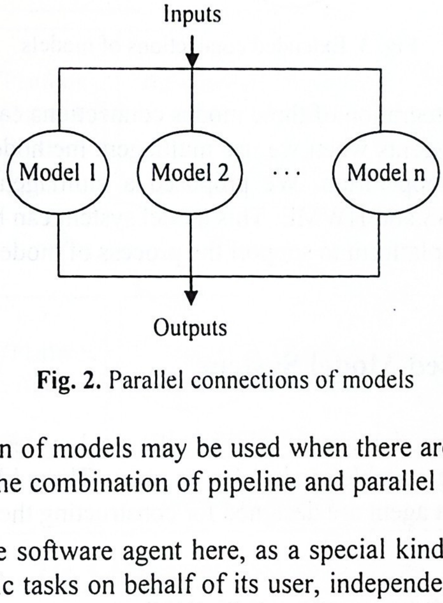

Fig. 2. Parallel connections of models

Extended connection of models may be used when there are three or more models to be integrated. It is the combination of pipeline and parallel connections of models. See Fig.3.

Agent, we mean the software agent here, as a special kind of computer program, can accomplish specific tasks on behalf of its user, independently or with little guidance. Architecture of an agent should include task processing method, knowledge, and the capabilities of task processing, learning, reasoning, decision-making, communication, cooperation etc.

A MAS is a system comprised of some agents, and all the agents can communicate with each other and for cooperation in procedure of performing tasks. According to whether the MAS if pursuing a specific system goal or not, two types of MASs can be distinguished: The specific goal and non-specific goal. A specific MAS has a specific goal, all agents in the system serve for the system goal, and each agent performs different tasks and can cooperate with others. Multiagent based DSS is a specific goal MAS. Non-specific goal MAS has not any specific system goal, and all the agents in the system have themselves goals, each agent running for pursuing itself goal. There may be competitions and benefit conflicts among them. Multiagent based simulation system is a typical non-specific MAS.

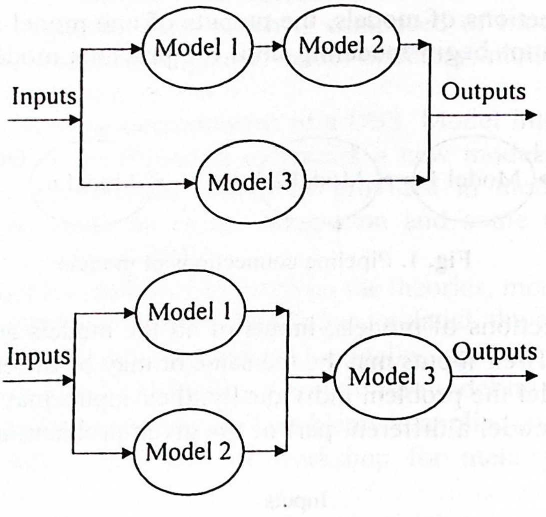

Fig. 3. Extended connections of models

Functional model integration of three modes connections can be implemented with use of cooperation of agents when we use multiagent methodology. Connections are implemented by the cooperation. We proposed a multiagent based model system which can be bused DSS and HWME. This model system can be an integrated modeling environment and a platform to support the process of model integration.

3 Multiagent Based Model System

The multiagent based model system is a specific goal system. All agents sever the same goal of providing suitable models for its users. Three kinds of working agents and one kind of support agent are designed for constructing the model system.

3.1 Components of Multiagent Based Model System

Three kinds of working agents are: management agent, interface agent and model agent.

(1) The management agent. Responsible for system management, it keeps all the information, such as the names, addresses, communication ports of agents and provides the functions of registration, updating, querying of the information for agents in the system.

(2) The interface agent. It is built for its user to interact with the model system. Every user has an interface agent. An interface agent has GUI for its user or modeler

to access data, query model information and call models. When more than one mod are used, a modeler can define the connection as pipeline, parallel or extended mod simply in graphic operating way such as selection and drag & drop.

(3) The model agent. Developed by its owner, is the main component of mod system to perform modeling and computing. Each model agent encapsulates the alg rithms of a model, which can be called by the users using their interface agents.

A multiagent based model system contains one management agent; many interfac agents for each user; many platform agents for each computer and many model agen For sake of reliability and efficiency of system, some agents may have their copi and can move to other computers.

One kind of support agent is called platform agent. It is platform for working age to be able to move from one computer to another. Each of computers in a netwo allowing the agents reside has a platform agent installed.

In multiagent based model system, all kinds of agents are cooperating for modeli in the procedure of problem solving. Cooperation of model agents can realize th model integration easily. Fig.4 shows the multiagent based model system.

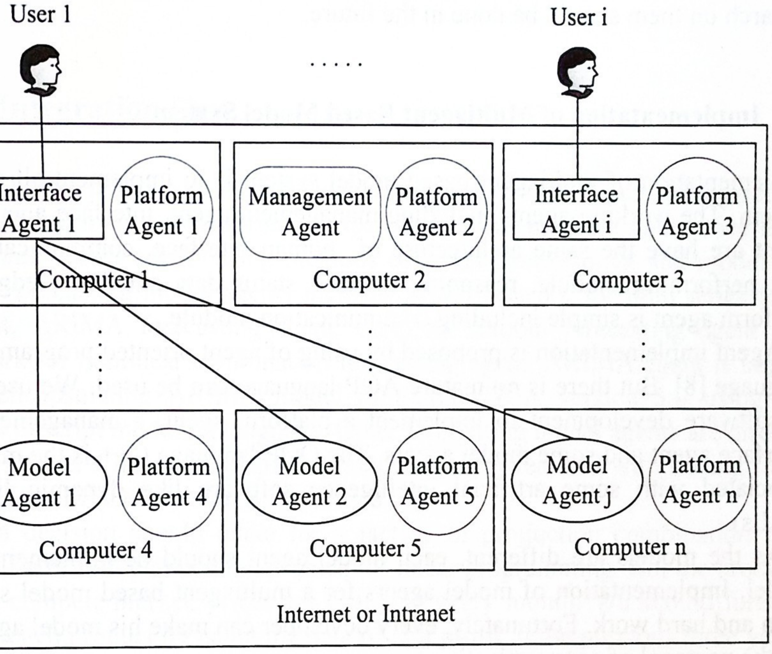

Fig. 4. Multiagent based model system

3.2 Procedure of Modeling

The procedure of modeling in manual can be described as follows:

Access or input data if it is necessary for a problem solving, such as forecasting time series forecasting, the series data in the past are absolutely necessary.

Query information of model agent in the system, and make sure that all the model agents you want to use are there in existence. One model agent may contain more than one functions, all functions will be showed. As an ARIMA model agent, the main function is time series forecasting, and functions of calculating SAC and SPAC are still provided.

Select the model agents by dragging and dropping them to the model integration window. Specify connections of models or selected functions of model agents by drawing lines between them to define the model integration in a graphic operating interface.

Start the modeling, the interface agent will call all the required model agents to work. Modeler may be prompted with some information and be asked to input some parameters if those models require them in the modeling process.

All model agents have finished their specified work, and the final results will be sent to the interface agent of the modeler.

Interface agent displays the results for the modeler.

Automatic model integration can be done if the interface agent has the ability of problem decomposition and the model agents are capable of modeling actively. Problem decomposition rules and model integration rules are necessary for agents. Farther research on them should be done in the future.

3.3 Implementation of Multiagent Based Model System

Implementation of multiagent based model system is to implement all agents in the system. The working agents including management agent, interface agent and model agent are have the same architecture of human interface, communication module, task performing module, reasoning module, status data and knowledge base. The platform agent is simple including communication module.

Agent implementation is proposed by using of agent oriented programming (AOP) language [8]. But there is no mature AOP language can be used. We use the method of software development to implement a platform agent, a management agent, an interface agent and some model agents. The OOP language C++ is the main tool, and associated with some artificial intelligence software like dynamic link libraries (DLLs).

As the models are different, each model agent should be implemented for each model. Implementation of model agents for a multiagent based model system is the main and hard work. Fortunately, every developer can make his model agents observing the protocol of communication.

Besides the direct development of model agent by coding it with the OOP languages, model encapsulation method we proposed is another choice for implementation of model agents. We developed an agent shell (showed as Fig.5) with most capabilities of a common agent to encapsulate models of some model software, statistics software such as the STATISTICA, SAS, and linear programming software like LINDO. The calculation of modeling task is accomplished by software; other things like the communication are done by the agent shell. Unfortunately, not all the model software can be capsulated by agent shell, only those support to execute modeling descript language or provide DLLs to call are within the bounds of possibility. Though the movement of the agent with model software encapsulated is not as agile as the directly developed model agent, the method of model encapsulation quickens the development of model agents.

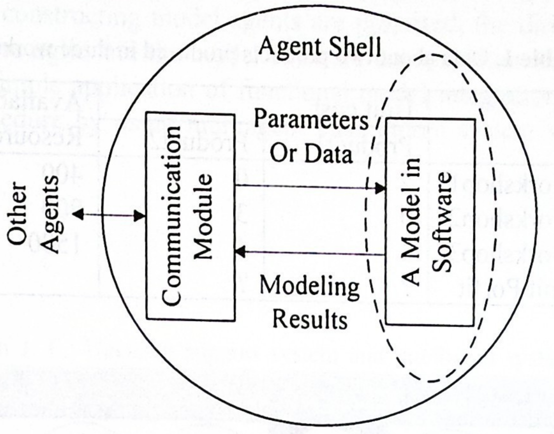

Fig. 5. Model encapsulation by agent shell

4. Model Integration Case

A multiagent based model system to support model integration has been developed by us. Besides management agent, some interface agents and platform agents, we have developed some model agents using methods of direct development and model encapsulation. Forecasting model agents includes neural network for time series forecasting agent, ARIMA forecasting agent, exponent smooth forecasting agent and neural network for nonlinear combination forecasting agent. ARIMA agent is implemented by encapsulating ARIMA model of STATSTICA version 5.0 using the STATSTICA descript modeling language. Other forecasting model agents are directly developed. There is a linear programming model agent is realized with calling functions in DLLs the LINDO version 6.1 provided.

There is a decision should make for a factory of production combination next month. That is a resource-allocation problem of linear programming. But the products are sold in a compete market, prices are changing every month. We should forecast unit profit of each product by using the data of unit profit in the previous three years. Therefore, to solve this problem, forecasting model and linear programming model must be used. This is an example of application for model integration that can demonstrate the procedure of model integration using agents in multiagent based model system.

We define the integration of models in interface agent graphically. For each of unit profit of two products, three time series forecasting models including NN time series agent, ARIMA agent, exponent smooth agent are connected in parallel, and then connect to NN combination agent. Linear programming agent is used to get the solution for problem of production combination. Fig.6 shows the model integration relationship. Unit profit forecasts of two products are 324.68 and 536.38. Then the forecasts and other data like unit cost, available resources are passed to linear programming agent, final solution of this problem is gotten: 100 for product1 and 300 for product2.

Table 1. Data about two products produced in three workshops

	Unit cost		Available
	Product1	Product2	Resource
Workshop1	2	0	400
Workshop2	0	3	900
Workshop3	3	4	1500
Unit Porfit	?	?	

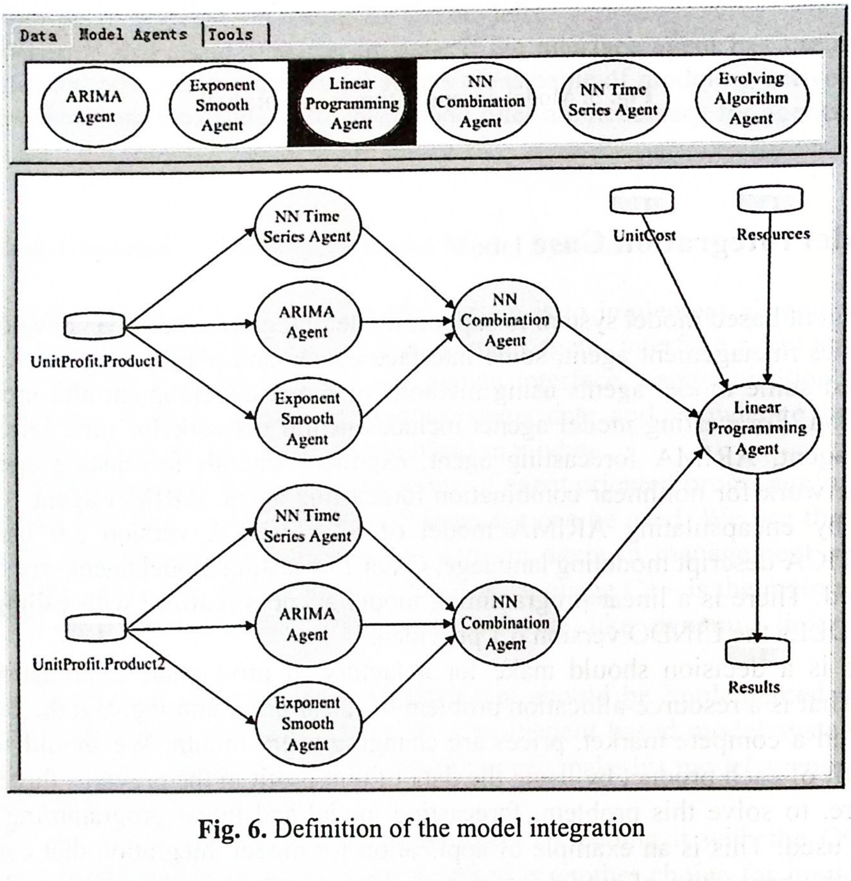

Fig. 6. Definition of the model integration

5 Conclusions

In this paper, we mainly studied the functional model integration using agents in an agent based integrated modeling platform we developed – the multiagent based model system. Connections of model are classified as two basic types: the pipeline and the parallel, which can extend a complex linkage among models. Integration of model can be realized with the cooperation of agents in multiagent based model system.

Two methods of constructing model agents are proposed, the direct development makes the agent more agile and the model software encapsulation lets the agent developing faster. A simple application of functional model integration to demonstrate the integration procedure by using multiagent base model system we developed is presented.

References

1. Turban E., Aronson J. E.: Decision support system and intelligent system. Prentice Hall, Upper Saddle River. NJ (1999)
2. Xijin T., Jifa G. and Nakamori Y.: Model integration for system modeling. Proceedings of 2001 international symposium on knowledge and systems sciences, (2001): 285-296
3. Tung B., Jintae L.: An agent-based framework for building decision support systems. Decision Support System, 25(1999):225-237
4. Daiping H., Huanchen W., A multiagent framework for building workshop for hall of meta-synthetic engineering. Proceedings of 2001 International Symposium on Knowledge and Systems Sciences, (2001): 161-168
5. Xuesen Q., Jingyuan Y., Ruiwei D.: A new discipline of science: the study of open complex giant system and its methodology. Chinese journal of systems engineering and electronics, 4(1993): 2-12
6. Dolk D. R., Kottenmann J. E.: Model integration and a theory of models. Decision support systems, 19(1993): 51-63
7. Geoffrion A. M.: Structured modeling: Surveys and future research directions. Interactive Transactions of ORMS, 1(1996)
8. Shoham Y.: Agent-oriented Programming, Artificial Intelligence, 60 (1993): 51-92