An Intelligent Assistant Agent in a Cooperative Distance Learning Environment

Li-Ping Kuo, Chi-Wei Hung, Maiga Chang and Jia-Sheng Heh

Dept. of Info and Comp. Eng., Chung Yuan Christian University, Chung Li, 32023, Taiwan
Tel: 886-3-4563171 ext. 4719 E-mail: jsheh@cycs01.ice.cycu.edu.tw

Abstract

This paper proposes a methodology of applying *Intelligent Assistant Agents* to th distance learning environment, especially a *virtual experiment environment*. This kind o *intelligent assistant agent* can play different roles in the CAI systems based on both characteristics of intelligent and computer software. In order to tutor the learners during learning, the seven stages of problem solving cycle should be satisfied. To achieve the goal of roles that an assistant agent play, several tools and a script are also necessary designed in this paper.

Keywords: *KQML, virtual experiment environment, problem solving, intelligent agents, distance learning*

1. Introduction

The most benefit of a *computer-aided education (CAE)* system is it can prompt learners some feedback and responses immediately in appropriate ways during learning process. [DiL92][FuL92] [Chan95] But most of this kinds of systems still can not teach learners specific knowledge, evaluate what learners has been learned and re-teach. Even in the school's experiment lesson, it is not easy for a teacher to take care of each student's learning progresses and status.

In the previous work, a simulated platform which provides learners experiment with visual equipment on computers, *virtual experiment environment (V.E.E.)*, had been constructed on the computer network. [Tung95][JCCH96][HSCX98][HHCH99] It proposes an open architecture of cooperative distance learning environment on WWW. Thus learners can devote themselves to the simulated experiment and acquired knowledge. In V.E.E., experiment's tools and equipment can be dynamically inserted just like Software IC at any time.

This paper is trying to analyze and design a methodology make the *intelligent assistant agent* is possible to participate in V.E.E. as different roles, such as tutor and tutee, for the issues of helping learners. Besides, in order to let heterogeneous agents in the same experiment environment can share the information and knowledge, a formal agent communication language, KQML, is proposed as the communication protocol.

Session 2 describes the relations between CAI systems and intelligent agents, hence, the requirements of applying the agents to distance learning system are also be discussed. Several roles in a cooperative learning system, such as virtual experiment system, are analyzed in Session 3. Session 4 designs a state-transition-flow-like knowledge base for the *intelligent assistant agents* with the theory of problem solving cycle. An experiment system built on the architecture of *virtual experiment environment* is used to evaluate the agents' functions in Session 5. And a simple conclusion and possible future works are described by Session 6.

2. CAI Systems and Agents

Computer-assisted instruction (*CAI*) systems have been developed for many years. Each system wants to find out the best teaching model for student's learning in order to help catch up the goals of lessons. Problem solving is a series of activities in which the learners are provided much more chances to manipulate the learned knowledge and skills.

For the improvement of the learning efficiency in science experiment, the problem solving cycle (AUP, 1986) is proposed as Figure 1 shown follows. With these seven stages of problem solving cycle, misconceive of knowledge and operations can be analyzed simply. (Galili, Bendall & Goldberg, 1993;Graham, 1994) When such methodology is applied to computer, more industrialized CAI software can be built to assist students in science experiment.

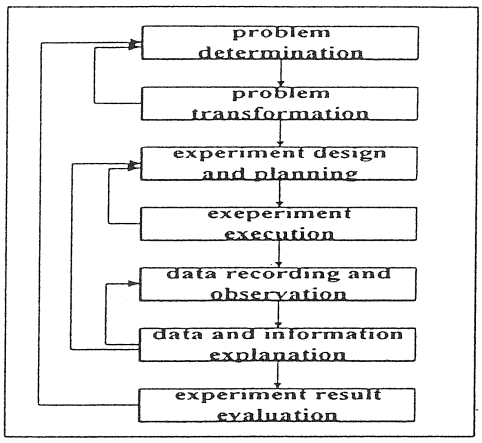


Figure 1. Seven Stages of Problem Solving Cycle

As mentioned in the Webster's New World Dictionary (Guralnik, 1970), an agent is considered as a person has capability of acting for another. In computer science, a software agent is a computing entity that performs user delegate tasks autonomously. With this definition, an agent can be analyzed to perform its abilities, such as interaction, delegation, negotiation and autonomy, in the seven stages of problem solving cycle to help learners.

This paper proposed a methodology for applying an *intelligent assistant agent* to a cooperative distance learning environment based on multi-agent system and the seven stages of problem solving cycle to help learners and accomplish the whole science experiment. Which means, the agent in this learning environment will be the greatest learning companion to serve different learners and provide several functions to achieve the goal of science experiments cooperate with learners.

3. Roles of Assistant Agents Analysis

For this purpose, roles that an *intelligent assistant agent* in the cooperative distance learning environment can play should be analyzed firstly as follows:

- Classmate
  Learners can learn with her little by little via solving the whole problem together.
- Tutor
  She usually provides a hint when leaner is in trouble or calls for help.
- Supervisor

Supervisor can not only provide solutions or hints for learners just like a tutor, but also figure out the process which learners do is right or not.

Before any role can be play by an agent, her own beliefs and intentions should be built. Figure 2 shows a knowledge base of experiment inside agent. This knowledge base of agents formed by state transition flow which trying to lead learners to understand/solved a unique problem. Any sort of knowledge her need can be obtained by two mechanisms: instructor specific, or derived from other knowledge sources.

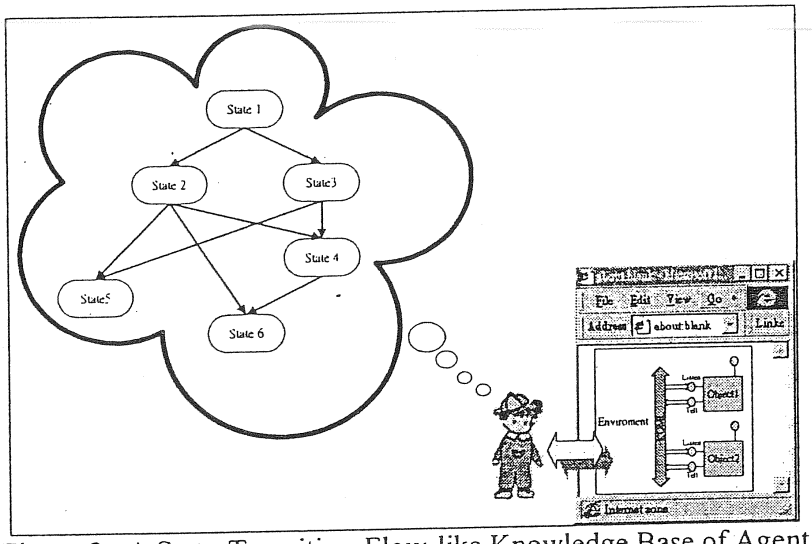


Figure 2. A State-Transition-Flow-like Knowledge Base of Agent

By using instructor specific mechanism, teaching model of the specific education domain can be built into the assistant agent's knowledge base during the design time. Besides the models can build in the beginning, agents can also retrieve information form other knowledge sources such as experiment *script*s and other agents.

According to the knowledge base owned by agent, those three roles described above, classmate, tutor and supervisor can be played. Besides the intelligent ability of assistant agent, some capabilities of computer software includes, recording, replaying or even navigating, might also be useful to achieve the goal of teaching.

Take a physics experiment for example, an *intelligent assistant agent* not only can be a teacher or tutor to tell which step is right or not, but also can be a camera to record each step made by learners. After the experiment is done, learners can figure out what going wrong during the experiment by replaying the record. Moreover, teacher can also understand the learning condition of learners, such as good, medium or bad, and adjust his/her teaching strategies through the replaying function of agents.

4. Intelligent Assistant Agents Design

The weakness of most learning system is that it can't refer the explicit and assertive goal for experiment. And teacher can't use it to monitor the whole progress of experiment for each learner. To overcome this weakness, this paper proposed a *script* file to let teacher design an experiment *script* as Figure 3 shown.

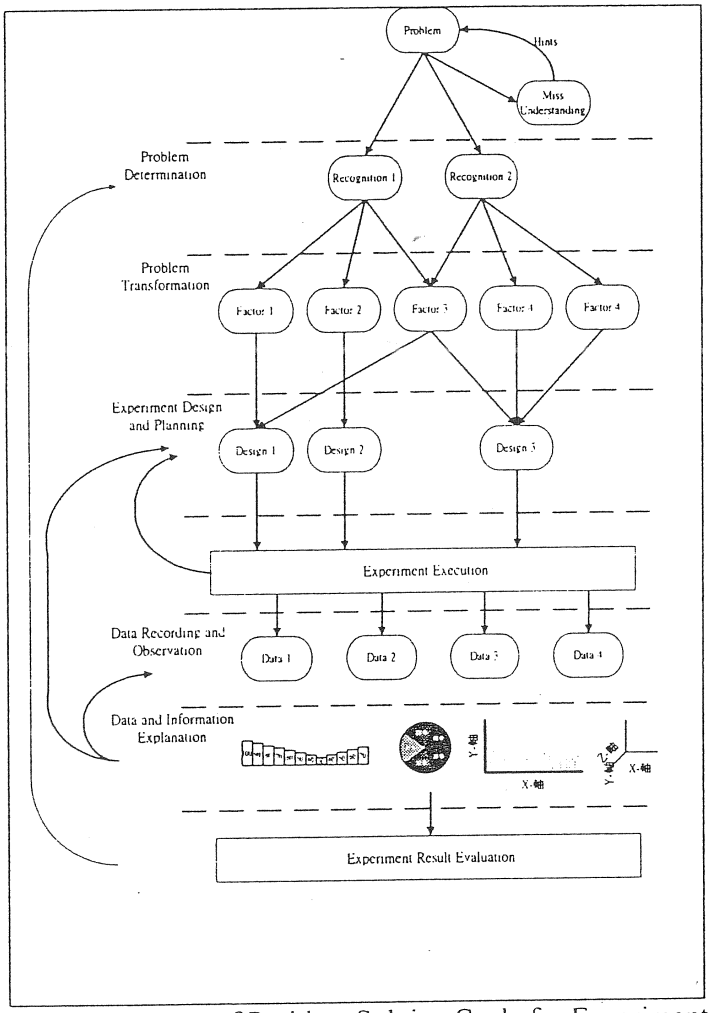


Figure 3. Seven Stages of Problem Solving Cycle for Experiment Script

The beginning state of the *script* is the problem description. Learners must determine the problem first and acquire the comparative recognition. If the learner go to the incorrect recognition, the agent have to prompt the learner and give some hints to help him/her go back to the correct recognition. After the problem transformation, recognition will be transformed into different experiment factors. Based on these factors learners can design the experiment by his/her knowledge, and experiment it realistically to prove his/her recognition.

After the experiment, several tools to record the experiment result are used by learners to evaluate the experiment is success or not. This means agent should be able to retrieve information from the experiment equipment as knowledge sources to build her own knowledge base. With the state-transition-flow-like knowledge base the agent can lead learners design an experiment step by step.

To decide what information to retrieve and what action to take, agents must be able to communicate with each other or even experimental equipment. Thus in order to let heterogeneous agents, agents and experiment equipment, can communicate, it needs a mechanism to exchange information and knowledge, is so called Agent Communication Language (ACL).

In the open architecture of agent-based *virtual experiment environment*, Knowledge Query and Manipulation Language (KQML), the most useful ACL and formal semantics among agent-based programs, is used as the communication protocol between objects in V.E.E..

KQML focuses on an extensible set of *performatives*, which defines the permissible

operations that agents may attempt on each other's knowledge and goal stores. The performatives comprise a substrate on which to develop higher-level models of inter-agent interaction such as contract nets and negotiation.

Conceptually, a KQML message consists of a performative, its associated arguments which include the real content of the message, and a set of optional arguments which describe the content in a manner which is independent of the syntax of the content language. For example, a login to a group learning system might be encoded as:

(tell :content *(Login(Ken(jk1234y, user)))*
:language *list*
:*ontology* Group-*Learning)*

In this message, the KQML **performative** is *tell*, the **content** is *(Login(Ken(jk1234y, user)))* and the assumed **ontology** is identified by the token *:Group-Learning*.

5. Experiment System

As mentioned previous sections, the *script* file is used to construct the *state-transition-flow-like* knowledge base inside the *intelligent assistant agent*. In the experiment system this paper built, a *Script Editor* is given for teacher to design his/her own experiment *scripts*. The editor provides some tools for teachers to construct the *script*s as Figure 4 shown follows. On the *script*, the stages placed on it are corresponded to those in problem solving cycle. It is easily for a teacher to choose any object in the tool bar, and then drag-n-drop it on the *script* in order to edit the *script* of an experiment.

For example, teachers can put some State Objects in one of these stages on the *script* and use Link Objects to construct the relations between states. In the part of the experiment, a teacher can choose the appropriate experimental environment he/she needs, such as a physics lab for "The law of universal gravitation," and design his/her own experiment handbook.

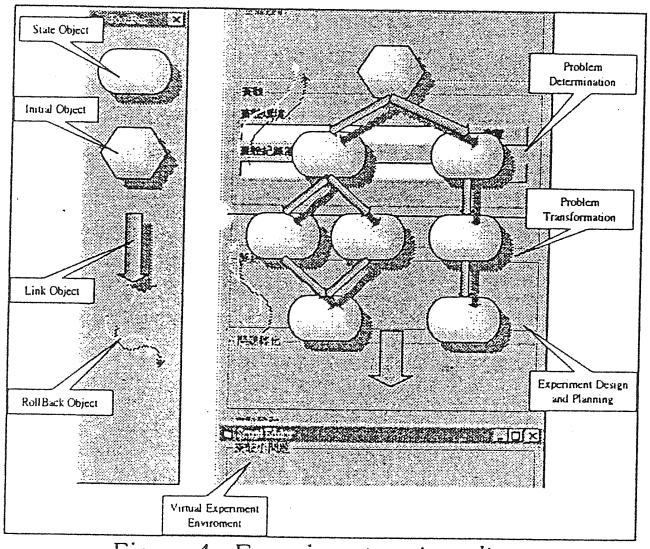


Figure 4. Experiment *script editor*

An assistant agent can record the activities and the learning path from learners. With these records, teachers can use it to analyze the behavior of the learners in the specific experiment and estimate the learning condition of learners. Moreover, several hints are given to those learners as they are in trouble during experiment from assistant agents. According to its own state-transition-flow-like knowledge base, the assistant agents try to give more knowledge

to lead learners achieve the goal of experiment through the correct solving way.

For the purpose of applying the *intelligent assistant agent* to the distance learning environment, an open architecture environment called V.E.E. (*Virtual experiment environment* as Figure 5 shown is designed. In this open architecture, assistant agents can easily monitor the experiment or help learners to complete the experiment. For improving the learning efficiency agents can watch out every step the learner do and record it.

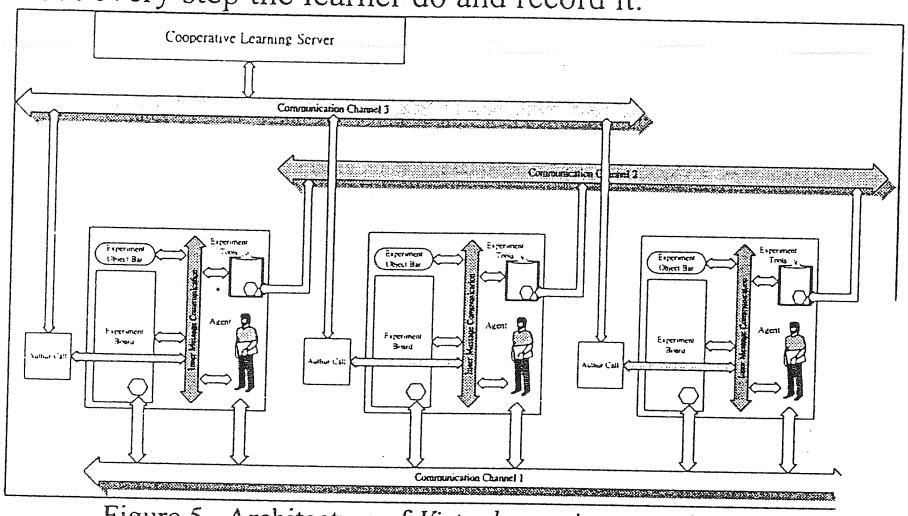


Figure 5. Architecture of *Virtual experiment environment*

Learners then can replay these records in order to check out what he/she had done in this experiment or even review the experiment he/she did before. Besides learners can trace what going wrong by using the records, teachers can also use them to see if the learner really understood this experiment or not.

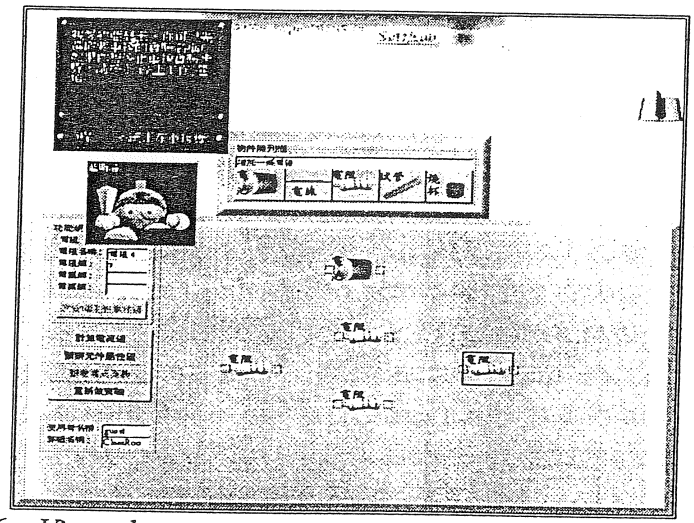


Figure 6. *Virtual experiment environment* of circuit experiment

Figure 6 illustrate an *intelligent assistant agent* lives in the experiment environment of circuit experiment based on the architecture of *virtual experiment environment*. During the experiment, agent might be able to monitor the actions learners made, give hints the learners required and evaluate the step learners taken is right or not. After the experiment, VEE supports some tools for experiment data analyzing.

6. Conclusion

In this paper, a better distance learning environment with an *intelligent assistant agent* for learners is fully analyzed and designed. There are four roles, including classmate, tutor, supervisor and even camera, can be played by the *intelligent assistant agent*s in the experiment system based on the architecture of *virtual experiment environment* constructed in this paper.

To make learners have the proper learning procedure, the agent guides the learners to the correct solving way based on its own state-transition-flow-like knowledge base and the seven stages of problem solving cycle. The agent also has the abilities of recording the steps of experiment when learners are experiment on V.E.E.. And by using those recorded data, agent can evaluate what learners had been learned and analyze student's learning progresses and status.

It will be a good research direction that agent can autonomously analyze those records and find out the appropriate learning path. It is also feasible for agent to analyze the experiment data record by learners in order to judge if the experiment of the student is success or not.

References

[CaK98] Jamie C. Cavalier and James D. Klein, "Effects of Cooperative Versus Individual Learning and Orienting Activities During Computer-based Instruction," ETR&D, Vol. 46, No. 1, 1998, pp.5-17

[Chan95] T.W. Chan, "A Tutorial on Social Learning Systems", *Emerging Computer Technologies in Education,* AACE, 1996

[DiL92] Th. Diessel, A. Lehman, "An ITS for engineering domains: concept, design and application," 4th International Conference, ICCAL'92 Wolfville, Nova Scotia, Canada, June 17-20, 1992 Proceedings

[FuL92] M. Futtersack, J.-M. Labat, "QUIZ: A distributed intelligent tutoring system," 4th International Conference, ICCAL'92 Wolfville, Nova Scotia, Canada, June 17-20, 1992 Proceedings

[GSCX92] J.C. Gonzalez, J.J Sancho, J.M. Carbo, A. Patak, F. Sanz, "Intelligent tutoring system in medicine through an interactive," 4th International Conference, ICCAL'92 Wolfville, Nova Scotia, Canada, June 17-20, 1992 Proceedings

[HHCH99] Chi-Wei Huang, Chang-Kai Hsu, Maiga Chang and Jia-Sheng Heh, "Designing an Open Architecture of Agent-based *Virtual experiment environment* on WWW", World Conference on Education Multimedia, Hypermedia & Telecommunications, ED-MEDIA 99, in Seattle, Washington U.S., 1999 (accepted)

[HSCX98] Chi-Wei Huang, Shiao-Ting Sun, Chia-Chin Chang, Kuo-Chang Jan and Kun-Yuan Yang, "Education Software and Hardware", 2nd Global Chinese Conference on Computer in Education, Hong Kong, 1998, pp.50-57

[JCCH96] Yu-Wei Jeng, Maiga Chang, Ivory Chung and Jia-Sheng Heh, "Designing Objects for Virtual Experiments," OOTA'96, Taiwan, 1996, pp.331-339

[Lia92] H. Lianjing, "A tool for developing intelligent tutoring system," 4th International Conference, ICCAL'92 Wolfville, Nova Scotia, Canada, June 17-20, 1992 Proceedings

[MaG98] Doug L. Maskell and Peter J. Grabau, "A Multidisciplinary Cooperative Problem-Based Learning Approach to Embedded Systems Design," IEEE Transactions on Education, Vol. 41, No. 2, May 1998, pp.101-103

[SiH98] Chanchai Singhanayok and Simon Hooper, "The Effects of Cooperative Learning and Learner Control on Students' Achievement, Option Selections, and Attitudes," ETR&D, Vol. 46, No. 2, 1998, pp.17-33

[Tung95] Sho-Huan Tung, "Visualizing Evaluation in Scheme", 1995