Int. J. Engng Ed. Vol. 25, No. 4, pp. 772–776, 2009
Printed in Great Britain.

0949-149X/91 $3.00+0.00
© 2009 TEMPUS Publications.

Pedagogical Agents for Personalized Multi-user Virtual Environments*

MARYAM ASHOORI
University of Waterloo, 200 University Avenue West, Waterloo, Ontario, Canada N2L 3G1.
E-mail: mashoori@uwaterloo.ca

ZHIQI SHEN and CHUNYAN MIAO
Nanyang Technological University, Singapore 639798

LIAM PEYTON
University of Ottawa, 75 Laurier Avenue East, Ottawa ON Canada K1N 6N5

Personalization is a key issue in adaptive virtual learning environments, which support interactive-engagement methods for learners. Recently there has been much progress on advanced graphical interfaces, but more progress is needed to fine tune and adapt such interfaces to end-user's abilities and preferences. One of the emerging challenges in such an adaptive multi-user virtual environment (MUVE) is the development of personalized services such as personalized content management, user-models, or adaptive instant interaction. This paper uses agent technology as an effective framework for developing personalized services in a MUVE. The proposed framework is validated by applying it to Virtual Singapura (VS), an agent augmented virtual environment designed to engage and motivate learners at the lower secondary level in Singapore as they learn important scientific knowledge and skills.

Keywords: personalization; pedagogical agents; multi-user virtual environments; Dempster–Shafer belief accumulation; user model

INTRODUCTION

ADAPTIVE MULTI-USER VIRTUAL ENVIRONMENTS (MUVEs) are an emerging pedagogical tool for imparting skills and knowledge related to engineering. A comprehensive study [1] of 62 physics courses involving 6500 students has shown that interactive-engagement methods can be far more effective than traditional lecture methods in imparting understanding of scientific concepts. However, providing such intensive personalized interaction is challenging without some form of online automated support. In the next generation of MUVEs, agent augmented learning environments support immersive virtual and mixed worlds. The synthetic characters in these environments are augmented with advanced agent technologies. For example, an agent can be used to support personalized interfaces for different learners.

Over the last decade, there have been several empirical research projects looking into agent augmented virtual environments, their usage, effectiveness and limitations [2, 3]. Some of the developed agents are life-like avatars that inhabit a virtual world and some are just intelligent agents with bounded rationality to facilitate learning and attract learners to get involved with the environment. However they failed to provide personalized interaction for individual learners.

* Accepted 24 February 2009.

This paper describes an agent-based framework for these agent augmented MUVEs in order to provide highly accurate personalized interaction according to the learner's background and preferences.

RELATED WORK

With respect to architecture, three main architectures have emerged for on-line generation of agent behavior [4]. The first approach is the *behavior sequencing approach,* which is based on a behavior space. This is a library of predefined primitives (actions, speech elements, etc.). Herman the Bug [3] follows this architecture. In this approach, life-like pedagogical agents have been specified by:

- a behavior space containing animated and vocal behaviors;
- a design-centered context model that maintains constructive problem representations, multi-modal advisory contexts, and evolving problem-solving tasks, and
- a behavior sequencing engine that dynamically selects and assembles agents' actions to create pedagogically effective, lifelike behaviors.

For example, in Herman the Bug, as the learners make their design decisions, the behavior sequencing engine monitors the activity, updates the task model, selects behaviors from the behavior space, and assembles them into a multi-modal behavior stream.

The second architecture is the *layered generative approach*, where animations are generated in real time. However it requires a much higher rendering computation load [4].

The third approach to the architecture of pedagogical agents is *State Machine Compilation Approach,* which composes behaviors out of primitives, but generates a state machine, so that the behavior of an agent can adapt at run time to learner actions. JACK [5] follows this architecture.

Learners generally expect pedagogical agents to be believable as virtual mentors, be entertaining, easy to communicate with, helpful and diversified. Regarding personalization, some of the current pedagogical agents, such as ADELE [6] suffer from a lack of proper help for learners who encounter problems. These kinds of pedagogical agents are informative agents who only provide and present information through interacting with the learner. A robust process of learner activities should be designed that affectively leads the agent to interact with learners based on the history of the learners' activities as well as their preferences [3]. Agents are supposed to be equipped with location awareness and situation awareness to trace learners and provide helpful hints when required. STEVE [7] can sense where the learner is and what he or she is looking at, and can adapt instructions. In addition to answering the learners' requests for help, MERLIN [8] provides unsolicited hints to overcome the learners' tendency to avoid seeking help even when they need it. To decide when to intervene and what hints to provide, the agent relies on a probabilistic model of the learner's factorization knowledge. However, there is still a long way to go to design an effective believable entertaining pedagogical agent.

AGENT-BASED FRAMEWORK FOR PERSONALIZED MULTI USER VIRTUAL ENVIRONMENTS

In this paper, we propose a framework where pedagogical agents cooperate and collaborate toward providing a personalized interface through interacting with learners. The framework is illustrated in Fig. 1. The agent model consists of a *Belief Model* and a *Goal Model*. The *Goal Model* follows a goal-net based architecture [9].

In virtual environments, learners should go through different scenarios that are designed and implemented in the environment (e.g. solving a problem or investigation of some evidence). Each scenario can be modeled as a collection of different states. Goal-net architecture [10] defines the states and their relationships according to these scenarios. A goal-net is hierarchically structured.

The root composite state at the highest level of the hierarchical structure represents the overall goal of the agent and the composite states in lower levels of the hierarchical structure represent sub-goals of the agent. A higher level of composite states (goal or sub-goals) can be split into lower-level states connected via transitions. An agent commences its goal pursuit from the root state; it then goes through the hierarchical structure to reach its final goal.

In addition to the *Goal Model*, agents need to have a *Belief Model* in order to decide how to travel different states of their *Goal Model*. The *Belief Model* is like the brain of an agent. A *Belief Model* describes the information about the agent's belief to decide and plan for goal permutation in their goal model. The decision making process for goal-selection is proposed on top of the Dempster–Shafer belief accumulation [10, 11]. The Dempster–Shafer theory offers a knowledge model about

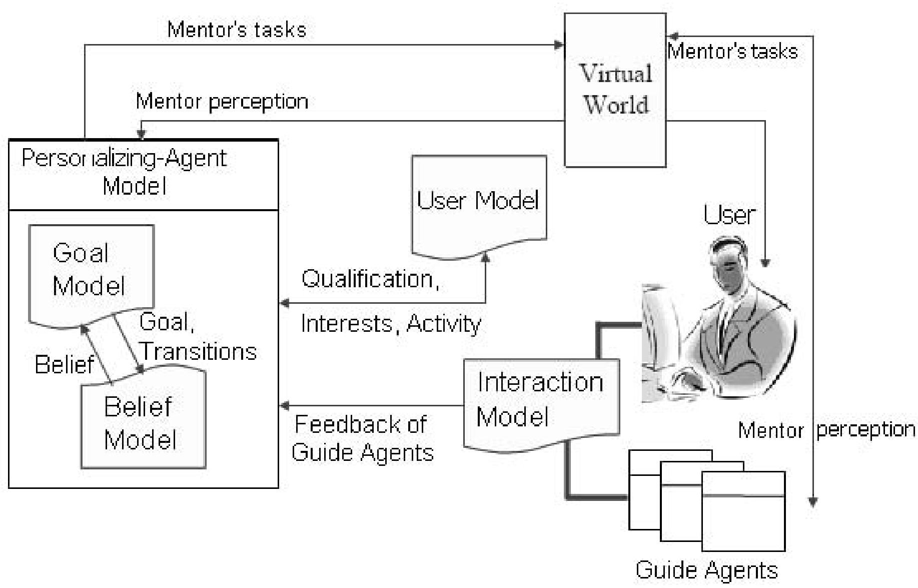

Fig. 1. Agent personalization framework.

one or more hypotheses, enables the quantification of such concepts as imprecise measurements or uncertainty, and agrees to allocate probability-like weights to a set of events in a way that allows statements of ignorance about likelihood of some of the events. Dempster–Shafer theory through allocating belief values to the same set of events could present a natural way of combination to give a fused allocation of belief that deals both with ignorance and with conflict between the original beliefs. It derives *Belief* and *Plausibility* towards making an optimum decision over more comprehensive information.

The *Learner-Model* defines all the information about learners in the environment. The *Interaction Model* defines protocols and mechanisms for interactions between agents and between agents and learners. The proposed negotiation protocol in the *Interaction Model* can be used to update the *Learner-Model* through a collaborative channel of heterogeneous agents.

Our main focus is on a situation-aware agent called *Mentor* that monitors all the learners' activities and updates the related learner-model accordingly. *Mentor* is kind of coordinator that interacts with all other *agents*, evaluates their efficiency and monitors learners in all locations.

Based upon how effective the other agents have been in influencing the learners, *Mentor* runs a credit assignment algorithm through the system, which is a kind of reward for those agents who communicate effectively with learners. Thereafter, next time that the learner asks the *Mentor* where to go to investigate, the *Mentor* will suggest those areas and those effective agents with a higher probability.

Looking at learner modeling, one of the most popular standards for modeling learner profile, IMS Learner Information Package (IMS LIP) [12], has been adapted. Among all the IMS LIP items, the items related to preferences are 'Activity', 'Identification' and 'Interests', while the system is basically built to serve as a supplement learning environment rather than organization selection of new employees. Learners are modeled based on four IMS attributes including:

1. *Identification*, which includes information about learners (*Name, Id, Gender, Age*)
2. *Activity*, which is a record of learner activities in the virtual world *(Visited places, . . .)*

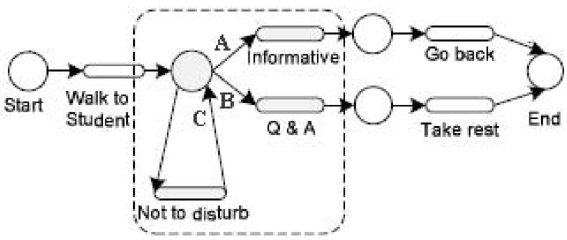

Fig. 2. A goal net of mentor in VS [9].

3. *Familiarity*, which is a fuzzy attribute to record the learner's performance through the system; it evaluates how familiar the learner is with the story of virtual world in each step
4. *Interests of learners (language, length of descriptions)*.

SAMPLE APPLICATION: PERSONALIZATION IN VIRTUAL SINGAPURA

As the sample application, *Mentor* is modeled in a sample agent augmented learning environment, Virtual Singapura (VS). VS is a virtual environment with different scenes where a society of agents is available to help learners in the process of learning. *Mentor* is considered to be the virtual mentor that monitors learners and provides personalized help in collaboration with other agents/bots in the virtual environment.

A *Goal Model* of the *Mentor* was modeled based on goal-net architecture as shown in Fig. 2. Three different modes of activities have been considered for the agent to follow.

- State A demonstrates Mentor as an informative agent that goes to the learner providing some information about the environment and the current situations.
- State B depicts Mentor as a question answering agent that provides personalized answers to learner's questions.
- State C would be the state where the agent determines that the learner does not need them and decides not to disturb the learner.

The learner can ask for help or the *Mentor* can decide to offer some help to the learner. This help can be specific or generic. A generic help provides examples, formulas and explanations for a pedagogical subject. The specific help is a personalized help for the learner based on their learner model. The learner can reject the help offered by the agent, but the agent must always provide the help asked by the learner.

The interaction model in this research is to personalize learner interaction in order to make *Mentor* as believable as a virtual teacher.

Mentor is the coordinator of the environment that interacts with all other *guide-agents* (other intelligent avatars of environment), evaluates their efficiency and monitors the learner. The proposed negotiation protocol in the Interaction Model can be used to update the Learner-Model through a collaborative channel of heterogeneous agents. Learners interact with both *Mentor* and *Guide-Agents.* Whenever learner leaves a scene, *Mentor* applies a linear update of *'Familiarity'* in the learner-model based on the feedback of other agents of environment that have interacted with learner on that scene.

By applying the proposed framework to this sample application, a personalized agent-mediated e-learning system was designed. In this case study, we developed a highly learner-centric e-learning system where the interface between *Mentor* and end-users is personalized based on learner preferences, interests, knowledge, and qualifications.

CONCLUDING REMARKS

One of the key issues of virtual learning environments is personalization. In addition to developing highly graphical interfaces for end-users, instant interactions should be personalized based on learner's knowledge, background, abilities, interests and preferences. In this paper, an agent-based framework was proposed for adaptive virtual environments. Before we made our additions with agent technology, learners often got stuck, and did not know how to get help. According to the experimental results, learners generally expect pedagogical agents to be as believable as virtual mentors, entertaining, easy to communicate, helpful and diversified. Our *Mentor* is moving, walking to the learner, waving a hand to grab their attention, standing in different poses to stimulate the learner emotionally.

The proposed agent personalization framework provides a more dynamic, learner-aware, and context-aware infrastructure for a virtual learning environment where the learners would get much more attractive personalized experiences. It is a good starting point for developing rational agents that can both stimulate learning and maintain the high level of engagement that games and virtual environments usually generate. The proposed approach could be used wherever there is a society of agents to personalize content. This idea is not limited to e-learning systems and can be applied to a wide variety of application domains where personalizing the content could improve the performance.

REFERENCES

1. R. R. Hake, Interactive-engagement versus traditional methods: A six-thousand-student survey of mechanics test data for introductory physics courses, *American Journal of Physics*, **66**(1), (1998).
2. C. T. dos Santos and F. S. Osrio, An intelligent and adaptive virtual environment and its application in distance learning, *Proc. of the Working Conference on Advanced Visual Interfaces*, Gallipoli, Italy, May 25–28, (2004).
3. J. C. Lester, B. A. Stone and G. D. Stelling, Lifelike pedagogical agents for mixed-initiative problem solving in constructivist learning environments, *User Modeling and User-Adapted Interaction*, (1999) pp. 1–44.
4. A. Mitrovic and P. Suraweera, Evaluating an animated pedagogical agent, *Proc. of the 5th international Conference on intelligent Tutoring Systems, LNCS*, vol. 1839. Springer-Verlag, London, (2000) pp. 73–82.
5. T. Noma and N. I. Badler, A virtual human presenter, *Proc. of the IJCAI Workshop on Animated Interface Agents: Making them Intelligent*, (1997), 45–51.
6. E. Shaw, W. L. Johnson and R. Ganeshan, Pedagogical agents on the Web, *Proc. of the Third Annual Conf. on Autonomous Agents, AGENTS '99*, ACM Press, NY, (1999), pp. 283–290.
7. J. Rickel and W. Lewis Johnson, STEVE: A Pedagogical Agent for Virtual Reality, *Proc. of the Second International Conf. on Autonomous Agents*, Minneapolis/St. Paul, ACM Press, (1998).
8. C. Conati and X. Zhao, Building and evaluating an intelligent pedagogical agent to improve the effectiveness of an educational game, *Proc. of the 9th International Conf. on Intelligent User Interface*, (2004), pp. 6–13.
9. M. Ashoori, C. Miao and Y. Cai. Socializing pedagogical agents for personalization in virtual learning environments, *Proc. of International Conferences on Web Intelligence and Intelligent Agent Technology*. WI-IATW. 346–349, (2007).
10. Z. Shen, C. Miao, X. Tao and R. Gay, Goal oriented modeling for intelligent software agents, *Proc. of International Conference on Intelligent Agent Technology*, (2004), pp. 540–543.
11. G. Shafer, *A Mathematical Theory of Evidence*, Princeton University Press, (1976).
12. M. Ashoori, C. Miao, A. Eck Soong Goh and W. Qiong, *Intelligent Market Based Learner Modeling*, Lecture Notes in Artificial Intelligence 4099, Springer-Verlag, (2006), pp. 101–111.

Maryam Ashoori is a Ph.D. student in Systems Design Engineering at the University of Waterloo. She received her BSc in software engineering from Amirkabir University of Technology in 2003, her MSc in artificial intelligence and robotics from the University of Tehran in 2005, and her MEng in computer engineering from Nanyang Technological University in 2008. Her research interests include intelligent systems and cognitive collaborative environments.

Jaclyn, Chunyan Miao is an Assistant Professor and Deputy Director of the Emerging Research Lab, the School of Computer Engineering at Nanyang Technological University (NTU). Her current research focus has been the exploration of novel real-time artificial intelligence and computational intelligence models for modeling human factors or human-like characteristics of agents in a virtual world and massive multiplayer games. Her research goal is to create new dimensions and experiences in real world applications through agent

mediated technology. To ensure high-quality agent systems in real world applications, she is also interested in organizational issues for multi-agent systems and agent-oriented software engineering.

Zhiqi Shen is currently a visiting assistant professor of Nanyang Technological University (NTU), Singapore. His research interests include goal-oriented modeling, intelligent software agent and interactive storytelling/games.

Liam Peyton, Ph.D., P.Eng., is the principal investigator for the Intelligent Data Warehouse laboratory and Associate Professor at the University of Ottawa, which he joined in 2002 after spending 10 years as an industry consultant and instructor specializing in business process automation, performance management, and software development methodologies. He has degrees from Aalborg Universitet (Ph.D. 1996), Stanford University (MSc 1989), and McGill University (BSc 1984).