Proceedings of 2nd International Conference on Information Technology Based Higher Education and Training, Kumamoto, Japan
July 4-6, 2001

Distributed Knowledge-base Execution System with User Adaptability for Mobile Distance Learning

Shin-ichirou Okada, Yoshitaka Fujiwara, Nobuki Takakura and Buntarou Matsuzawa
Department of Computer Sciences, Kitami Institute of Technology
okada@ialab.cs.kitami-it.ac.jp, fujiwara@cs.kitami-it.ac.jp
matuzawa@ialab.cs.kitami-it.ac.jp

Abstract

We have proposed the Java Virtual Execution Environment Control (JVEEC) which is a knowledge base system for distance learning assistance. JVEEC has two characteristic features. One is a feature which improves response performance by using prefetch transmission. The other is a self-adaptation feature whereby the system can change its behavior to suit the user's learning conditions. This paper reports the evaluation of the performance of JVEEC using PHS, and the result of the verification of the self-adaptive function based on a causal network. Through the experimental evaluation, the improvement of response time and the validity of the self-adaptive function using a causal network have been verified.

1. Introduction

Recently, various research studies relating to distance learning assistance systems have been undertaken, prompted by the spread of wide area communications, like the internet.

In order to implement a distance learning assistance system with no instructors, there is a strong requirement for the following features: rapid response of the system to a user's input, and a self-adaptive function enabling flexible changes to the assistance contents in accordance with the "user profile" (characteristics of the understanding of the user).

This paper proposes a distributed knowledge base execution system with the above features for a distance learning system for use with a mobile communication system such as PHS (Personal Handy-phone System) [1]. It is called Java Virtual Execution Environment Control (JVEEC) [2].

In this system, the knowledge base is organized with a set of independently executable partial knowledge bases, "active nodes", and a meta-knowledge for their execution control, the "context graph". Rapid response is achieved on the client computer by a partial knowledge base execution mechanism with a prefetching control of active nodes using the context graph. The response time of the distributed system has been assessed using a prototype system and PHS.

In addition, a method for adapting a sequence of active nodes of the target knowledge base is proposed. In this method, "priorities" for the active nodes of the target knowledge base are introduced, to determine the most appropriate active node among the candidates at each node selection stage in the execution of the target knowledge base. These priorities are calculated using a causal network (also called a Bayesian network) derived from the cause-effect relations between reaction events defined for each action event of the target knowledge base. Through evaluations using an experimental knowledge base for a hierarchical learning assistance situation, it was observed that the results of adaptation using this method were consistent with those provided by a human expert.

There has been considerable research on learner modeling using a causal network[3]–[11]. The purpose of the work described in this paper is to further improve the degree of understanding of the user in the assistance paths, based on a path selection criteria provided by a causal network. This technique is a new idea, distinct from others.

2. Learning Assistance Model

An interactive learning assistance system with bidirectional communication between user and system is assumed. The user assistance function is assumed to be constructed hierarchically from general content to detailed content. This structure is illustrated diagrammatically in Figure 1. In this figure, a specific path from the root node to a leaf node, called an assistance path, represents a meaningful sequence of assistance actions. For instance, the path indicated by thick lines represents a specific sequence of actions: explanation of field f_1, then explanation of topic t_{12} of f_1, and finally execution of exercise problem e_{121} of t_{12}. Usually var-

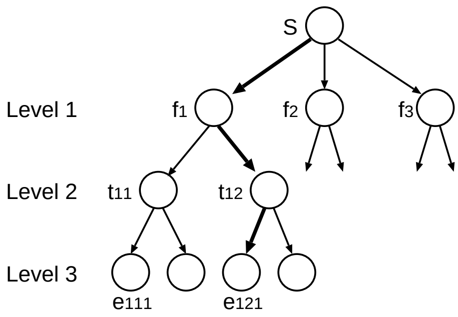

Figure 1: Learning assistance model with a hierarchical structure

ious assistance paths are selected repeatedly and each action on the selected path is executed until a certain end condition occurs.

3. Overview of JVEEC

First, we describe the structure of the knowledge base in JVEEC, that is, teaching material for distance learning. This knowledge base consists of active nodes and a context graph. An active node is a partial knowledge base. Its fundamental element is a Java class file, and it includes text files and image files used by the class file. Each of the active nodes is a part of the teaching material relating to one problem or one explanation or one menu, and the set of active nodes behave as a single knowledge base (teaching material). Now active nodes don't have the information necessary to execute themselves in sequence. This information is described as a context graph, and it performs a role of meta-knowledge for the active nodes. The self-adaptive function of JVEEC is achieved by modifying the context graph. Figure 2 shows the concept of active nodes and context graph.

Secondly, we describe a method of implementing a knowledge base constructed from active nodes and a context graph. Figure 3 shows the system structure of JVEEC. Active nodes are initially stored in a server computer. These are transmitted to a client computer on request from the client computer, and are then executed on the client computer. There is a controller in the client computer, which requests the transmission of active nodes from the server and controls the action of active nodes according to context graph. While an active node is being executed by the controller (in Figure 3, active node A), thus starting interactive service for the user, the controller infers which active nodes may need to be executed next (in Figure 3, B,C) and transmits the prefetch request to the server. The server transmits the corresponding active nodes to the client computer. The controller in the client computer receives the active nodes and stores them in a cache. After the currently-active node has finished interactive service for the user, the controller decides which active node should be executed next and executes it. Thus, active nodes are executed in sequence on the client computer. Consequently all active nodes behave as a single knowledge base. Because of the above mechanism, the execution of one active node and transmission of the next active nodes can be carried out in parallel in JVEEC. Therefore, the user's waiting time due to transmission of data is considerably reduced.

4. Performance Evaluation of JVEEC using PHS

For this study, we built a communication environment using PHS. We selected PHS among the options for readily available mobile communication links, since it is inexpensive and yet offers a reasonably high-speed data link. As a client computer we used a notebook PC having a PC card type PHS terminal. On the server side, we set up an ISDN communication line and a dialup router so that the client computer can access the server computer on the LAN. Since the products support PIAFS 64K as the PHS terminal and the dial-up router in this environment, the server and client computers can communicate at 64kpbs (effective bit-rate is 58.4 kbps on the 64kbps service).

Using this environment, we evaluated the performance of JVEEC using PHS. In this evaluation, the measure of performance is the time of response to user input with a specific think time. Here, the response time has been defined as the interval between the user's input and the corresponding system's answer. And, the think time has been defined as the interval between the system's displaying and the corresponding user's input. The knowledge base of exercises and commentaries for "Class I Information Technology Engineer Examination" was used as the experimental knowledge base. This knowledge base has a hierarchical structure in which a user selects a field, selects a topic, answers several problems, and returns to select another field, in sequence. Each active node included a single selected field or topic or problem. The number of active nodes was 41 and the average size of an active node was 18.3Kbyte. Interaction between each active node and the user was realized with a random selection from the choices offered by an active node. Response times were measured in the case of think times of 10, 20, and 30 seconds. The results of these experiments are shown in Figure 4. This graph is shows cumulative distribution

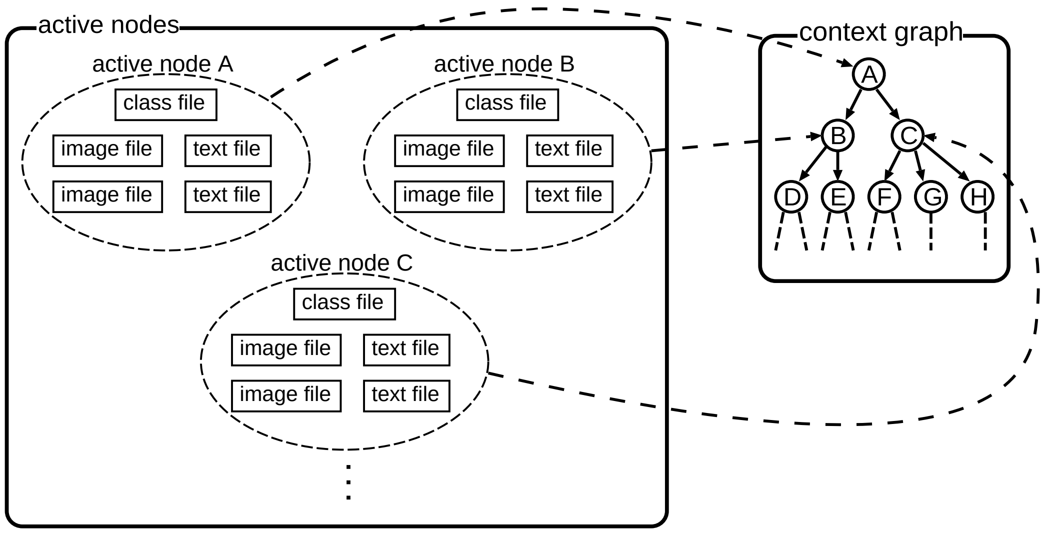

Figure 2: Structure of knowledge base for JVEEC

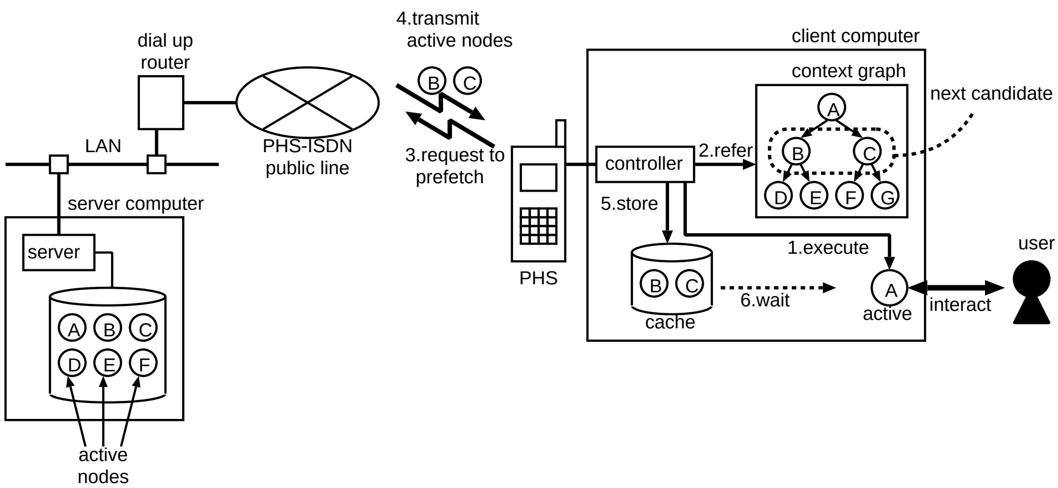

Figure 3: Structure of JVEEC using PHS

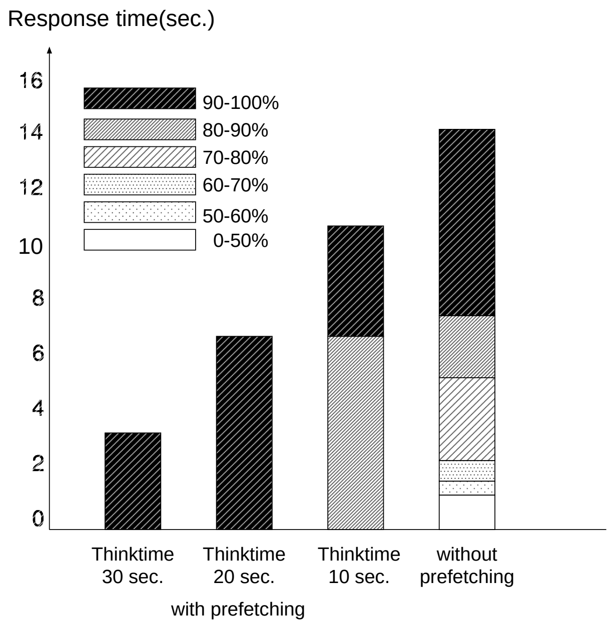

Figure 4: Cumulative distribution of response time

in which the response times are arranged in order and classified in groups up to 50%, 60%, 70%, 80%, 90% and 100%. 80% of all the response times when active nodes were prefetched with an think time 10sec. are in the neighborhood of 0 sec. in the graph. Therefore, this graph shows that prefetch improves the response time. Furthermore, 90% of all response times are in the regions of 0 sec. when think times are 20sec. and 30sec. Hence, it is confirmed that long think time improve the effectiveness of prefetching.

5. Self-adaptation Method using a Causal Network

The causal network is defined as following. First, assume that the graph $G = (V, E)(V$: a set of nodes, E: a set of edges) forms a DAG (Directed Acyclic Graph). Let $c(v)$ be the set of all parent nodes of a node v, and let $p(v|c(v))$, the conditional probability of v given its parent $c(v)$, be assigned under the following constraint

$$\sum_{v} p(v|c(v)) = 1 \tag{1}$$

Here, $\sum_v$ means the sum over all possible values of v. Then a join probability distribution of the vertices in V is uniquely determined by the following formula.

$$P(V) = \prod\nolimits_{v \in V} p(v|c(v)) \tag{2}$$

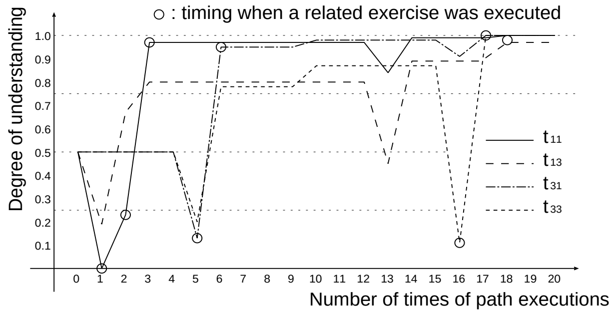

Figure 6: An example of the relationships between degree of understanding of a topic and the number of times a path is executed

The DAG along with its joint probability distribution $P(V)$ is said to constitute a causal network. As an example, the structure of the causal network obtained from the experimental knowledge base used in section 4 is shown in Figure 5.

The causal network for JVEEC to achieve self-adaptation has the following structure corresponding to the assistance learning model shown as Figure 1. First, there is a start node S that memorizes the most suitable next field inferred by a causal network. S is linked to nodes F_i that indicate whether field f_i is understood or not by the user. Each node F_i is linked to nodes T_{ij} that indicate whether topic t_{ij} is understood by the user or not. Each node T_{ij} is linked to nodes E_{ijk} that indicate whether the user's answer for problem e_{ijk} is correct or not.

In the following, the process of self-adaptation using this causal network is described. Suppose that a user answers a problem in the target knowledge base. The fact of whether the answer is correct or incorrect, is input to E_{ijk} and the degrees of understanding for all fields and topics are calculated by a probability propagation algorithm. Then, when the user next selects a field or a topic of choice, the most suitable selection is decided for the user based on the priority and displayed to user. The priority of each acitve node is calculated as the probability $P(F_i = 1)$ that the user does not understand the field f_i or the probability $P(T_{ij} = 1)$ that the user does not understand the topic t_{ij}. In the current version of prototype JVEEC, the most suitable selection, based on the adaptive function, is displayed as the item on the extreme left in the display of choices.

6. Verification of Self-adaptive Function

In this section, we report a result of the verification of the self-adaptive function in JVEEC. To verify the

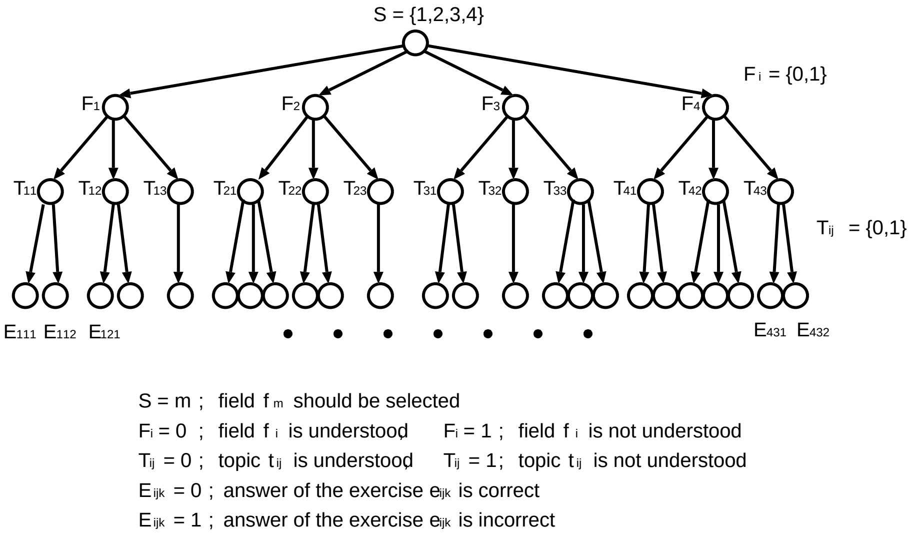

Figure 5: Structure of the causal network obtained from the experimental knowledge base

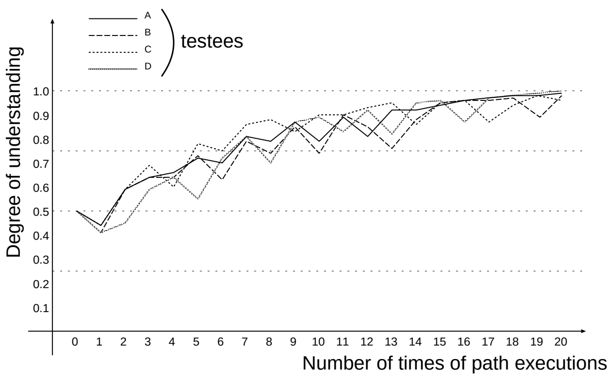

Figure 7: An example of the degree of understanding of a field and the number of times a path is executed

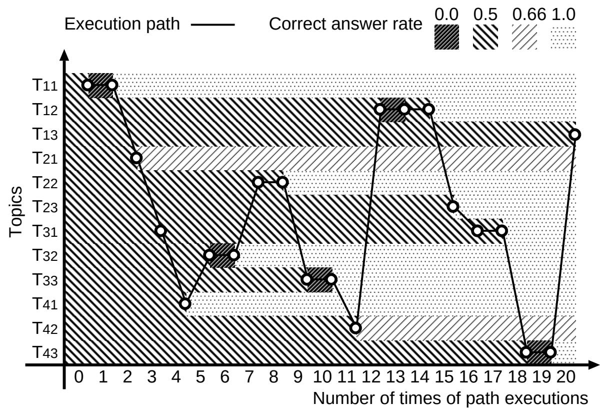

Figure 8: An example of the relationships between changes of the correct-answer- rates of the topics and the sequence of selected topics

practical operation of the adaptive function, we asked students who had not yet passed in "Class I Information Technology Engineer Examination" to use the experimental knowledge base, and record selected fields and topics and the changes of degree of understanding of these (the degree of understanding = 1.0 - the priority). Here, we requested the students to select the most suitable field or topic shown by the system.

Figure 6 shows relationships between the degree of understanding of topics and the number of times of execution of an assistance path, for a specific user. The symbol ○ indicates that the corresponding topic was shown as the most suitable topic to the user and the relating exercise was executed. It is observed that the degree of understanding of every topic changes by the correct answer or the incorrect answer. Figure 7 shows the relationship between the degree of understanding of a specific field and the number of times an assistance path is selected, for several users. It can be seen that in each case the degree of understanding gradually approaches 1.

It has been shown that the histories of the learning assistance paths obtained by the self-adaptation method are not contradictory to those provided by a human expert under the same conditions.

Figure 8 shows the relationships between changes of the correct-answer-rates of the topics and a sequence of the selected topics. In this figure, the topics which have high correct-answer-rates are not selected. this indicates that the most suitable selection decided by the proposed method is valid.

It is therefore concluded that the self-adaptation method described in this paper is effective for the hierarchical learning assistance model.

7. Conclusion

This paper has proposed a distributed knowledge base execution system JVEEC which aims to achieve rapid response and provide a self-adaptive function. The knowledge base in JVEEC comprises a set of active nodes and a context graph.

JVEEC can make the response time approximately 0 by prefetching the next active nodes. It has been verified by an experimental evaluation using PHS.

JVEEC can change the order of execution of the active nodes by rewriting the context graph. The priority of each active node is calculated by a causal network, and the validity of the method has been demonstrated by the experiment described in this paper.

References

[1] Tanaka, T. : PHS Data Communication System, *Journal of IPSJ*, vol. 38, no. 5, pp. 428–434 (1997).

[2] Takakura, N., Fujiwara, Y. et al.: Performance evaluation of Virtual Execution Environment Control System, *IPSJ SIG Notes*, vol. 2000, no. 117, pp. 69–79 (2000).

[3] Martin, J. and Vanlehn, J.: A Bayesian approach to cognitive assessment, Nichols, P.,Chipman, S., and Brennan, R. L. (eds.), *Cognitively Diagnostic Assessment*, Hillsdale, NJ: Erlbaum (1995).

[4] Conati, C. and Vanlehn, K.: POLA: a student modeling framework for Probabilistic On-Line Assessment of problem solving performance, *Proceedings of the 5th International Conference on User Modeling*, pp. 75–82 (1996).

[5] Conati, C., Gertner, A. S., Vanlehn, K. and Druzdzel, M. J.: On-Line Student Modeling for Coached Problem Solving Using Bayesian Networks, *Proceedings of the Sixth International Conference on User Modeling*, pp. 231–242 (1997).

[6] Collins, J. A., Greer, J. E. and Huang, S. X.: Adaptive Assessment using Granularity Hierarchies and Bayesian Nets, *Proceedings of the 3rd International Conference on Intelligent Tutoring Systems ITS'96*, pp. 569–577 (1996).

[7] Charniak, E. and Goldman, R. P.: A Bayesian model of plan recognition, *Artificial Intelligence*, Vol. 64, No. 1, pp. 53–79 (1993).

[8] Huber, M. J., Durfee, E. H. and Wellman, M. P.: The Automated Mapping of Plans for Plan Recognition, *Proceedings of the 10th Conference on Uncertainty in Artificial Intelligence*, pp. 344–351 (1994).

[9] Pynadath, D. V. and Wellman, M. P.: Accounting for Context in Plan Recognition with Application to Traffic Monitoring, *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence*, pp. 472–481 (1995).

[10] Jameson, A., Schafer, R., Simons, J. and Weis, T.: Adaptive Provision of Evaluation-Oriented Information: Tasks and Techniques, *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pp. 1886–1893 (1995).

[11] Mulken, S.: Reasoning About the User's Decoding of Presentations in an Intelligent Multimedia Presentation System, *Proceedings of the 5th International Conference on User Modeling*, pp. 67–74 (1995).