

Personalized Learning Path Generation Approach for Web-based Learning

Chin-Ming Hong¹, Chih-Ming Chen², Mei-Hui Chang²
Institute of Applied Electronic Technology
National Taiwan Normal University¹
Institute of Learning Technology
National Hualien Teachers College²
123 Hua-His Rd., Hualien, Taiwan 970, R.O.C.

Abstract - Curriculum sequencing is an important research issue for Web-based learning systems because no fixed learning paths will be appropriate for all learners. Therefore, many researchers focused on developing e-learning systems with personalized learning mechanisms to assist on-line Web-based learning and adaptively provide learning paths. However, although most personalized systems consider learner preferences, interests and browsing behaviors in providing personalized curriculum sequencing services, these systems usually neglect to consider if learner ability and the difficulty level of the recommended courseware are matched to each other. Generally, inappropriate courseware leads to learner cognitive overload or disorientation during learning processes, thus reducing learning performance. Besides, the problem of concept continuity of learning paths also needs to be considered while implementing personalized courseware generation because smoother learning paths increase learning performance, avoiding unnecessarily difficult concepts. Therefore, this paper presents a genetic-based curriculum sequencing approach to generate personalized courseware through simultaneously considering courseware difficulty level and the concept continuity of learning paths for Web-based learning. In the proposed method, a pretest through the computerized adaptive testing (CAT) is first applied to collect the incorrect learning concepts of learners. After that, the difficulty parameter of courseware and the concept relation degrees of courseware are simultaneously considered to determine the fitness function of genetic algorithm in order to generate personalized learning path for individual learner. Experiment results indicate that applying the proposed genetic-based curriculum sequencing approach for Web-based learning can generate high quality learning paths, and help learners to learn more effectively.

Keywords: *Genetic Algorithm, Personalized Learning Path, Intelligent Tutoring System, Web-based learning*

1. Introduction

Traditional teaching resources, such as textbooks, typically guide the learners to follow fixed sequences to other subject-related sections related to the current one during learning processes. Web-based instruction researchers have given considerable attention to flexible curriculum sequencing control to provide adaptable, personalized learning programs [1-8]. Curriculum sequencing aims to provide an optimal learning path to individual learners since every learner has different prior background knowledge, preferences, and often various learning goals [9-12]. In an educational adaptive hypermedia system, an optimal learning path aims to maximize a combination of the learner's understanding of courseware and the efficiency of learning the courseware [9].

Moreover, as numerous Web-based tutoring systems have been developed, a great quantity of hypermedia in courseware has created information, and cognitive overload and disorientation [13-14], such that learners are unable to learn very efficiently. To aid more efficient

learning, many powerful personalized/adaptive guidance mechanisms, such as adaptive presentation, adaptive navigation support, curriculum sequencing, and intelligent analysis of student's solutions, have been proposed [4][5][10][12]. Nowadays, most adaptive/personalized tutoring systems [3-5] consider learner/user preferences, interests, and browsing behaviors when investigating learner behaviors for personalized services. However, these systems neglect the importance of learner ability when implementing personalized mechanisms. On the other hand, some researchers emphasized that personalization should consider levels of learner knowledge, especially in relation to learning [5][12]. That is, the ability of individuals may be based on major fields and subjects. Therefore, considering learner ability can promote personalized learning performance.

Over the years, designers of Web-based learning have evolved several common lesson structures for different learning occasions. These lesson structures include the classic tutorial lessons, active-centered lessons, learner-customized tutorial lessons, knowledge-placed

tutorial lessons, exploratory tutorial lessons, and generated lessons [15]. Among the six kinds of lessons, the generated lessons aim to customize learning for those who have very specific needs and not much time or patience to complete topics they have learned [15]. The generated lessons tailors a learning sequence based on the learner's answers to questions on a pretest or questionnaire at the start of the lesson [15]. To construct a personalized learning path based on simultaneously considering courseware difficulty level and learning concept continuity during learning processes, a genetic-based curriculum sequencing approach is here presented to provide personalized curriculum sequencing generation. The proposed approach is based on a pretest to collect incorrect learning concepts of learners through the computerized adaptive testing (CAT) [16], then the genetic algorithm is employed to construct a near optimal learning path according to these incorrect response patterns of pretest. Our goal is to help learners to learn more effectively by skipping the corresponding learning concept that learner has given correct response for the testing item in computerized adaptive testing process. Since the fitness function of genetic algorithm is determined by the difficulty parameter of courseware and the concept relation degree between two successive courseware in learning paths, the proposed curriculum sequencing approach can generate high quality learning path for individual learner. Experimental results indicate that the proposed personalized curriculum sequencing approach can generate appropriate course materials to learners based on individual learner requirements, and help them to learn more effectively in a web-based environment.

2. Courseware Modeling Process

The courseware modeling process presents a detailed courseware design procedure to establish the difficulty parameters of courseware and courseware contents for personalized courseware generation. This study presents a statistics-based method derived from computerized adaptive testing (CAT) theory [16] through a conscientious test process to determine the difficulty parameters. The detailed flowchart of the courseware modeling process is illustrated as Fig. 1.

To design a course of C language programming as an example, several experienced teachers were invited as courseware experts to analyze the primary concepts for the course of C language programming in the courseware modeling process. The courseware experts designed the corresponding testing item for each learning concept. That is, the testing items are regarded as key characteristic of the corresponding learning content. Besides, about 500 examinees who have majored in the course of C language programming to join the exam, which contains 33 testing items to cover those learning concepts. According to the Item Response Theory [16][17] in CAT, their testing data

was analyzed by the statistics-based BILOG program to obtain the appropriate difficulty parameters for these testing items. After that, the web page of courseware was designed following the conveying content of the corresponding testing item. Since the content of courseware is derived from the concept of the testing item, it is assumed the difficulty of courseware equals the difficulty of the corresponding testing item. That is, each testing item in the testing item database has a corresponding courseware that conveys the learning concept of the corresponding testing item.

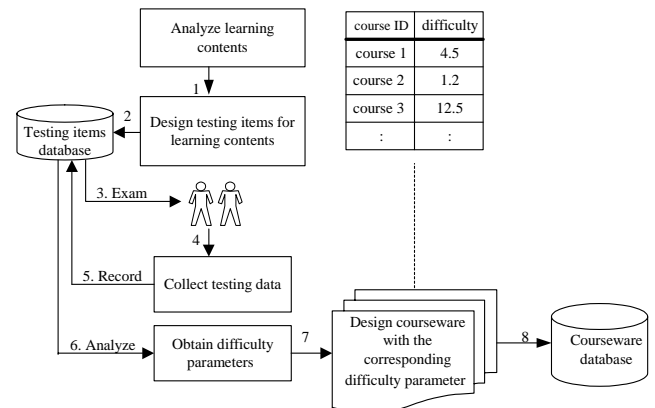


Fig.1 Courseware modeling process

3. Evaluating Concept Relation Degrees Among Courseware

In order to facilitate easier courseware concept relation analysis, all courseware in the courseware database has followed the standard of SCORM 1.2 (Sharable Content Object Reference Model) metadata information model [18-19]. Restated, each courseware in the courseware database has a corresponding XML binding file to record important SCORM metadata, which conveys the main courseware concept. Meanwhile, we also develop an interface for teachers to maintain the SCORM metadata for the relevant courseware. In order to generate a near optimal learning path for learner based on the results of pretest, these SCORM metadata are applied to calculate the concept relation degrees among courseware by using Chinese natural language processing [20] and information retrieval [21] methods. Figure 2 illustrates the maintained interface of SCORM metadata. Next, we explain how the concept relation degrees can be obtained for personalized courseware generation.

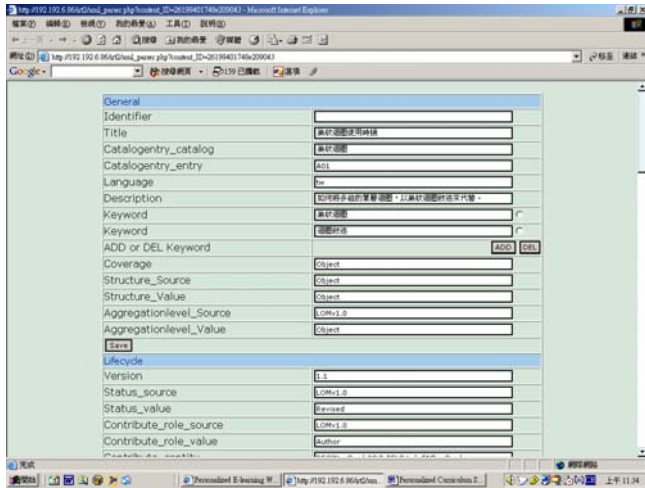


Fig. 2 The maintained interface of SCORM metadata in our system

3.1 Metadata Preprocessing

First, two metadata fields of the corresponding XML binding file of courseware are selected to represent the conveyed learning concept for a courseware. They are keyword and description fields in the SCORM 1.2 metadata information model shown as Fig. 2, respectively. In order to calculate the concept relation degrees for personalized courseware generation, metadata preprocessing is required because the description field in the SCORM 1.2 metadata information model is described by Chinese natural language in this research. Thus, the first phase of metadata preprocessing aims to perform word segmentation by using Chinese knowledge information processing (CKIP) [20] in order to describe the metadata field of the corresponding XML binding file of courseware so that separated linguistic terms can be obtained. The second phase of metadata preprocessing filters out non-textual words (e.g., numeric data, symbols, notation and ASCII drawings) and one-word terms because they do not carry any usable information for calculating concept relation degrees.

3.2 Estimation of Concept Relation Degree

To estimate the concept relation degree between two courseware packages, the vector space model [21] is applied to represent each courseware as vectors in a multidimensional Euclidean space. Each axis in this space corresponds to a linguistic term obtained from word segmentation process. The coordinate of the i^{th} courseware in the direction corresponding to the k^{th} linguist term can be determined as follows:

$$w_{ik} = tf_{ik} \times \log \frac{N}{df_k} = tf_{ik} \times IDF \quad (1)$$

where w_{ik} represents the importance/weight of the k^{th} term in the i^{th} courseware, tf_{ik} is term frequency of the k^{th} term, which appears in the i^{th} courseware; N

denotes the total number of courseware in a course unit, df_k is the document frequency of the k^{th} term, which appears in a course unit.

Assume that there are total m terms under union of all linguistic terms of the i^{th} courseware and j^{th} courseware. The concept relation degree for the i^{th} and j^{th} courseware can be found by using the cosine-measure, and listed as follows:

$$r_{ij} = \frac{\sum_{h=1}^m w_{ih} w_{jh}}{\sqrt{\sum_{h=1}^m w_{ih}^2 \sum_{h=1}^k w_{jh}^2}} \quad (2)$$

where $c_i = \langle w_{i1}, w_{i2}, \dots, w_{ik}, \dots, w_{im} \rangle$ and $c_j = \langle w_{j1}, w_{j2}, \dots, w_{jk}, \dots, w_{jm} \rangle$, respectively, represent the vectors in a multidimensional Euclidean space for the i^{th} and j^{th} courseware, r_{ij} denotes the concept relation degree between the i^{th} and j^{th} courseware.

Assume that there are totally n courseware in the courseware database, the concept relation matrix for all courseware can be expressed by the matrix \mathbf{R} , and listed as follows:

$$\mathbf{R} = \begin{matrix} & \begin{matrix} c_1 & c_2 & \Lambda & c_n \end{matrix} \\ \begin{matrix} c_1 \\ c_2 \\ M \\ c_n \end{matrix} & \begin{bmatrix} r_{11} & r_{12} & \Lambda & r_{1n} \\ r_{21} & r_{22} & K & r_{2n} \\ M & M & M & M \\ r_{n1} & r_{n2} & \Lambda & r_{nm} \end{bmatrix}_{n \times n} \end{matrix} \quad (3)$$

4. Personalized Courseware Generation Based On Genetic Algorithm

This section explains how to generate the personalized learning path utilizing the genetic algorithm [22] for web-based instruction.

4.1 Genetic Algorithm for Personalized Courseware Generation

4.1.1 Definition of Chromosome Strings

In this study, a serial number is assigned to each courseware from 1 to n if there are totally n courseware in the courseware database for learning path generation. Thus, the assigned serial number of each courseware is directly combined with the serial number of successive courseware as strings to represent the generated learning path for the genetic algorithm. The whole individual represented by chromosome of all courseware parameters for the genetic algorithm is illustrated as Figure 3.

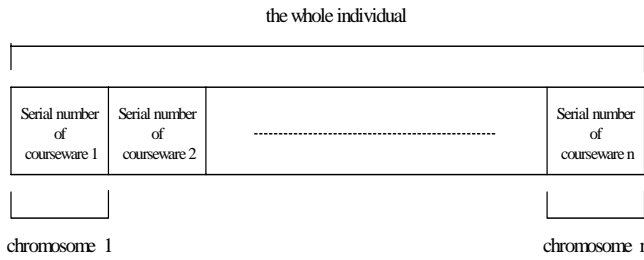


Fig. 3 The individual strings combined by the serial number of courseware for the genetic algorithm

4.1.2 Initial Population Size

Generally, the initial population size can be determined according to the complexity of the solved problem. A larger population size will reduce the searching speed of genetic algorithm, but it will increase the probability of finding high quality solution. To construct high quality learning path for individual learner, the initial population size is chosen as one hundred for personalized courseware generation in this study.

4.1.3 Selecting Fitness Function

Fitness function is a performance index that it is applied to judge the quality of generated learning path for the genetic algorithm. In order to generate personalized learning path for individual learner based on the pretest results, the difficulty parameters of courseware and the concept relation degrees of courseware are simultaneously considered to determine the fitness function. In our method, the learning path constructed by the genetic algorithm only considers the mapped courseware that learner gives incorrect pretest results. Moreover, the courseware with the smallest difficulty parameter is selected as the first courseware ranked in the constructed learning path. Therefore, the fitness function is formulated as follows:

$$f = \sum_{i=2}^n ((1-w) \times r_{(i-1)i} + w \times (1-b_i)) \quad (4)$$

where f is the fitness function for the genetic algorithm, $r_{(i-1)i}$ represents the concept relation degree of the $(i-1)^{th}$ courseware with the i^{th} courseware in the constructed learning path, b_i is the difficulty parameter of the i^{th} courseware, w is a adjustable weight, and n stands for the total number of courseware considered for personalized courseware generation.

4.1.4 Reproduction Operation

In the reproduction operation, the chromosome with larger fitness function value will have higher probability to reproduce next generation. This operation aims to choose good chromosome to achieve the goal of gene evolution. The most common used method of reproduction operation is the weighted roulette selection [22]. In this study, we

also use this method to perform reproduction operation.

4.1.5 Crossover Operation

In the crossover operation, the randomly selected two serial numbers of chromosomes in two individuals exchange entire chromosome each other by probability decisions. This operation aims to combine two parent chromosomes to generate better child chromosomes. In our study, the uniform crossover operation [22] is used, but to avoid the generated learning path has duplicate serial number of chromosome or the serial number of courseware is over the total number of courseware after the crossover operation is performed. The crossover operation will exchange the whole chromosome by probability decision. Figure 4 illustrates the used crossover operation. Restated, this performed crossover operation can guarantee to obtain a logical learning path. In this paper, the probability of mutation is set to be 0.9.

4.1.6 Mutation Operation

In mutation operation, the randomly selected two chromosomes in the selected individual are forced to exchange the chromosome. The mutation operation will evolve some new individuals that might not be produced by the operations of reproduction and crossover. Generally, a lower probability of mutation can guarantee the convergence of genetic algorithm, but it may lead to poor solution quality. In contrast, a higher probability of mutation may lead to the phenomenon of random walk for the genetic algorithm. In this paper, the probability of mutation is set to be 0.001.

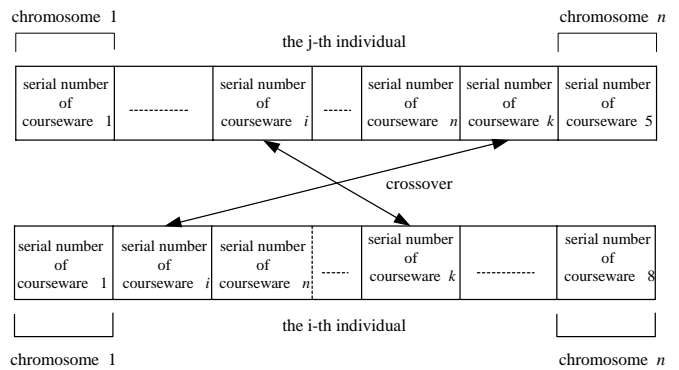


Fig. 4 Crossover operation

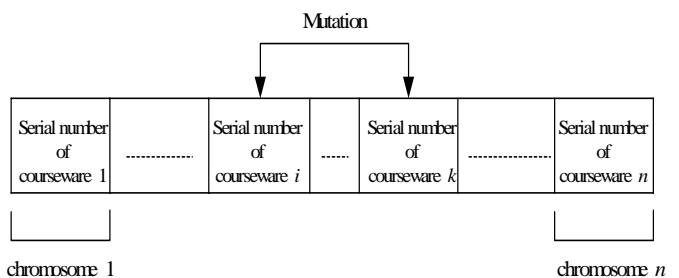


Fig.5 Mutation operation

4.1.7 Stop Criterion

The genetic algorithm runs the reproduction, crossover, mutation and replacement operations repeatedly until it meets the stop criterion. The stop criterion is set to be 200 generations because this criterion can obtain satisfied learning paths for personalized e-learning service.

4.2 Procedure of Personalized Courseware Generation

In sum, the procedure of the proposed genetic-based courseware generation approach can be summarized as follows:

- Step 1.** Learner performs a pretest based on computerized adaptive testing for personalized courseware generation
- Step 2.** The system collects the incorrect testing items in the pretest and their corresponding courseware in the courseware database
- Step 3.** The corresponding courseware with the smallest difficulty parameter among the incorrect testing items is selected as initial program for personalized courseware generation
- Step 4.** The system generates a near optimal learning path for individual learner utilizing the genetic algorithm according to the incorrect response testing items
- Step 5.** Learner performs personalized e-learning according to the generated learning path
- Step 6.** Terminate the learning process or return Step 1 for further learning

5. Experiments

To verify the quality of generated learning path for personalized web-based instruction, some university students who have majored in the course of C language programming were invited to test this system. The experimental results are described as follows.

5.1 Experimental Environment

The course modeling process mentioned in Section 2 is used to determine the difficulty parameter of each piece of courseware. Restated, courseware organized on a single Web page is the smallest course element in the proposed personalized courseware generation approach. In our experiments, the course unit, “Loop”, in the course category, “C Language Programming”, is used to generate personalized learning path, which includes many courseware with various levels of difficulty to convey the concept of the “Loop”. Assume that a pretest in the course unit “Loop” is performed by a learner, and occurs totally 12 incorrect testing items. Assume that table 1 illustrates the concept relation degrees of corresponding courseware that learner gives incorrect testing item responses.

lists the titles of corresponding courseware and their difficulty parameters that the learner gives incorrect testing item responses.

Table 1. The concept relation degrees for the incorrect testing items

r_{ij}	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15
C1	1	0.1054	0.4214	0	0.3535	0.3149	0	0.4140	0.3746	0.6123	0.5685	0.1020	0.3311	0.4107	0.0912
C2	0.1054	1	0	0.0496	0.2236	0.3984	0	0.4364	0.2369	0.1291	0.1438	0.1291	0.3769	0	0.2309
C3	0.4214	0	1	0	0	0.2124	0	0.3102	0	0.4588	0.3834	0	0.2233	0.9233	0
C4	0	0.0496	0	1	0.0555	0.0297	0.2800	0	0.0441	0	0	0.0481	0.0624	0	0.1291
C5	0.3535	0.2236	0	0.0555	1	0.2672	0	0.2927	0.8609	0.4330	0.4824	0.2165	0.3278	0	0.1936
C6	0.3149	0.3984	0.2124	0.0297	0.2672	1	0	0.5738	0.2832	0.3086	0.3438	0.1543	0.8761	0.2415	0.2415
C7	0	0	0	0.2800	0	0	1	0	0	0	0	0.1212	0	0	0.1084
C8	0.4140	0.4364	0.3102	0	0.2927	0.5738	0	1	0.3102	0.5070	0.4394	0.0845	0.5758	0.3023	0.0755
C9	0.3746	0.2369	0	0.0441	0.8609	0.2832	0	0.3102	1	0.4588	0.5112	0.2294	0.3349	0	0.2052
C10	0.6123	0.1291	0.4588	0	0.4330	0.3086	0	0.5070	0.4588	1	0.9284	0.125	0.3244	0.4472	0.1118
C11	0.5685	0.1438	0.3834	0	0.4824	0.3438	0	0.4394	0.5112	0.9284	1	0.1392	0.3614	0.3737	0.1245
C12	0.1020	0.1291	0	0.0481	0.2165	0.1543	0.1212	0.0845	0.2294	0.125	0.1392	1	0.2027	0.055	0.3913
C13	0.3311	0.3769	0.2233	0.0624	0.3278	0.8761	0	0.5758	0.3349	0.3244	0.3614	0.2027	1	0.2539	0.2176
C14	0.4107	0	0.9233	0	0	0.2415	0	0.3023	0	0.4472	0.3737	0.0559	0.2539	1	0.1
C15	0.0912	0.2309	0	0.1291	0.1936	0.2415	0.1084	0.0755	0.2052	0.1118	0.1245	0.3913	0.2176	0.1	1

5.2 Experimental Results

Based on the corresponding concept relation degrees and difficulty parameters listed in Table 1 and 2 that learner gives incorrect testing item responses, the genetic algorithm was employed to construct high quality of personalized learning path according to the designed fitness function. Table 3 illustrates the generated learning path by the proposed genetic algorithm. We find that the generated learning path recommends smooth learning path to learner under simultaneously considering the difficulty parameters of courseware. Restated, the learning concepts with high concept relation degree will be successively recommended during the learning process under simultaneously considering the difficulty parameters of courseware. This is very beneficial for learner because it can guide learner to achieve more effective learning. Besides, Figure 6 shows the convergence curve of fitness function. This result demonstrates that the proposed genetic algorithm can indeed generate high quality learning path for personalized learning.

Table 2. The corresponding difficulty parameter for each courseware in the “Loop” unit

Courseware	Title of Courseware	Difficulty Parameter
C1	Example 3 of nested loop	1.6
C2	Flowchart of “do while” instruction	0.3
C3	Example 1 of nested loop	0.9
C4	Introduction of syntax of “goto” instruction	0.3
C5	Using opportunity of nested loop	1.2

C6	Introduction of syntax of "do while" instruction	0.5
C7	Example 1 of "goto" instruction	1.1
C8	Example 1 of "while" instruction	1.2
C9	Definition of nested loop	0.8
C10	Example 2 of "for" instruction	1.7
C11	Example 1 of "for" instruction	-0.3
C12	Example 2 of "continue" instruction	0.7
C13	Introduction of syntax of "while" instruction	0.3
C14	Example 2 of nested loop	2.1
C15	Example 2 of "break" instruction	0.4

Table 3. The generated learning path by GA

Learning Path		Difficult Parameter	Concept Relation Degree between Two Successive Courseware
C11	Example 1 of "for" instruction	-0.3	-----
C10	Example 2 of "for" instruction	1.7	0.9284
C3	Example 1 of nested loop	0.9	0.4588
C14	Example 2 of nested loop	2.1	0.9233
C1	Example 3 of nested loop	1.6	0.4107
C9	Definition of nested loop	0.8	0.3746
C5	Using opportunity of nested loop	1.2	0.8609
C13	Introduction of syntax of "while" instruction	0.3	0.3278
C6	Introduction of syntax of "do while" instruction	0.5	0.8761
C8	Example 1 of "while" instruction	1.2	0.5738
C2	Flowchart of "do while" instruction	0.3	0.4364
C15	Example 2 of "break" instruction	0.4	0.2309
C12	Example 2 of "continue" instruction	0.7	0.2294
C7	Example 1 of "goto" instruction	1.1	0.3913
C4	Introduction of syntax of "goto" instruction	0.3	0.2800

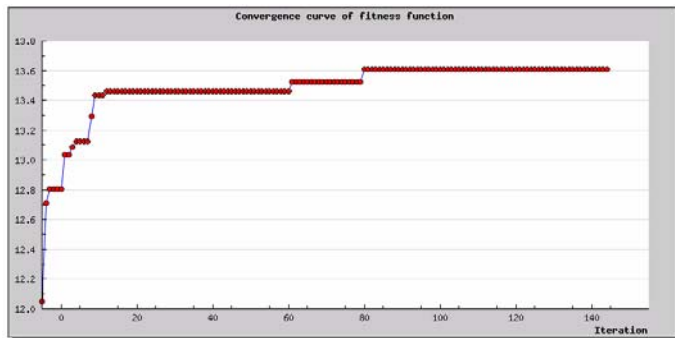


Fig.6 Convergence curve of fitness function

6. Conclusion

This study proposes a personalized courseware generation approach based on the genetic algorithm for personalized learning path generation. The proposed learning path generation approach can simultaneously consider courseware difficulty level and the concept continuity of successive courseware while implementing personalized curriculum generation in learning processes. The proposed method can provide personalized Web-based instruction according to incorrect testing item responses in a pretest. Experimental results indicate that the proposed method can

precisely provide personalized curriculum sequencing based on difficulty parameter and concept continuity of successive courseware, and moreover can accelerate learner learning effectiveness. Meanwhile, the generated learning path customizes learning for those who have very specific needs and not much time or patience to complete topics they have learned.

References:

- [1] Mia K. and Beverly Park Woolf, "Curriculum Sequencing in a Web-based Tutor," *Proceedings of Intelligent Tutoring Systems*, 1998.
- [2] Binshan Lin and Chang-Tseh Hsieh, "Web-based Teaching and Learner Control: a Research Review," *Computers & Education*, vol. 37, pp. 377-386, 2001.
- [3] Myung-Geun Lee, "Profiling Students' Adaptation Styles in Web-based Learning," *Computers & Education*, vol. 36, pp. 121-132, 2001.
- [4] Tiffany Ya Tang and Gordon Mccalla, "Smart Recommendation for Evolving E-Learning System," *11th International Conference on Artificial Intelligence in Education, Workshop on Technologies for Electronic Documents for Supporting Learning*, July 20-24 July, Sydney, Australia, pp. 699-710, 2003.
- [5] Kyparisia A. Papanikolaou and Maria Grigoriadou, "Towards New Forms of Knowledge Communication: the Adaptive Dimension of a Web-based Learning Environment," *Computers & Education*, vol. 39, pp. 333-360, 2002.
- [6] Hueyching Janice Jih, "The Impact of Learners' Pathway on Learning Performance in Multimedia Computer Aided Learning," *Journal of Network and Computer Applications*, vol. 19, pp. 367-380, 1996.
- [7] Changjie Tang; Lau, R.W.H.; Qing Li; Huabei Yin; Tong Li; Kilis, D., "Personalized Courseware Construction based on Web Data Mining," *Proceedings of the First IEEE International Conference on Web Information Systems Engineering*, vol. 2, 19-21 June 2000, pp. 204-211.
- [8] Peter Brusilovsky, John Eklund and Elmar Schwarz, "Web-based Education for All: a Tool for Development Adaptive Courseware," *Computer Networks and ISDN Systems*, vol. 30, pp. 291-300, 1998.
- [9] Roland Hübscher, "Logically Optimal Curriculum Sequences for Adaptive Hypermedia Systems," *International Conference on Adaptive Hypermedia and Adaptive Web-based System*, Springer, 2000, pp. 121-132.
- [10] Gerhard Weber and Marcus Specht, "User Modeling and Adaptive Navigation Support in WWW-based Tutoring Systems," *Proceedings of the Sixth International Conference on User Modeling*, 1997, pp. 289-300.

- [11] Brusilovsky P. and Vassileva, J., "Course Sequencing Techniques for Large-Scale Web-based Education," *International Journal of Continuing Engineering Education and Lifelong Learning*, 13 (1-2), pp. 75-94, 2003.
- [12] Chih-Ming Chen, Hahn-Ming Lee and Ya-Hui Chen, "Personalized E-Learning System Using Item Response Theory," *Computers & Education*, vol. 44, no. 3, pp. 237-255, 2005.
- [13] Berghel H., "Cyberspace 2000: Dealing with Information Overload," *Communications of the ACM*, vol. 40, no. 2, pp.19-24, Feb. 1997.
- [14] Borchers A., Herlocker J., Konstanand J., Riedl J., "Ganging Up on Information Overload," *Computer*, vol. 31, issue: 4, pp. 106-108, Apr. 1998.
- [15] William Horton, *Designing Web-Based Training*, John Wiley & Sons, Inc., 2000.
- [16] Hsu, T. C. and Sadock, S. F., *Computer-assisted Test Construction: A State of Art*, TME report 88, Princeton, New Jersey, Eric on Test, Measurement, and Evaluation, Educational Testing Service, 1985.
- [17] Baker and Frank B., *Item Response Theory: Parameter Estimation Techniques*, New York: Marcel Dekker, 1992.
- [18] Advanced Distributed Learning, Available at <http://www.adlnet.org>.
- [19] ADL, SCORM version 1.2-The SCORM Content Aggregation Model, 2001, Advanced Distributed Learning Initiative. from WWW: http://www.adlnet.org/screens/shares/dsp_displayfile.cfm?fileid=476
- [20] Chinese Knowledge Information Processing (CKIP), Available at <http://blackjack.iis.sinica.edu.tw/~ma/uwextract/>.
- [21] Frakes William B., Baeza-Yates R., *Information Retrieval Data Structures and Algorithms*, Englewood Cliffs, N. Prentice Hall, 1992.
- [22] Rothlauf Franz, *Representations for Genetic and Evolutionary Algorithms*, Heidelberg, Physica-Verlag, New York, 2002.