Dynamic Calculation of Concept Difficulty Based on Choquet Fuzzy Integral and the Learner Model

Ahmad Kardan and Roya Hosseini

Department of Computer Engineering and IT, Amirkabir University of Technology
424 Hafez Ave, Tehran, Iran, 15875-4413
{aakardan,hosseini.ec}@aut.ac.ir

Abstract. Adaptation and personalization of information in E-learning systems plays a significant role in supporting the learner during the learning process. Most personalized systems consider learner preferences, interest, and browsing patterns for providing adaptive presentation and adaptive navigation support. However, these systems usually neglect to consider the dependence among the learning concept difficulty and the learner model. Generally, a learning concept has varied difficulty for learners with different levels of knowledge. Hence, to provide a more personalized and efficient learning path with learning concepts difficulty that are highly matched to the learner's knowledge, this paper presents a novel method for dynamic calculation of difficulty level for concepts of the certain knowledge domain based on Choquet Fuzzy Integral and the learner knowledge and behavioral model. Finally, a numerical analysis is provided to illustrate the proposed method.

Keywords: E-learning, Difficulty Level, Choquet Fuzzy Integral, Learner Model, Personalization, Learning Concept.

1 Introduction

Rapid progress of the Internet as well as high adoption of Information Technology in educational context, have positively influenced educational processes. Moreover, significant role of Information and Communication Technologies (ICTs) in improvement of learning has greatly changed the traditional approaches on education. In this regard, educational tools and methods have been greatly changed to support learning i.e. gaining knowledge and experience through instruction or study.

Computer-Based Training (CBT) is among the staple tools in the learning process. According to the predicted results and based on literature researches, E-learning is a new form of CBT with the potentiality to increase the efficiency and quality of learning [1]. However, as numerous Web-Based tutoring systems have been developed, a great quantity of hypermedia in courseware has created information, cognitive overload and disorientation, such that learners are unable to learn very efficiently [2]. To overcome this problem, Adaptive Educational Hypermedia Systems (AEHS) were introduced in the late 1990s with the aim of increasing hypermedia efficiency by personalization. AEHS adapts the presentation of information to the learner model by

H. Cherifi, J.M. Zain, and E. El-Qawasmeh (Eds.): DICTAP 2011, Part II, CCIS 167, pp. 517–530, 2011.

tracking the user performance during his/her navigation and then updating the information of the learner model accordingly [3].

Personalized education not only supports learners to learn better by providing different strategies to create various learning experiences, but also considers teachers' tutorial needs in designing instructional packages [4]. Web-Based education considers a wide range of learners with differences in the knowledge level, age, experience, and cultural backgrounds. Hence, personalization in this context is of high significance [5].

According to [2] adaptation is quintessential in Web-Based education for mainly two reasons. First, most of the Web-Based applications have much wider variety of users with different interests than standalone applications. Second, the user is usually alone while working with a Web-Based tutoring system.

So far, different mechanisms have been introduced for the personalization and adaptation of learning process such as Adaptive Presentation, Adaptive Navigation Support, Curriculum Sequencing, Intelligent Analysis of Student Solutions, and Problem Solving Support Technologies [6]. ActiveMath [7], Personal Reader [8], iClass [9], PIMS [10], PELS [11], and ELM-ART [12] are example of systems which use these techniques to provide adaptive learning environment.

Motivation and problem definition: Currently, most adaptive tutoring systems consider learner preferences, interests, and browsing patterns when investigating learner behavior for personalized services. One important fact that is mostly ignored for providing personalized services is the dependence among the difficulty level of learning concepts and the learner's knowledge. To illustrate it more, a learning concept may have different difficulty for learners with different knowledge level. In the current studies, difficulty level of concepts is determined statically and independent of the learner model by instructors. Obviously, dynamic problem difficulty is better matched to the leaner's knowledge level and hence, can ease the comprehension of learning concepts.

The related work in [13] aims to calculate problem difficulty with regards to the learner model. The achieved results of [13] proves the fact that dynamic computed problem difficulty performs well for a wide range of learners, whereas static problem complexity performs well for students of intermediate ability, but rather badly for beginners and advanced learners.

The main drawback of the proposed method in [13] is the use of constraint based modeling for calculation of problem difficulty. According to [14] this model requires a huge set of constraints to be defined and this adds to its complexity. Moreover, it is not always possible to define constraint for some knowledge domains.

Findings: The most significant finding of this research is in presenting a method for calculation of concept difficulty with the following strengths:

- Constraint-Independent learner model which consists of the learner knowledge and behavioral model and is described in section 3.1.
- Constraint-Independent knowledge domain by using OWL which will be described in section 3.2.
- Dynamic calculation of learning concept difficulty by using Choquet fuzzy integral which will be described in section 3.3.

The novelty of this research is in using Fuzzy Integral and specifically Choquet Fuzzy Integral for calculation of learning concept difficulty. Leaner model will be used for retrieving some necessary parameters. OWL (Ontology Web Language) is also used for classification of concepts in the certain knowledge domain and retrieval of other complementary parameters.

So, the organization of this paper is as follows: The following section presents the architecture of the system for the proposed method. The method is addressed in section 3. Section 4 provides a numerical analysis for the proposed method. Finally, Section 5 provides a conclusion for this work.

2 System Architecture

This section presents the architecture of the system that dynamically calculates the learning concept difficulty based on Choquet Fuzzy Integral and the learner model. As it is shown in Fig. 1, this system consists of six major units as well as five major databases. The six major units are: Ontology Modeling Unit, Learner Interface Unit, Dynamic Concept Difficulty Calculation (DCDC) Unit, Feedback Unit, Content Recommendation Unit, and Learning Content Repository. The five major databases are: Teacher Account Database, Knowledge Domain Ontology Database, Learner Account Database, Concept Difficulty Database, and Learner Model Database.

The Ontology Modeling Unit builds the subject ontology by using the concepts and their relations that are provided by the teacher. The processes in this unit are performed in an Off-line manner. The Learner Interface Unit aims to provide a flexible learning interface for learners to interact with the Feedback Unit and the Content Recommendation Unit. DCDC Unit dynamically calculates concept difficulty based on the learner model and the knowledge domain ontology. The Feedback Unit aims to collect learner explicit feedback information from the Learner Interface Unit and stores it in the Learner Model Database. Content Recommendation Unit is in charge of recommending suitable contents to learner based on the learner model and difficulty of learning concepts. The numbers on the arrows in Fig. 1 show the steps related to data flow within the system. These steps are as follows:

Step 1. The Teacher logs in the system by providing his/her account information.

Step 2. The teacher login information is compared to the information in the Teacher Account Database. If the information is valid, then teacher models the ontology for the knowledge domain.

Ontology Modeling Process: The teacher models the knowledge domain by providing a set of concepts with their relations.

Step 3. The output of the Ontology Modeling Process, namely the Knowledge Domain Ontology, is stored in the Knowledge Domain Ontology Database.

Step 4. The relations among the Knowledge Domain Ontology and the Concepts Hierarchy are determined by the teacher. Each node in the Knowledge Domain Ontology is related to at least one concept in the Concepts Hierarchy.

Step 5. The learner logs in the system by providing his/her user account information.

Step 6. The user login information is compared to the information in the Learner Account Database.

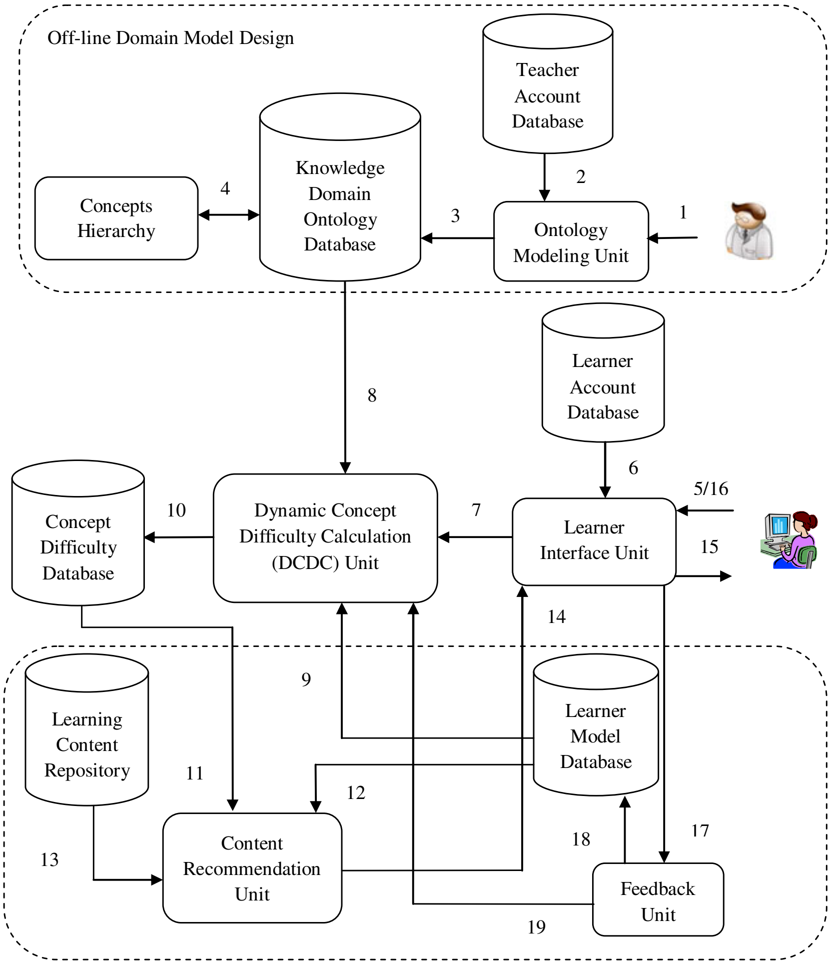

Fig. 1. The System Architecture

Step 7. If the information is valid, then the user selects the concept to be learned from the Concepts Hierarchy and step 8 is followed. Otherwise, the learner remains on Step 5.

Step 8. To calculate the concept difficulty for the concepts that are related to the user selected concept, knowledge domain ontology is used as one of the inputs of the DCDC Unit.

Step 9. The learner information is the other input that is used by the DCDC unit.

DCDC Process: This process uses the inputs provided by Step 8 and Step 9 and then calculates the concept difficulty by using Choquet Fuzzy Integral.

Step 10. Concept difficulty is temporarily stored in the Concept Difficulty Database.

Step 11. To recommend suitable contents, the information of the Concept Difficulty Database serves as one of the inputs of the Content Recommendation Unit.

Step 12. The information in the learner model is the other input that is used by the Content Recommendation Unit.

Step 13. The information required for content recommendation is also related to the contents in the Learning Content Repository. This information serves as the third input of the Content Recommendation Unit.

Content Recommendation Process: Using the information provided by Step 11, Step 12, and Step 13, this unit selects suitable learning contents for learner.

Step 14. The list of recommended contents is transferred to the Learner Interface Unit.

Step 15. The Learner Interface Unit presents the recommended contents to learner.

Step 16. The Learner Interface Unit tracks the learner navigation during the learning process. Also, in the case of the test, this unit receives the learner's response.

Step 17. The Leaner Interface Unit sends the received Feedback to the Feedback Unit for updating the information in the learner model and recommending suitable contents.

Step 18. In this step, the Feedback Unit updates the information in the learner model based on the received feedback.

Step 19. To recommend the learning content based on the learner feedback, concept difficulty level is calculated again. Steps 7 to Step 19 are repeated until the learner learns the selected concept from the Concepts Hierarchy.

3 Methodology

This section presents the proposed method for dynamic calculation of learning concept difficulty by using Choquet Fuzzy Integral and the learner model. In this regard, it is quintessential to first determine the influential parameters for calculation of the concept difficulty. Second, the knowledge domain should be modeled. Therefore, the following subsections first describe each of these steps and finally describe the proposed method.

3.1 Parameters

Dynamic calculation of difficulty level for a concept relies on the information related to the learner and the knowledge domain. Therefore, both the learner and the domain model are two influential elements in determining the learning concept difficulty. The learner model consists of two parts, namely the learner knowledge model and the learner behavioral model. The former models the learner's knowledge in concepts of the knowledge domain and the latter models the learner behavior and more specifically the number of time the learner studies each concept of the knowledge domain. The details of these parameters are as follows:

Domain Model: The domain model consists of the concepts as well as the relations between them. In this paper, the concept that the difficulty level is calculated for it is called the Main Concept. Obviously, the increase in the number of the relations between the Main Concept and other concepts has direct impact on the Main Concept difficulty. Semantic relationship between the Main concept and other concepts of the certain knowledge domain can be described in two forms: 1- Prerequisite Concepts that are necessary to perceive the Main Concept, and 2- Concepts that are related to the main concept and are part of the same Sub-domain. These concepts are called Related Concepts.

Learner Behavioral Model: This model stores the information about the learner's activity, i.e. the number of times the certain concept is studied. Hence, the learner activity in the Main Concept, its Prerequisite Concepts, and its Related Concepts is important in determining the difficulty level of the Main Concept. The learner study activity has direct impact on the concept difficulty. As the difficulty of the Main concept increases, it is more probable for learner to study it again.

Leaner Knowledge Model: This model reflects the learner's knowledge in the concepts of the knowledge domain. This model is used to determine the learner's knowledge in the Main concept, its Prerequisite Concepts, and its Related Concepts. The lower the learner's knowledge in these concepts, the more the difficulty level of the Main Concept will be.

All the identified parameters in the *Domain Model* and the *Learner Behavioral Model* have direct relationship with the concept difficulty. However, the learner Knowledge level has inverse relationship with the Main Concept difficulty. Hence, to ease the calculation of the Main Concept difficulty, the learner's wrong answer to the tests related to the Main concept, its Prerequisite Concepts, and its Related Concepts is considered as the input parameter in the proposed method. Hence, all the parameters will have direct relationship with the Main Concept.

So, the influential parameters for determining the concept difficulty can be summarized as follows:

- Number of the relations between the Main Concept, its Prerequisite Concepts, and its Related Concepts.
- Learner's wrong answers to the tests: related to Main Concept, its Prerequisite Concepts, and its Related Concepts.
- Learner's study activity: in the Main Concept, its Prerequisite Concepts, and its Related Concepts.

In this research, an Overlay Model is used for modeling the learner's knowledge. To this end, the knowledge domain needs to be modeled. Ontology and Concept Map are two common tools that are widely deployed for this purpose. Since the Ontology models the hierarchical structure among the concepts, it is used to model the knowledge domain in this research. The following section presents the steps required to model the knowledge domain by using OWL which is a standard ontology language.

3.2 Knowledge Domain Modeling Using OWL

The proposed method uses ontology to model the knowledge domain. Ontology describes the concepts in the domain and also the relationships that hold between those concepts. Different ontology languages provide different facilities. The most recent development in standard ontology languages is OWL (Ontology Web Language) from the World Wide Web Consortium (W3C) [15]. In this paper, OWL is used for modeling the knowledge domain. The modeling process consists of the following steps:

Step 1. An expert determines the subject i.e. the domain of knowledge.
Step 2. Concepts in the knowledge domain are defined by an expert.
Step 3. The relationships between the concepts of the knowledge domain are determined.
Step 4. OWL is used to model the knowledge domain. The output of this step is the knowledge domain ontology.

After the preparation of the knowledge domain ontology, the difficulty of knowledge domain concepts can be calculated by using the parameters introduced in section 3.1. The following section, presents the proposed method.

3.3 Concept Difficulty Calculation Using Choquet Fuzzy Integral

After determining the value of influential parameters on difficulty of a concept, it is necessary to aggregate these values to a single value that represents the concept difficulty. Weighted Arithmetic Mean and the Regression methods are among the most common aggregation operators. However, none of these operators is able to model in some understandable way an interaction between the input parameters [16]. Hence, these operators are not suitable for calculation of concept difficulty.

The Choquet Fuzzy Integral is an operator that is used for the aggregation of the interdependent parameters based on the fuzzy measure. According to [17] the suitability of this Integral is proved for the Real-Time applications. Therefore, Choquet Fuzzy Integral can improve the response time of an E-learning system [18]. In this paper, Choquet Fuzzy Integral is used to aggregate the influential parameters on concept difficulty. The definition of Choquet Fuzzy Integral is as follows:

Definition: Choquet Fuzzy Integral is an integral that uses fuzzy measure to aggregate the set of input parameters. According to [19] it is defined as Eq. 1:

$$E_g(h) = \int_X h(.) \circ g(.) = \sum_{i=1}^{n} [h(x_i) - h(x_{i-1})] g(A_i) \ . \tag{1}$$

where $h(x_1) \le h(x_2) \le \ldots\ldots\ldots\ldots \le h(x_n)$ and $h(x_0) = 0$. The definition of each variable used in Eq. 1 is provided herein.

n : Number of the input parameters. In this research, $n = 3$.

X : Input parameters of the Choquet Fuzzy Integral. This set is shown as $X = \{x_1, x_2, \ldots, x_n\}$.

x_1 : Represents the number of the relations between the Main Concept, its Prerequisite Concepts, and its Related Concepts.

x_2 : Represents Learner's wrong answers to the tests related to the Main Concept, its Prerequisite Concepts, and its Related Concepts.

x_3: Represents Learner's activities during his/her course of study regarding a Main Concept, any Prerequisite for the Main Concept, and also Related Concepts. Hence, the parameter x_3 is modeled by counting the number of times the Main Concept, its Prerequisite Concepts, and its Related Concepts are studied by the learner.

h : The function that determines the value of the input variables. For example, $h(x_i)$ is the value of x_i .

g : The λ-fuzzy measure which is defined as $g : P(X) \to [0,1]$ such that:

$$g(\phi) = 0, g(X) = 1 \ . \tag{2}$$

$$\text{If } A, B \in P(X) \text{ and } A \subset B, \text{ then } g(A) \le g(B) \ . \tag{3}$$

$$A, B \subset X \ , \ A \cap B = \phi \ , \ g(A \cup B) = g(A) + g(B) + \lambda g(A) g(B) \ . \tag{4}$$

for some fixed $\lambda > -1$.

The value of λ is found from the equation $g(X) = 1$ that is equivalent to solve the Eq. 5:

$$g_\lambda(X) = \frac{1}{\lambda}\left(\prod_{i=1}^{n}(1 + \lambda g_i) - 1\right) \ , \lambda \ne 0 \ . \tag{5}$$

A_i : Set of input parameters in the form of $A_i = \{x_i, x_{i+1}, \ldots.., x_n\}$.

$g(A_i)$: This value is recursively computed by Eq. 6 and Eq. 7:

$$g(A_n) = g(\{x_n\}) = g_n \ . \tag{6}$$

$$g(A_i) = g_i + g(A_{i+1}) + \lambda g_i g(A_{i+1}) \ . \quad 1 \le i < n \tag{7}$$

This research uses the Choquet Fuzzy Integral to calculate the difficulty level of the concepts in the knowledge domain through the following steps:

Step 1. Having logged in the system, the learner selects the concept which he/she wants to learn from the Concepts Hierarchy.

Step 2. The system searches and selects the set of concepts that are related to the learner's selected concept.

Step 3. Regarding the learner knowledge model, the concepts that the learner has already learned are removed from the concepts obtained in Step 2. The result is the set of Main Concepts that their difficulty needs to be calculated by Choquet Fuzzy Integral. For each of the Main Concept, Step 4 through Step 13 should be followed:

Step 4. The first input parameter x_1 is determined by counting the number of Prerequisite and Related Concepts of the Main Concept in the Knowledge Domain Ontology.

Step 5. The Second input parameter x_2 is determined by retrieving the Learner's wrong answers to the tests related to Main Concept, its Prerequisite Concepts, and its Related Concepts from the learner knowledge model.

Step 6. The third input parameter x_3 is determined by retrieving the Learner's study activity in the Main Concept, its Prerequisite Concepts, and its Related Concepts from the learner behavioral model.

Step 7. The values obtained in Step 4, Step 5, and Step 6 are aggregated by using Choquet Fuzzy Integral as mentioned in Eq. 1. $h(x_1)$ is the number of Prerequisite and Related Concepts of the Main Concept; $h(x_2)$ is Learner's wrong answers to tests related to the Main Concept, its Prerequisite Concepts, and its Related Concepts; and $h(x_3)$ is the learner's study activity in the Main Concept, its Prerequisite Concepts, and its Related Concepts.

Step 8. According to the assumption of Eq. 1, the values of $h(x_i)$ are sorted ascendingly such that $h(x_1) \leq h(x_2) \leq ... \leq h(x_n)$.

Step 9. The value of the fuzzy measure is calculated for each of the three input parameters. This research uses the function introduced in [20] which calculates the fuzzy measure values as Eq. 8:

$$g_i = \frac{1}{1 + d(h_i, h_0)} \quad . \qquad i = 1,2,3 \tag{8}$$

g_i represents the fuzzy measure value of x_i. $d(h_i, h_0)$ is the Euclidean distance between $h(x_i)$ and h_0. h_0 is an optional value from which the distance of all $h(x_i)$ is calculated. In this research, it is assumed that $h_0 = 0$. Since the values of $h(x_i)$ are always positive, the range of g_i is between zero and one.

Step 10. λ is calculated by using Eq. 5. For the three input parameters, Eq. 5 is equivalent to solve Eq. 9:

$$(g_1 g_2 g_3) \times \lambda^2 + (g_1 g_2 + g_1 g_3 + g_2 g_3) \times \lambda + (g_1 + g_2 + g_3 - 1) = 0 \ . \tag{9}$$

Step 11. $g(A_1)$, $g(A_2)$, and $g(A_3)$ are calculated according to the value of λ using Eq. 6 and Eq. 7.

Step 12. To calculate the difficulty of the Main Concept, the values of $h(x_i)$ and g are used in Eq. 1. For three input parameter, Eq. 1 is equivalent to solve Eq. 10:

$$E_g(h) = h(x_1) \times g(A_1) + (h(x_2) - h(x_1)) \times g(A_2) + (h(x_3) - h(x_2)) \times g(A_3) \ . \tag{10}$$

Step 13. The value of $E_g(h)$ is the difficulty of the Main Concept for the learner.

4 Numerical Analysis

This section provides a numerical analysis of the proposed method. To this end, Java Programming Language is selected as the subject of the knowledge domain. Java Curriculum for AP ™ Computer Science is used for determining the knowledge domain ontology [21]. The modeling of this ontology is done by using Protégé software [22]. The ontology consists of 187 concepts which are categorized in 33 Sub-domains. The Concepts Hierarchy is designed according to the 33 Sub-domains. Fig. 2 shows part of the output of the knowledge domain modeling process using Protégé. This figure depicts the relationships among the concepts of the "Iterations" Sub-domain. The yellow and red arrow shows the Prerequisite and Related Concepts respectively. Each node in Fig. 2 represents the id of the concepts in the "Iterations" Sub-domain. The "Iterations" Sub-domain consists of 8 concepts that are shown is Table 1.

The following steps illustrate the proposed method for a specific learner:

Step 1. Learner selects the "Iterations" Sub-domain from the Concepts Hierarchy.
Step 2. Related Concepts of "Iterations" is selected from the knowledge domain ontology.

Table 1. Related Concepts of "Iterations" in the knowledge domain ontology

	Concept
1	While Loop
2	Loop Boundaries
3	Conditional Loop Strategies
4	For Loop
5	Nested Loops
6	Do-While Loop
7	Choosing a Loop Control Structure
8	Loop Invariants

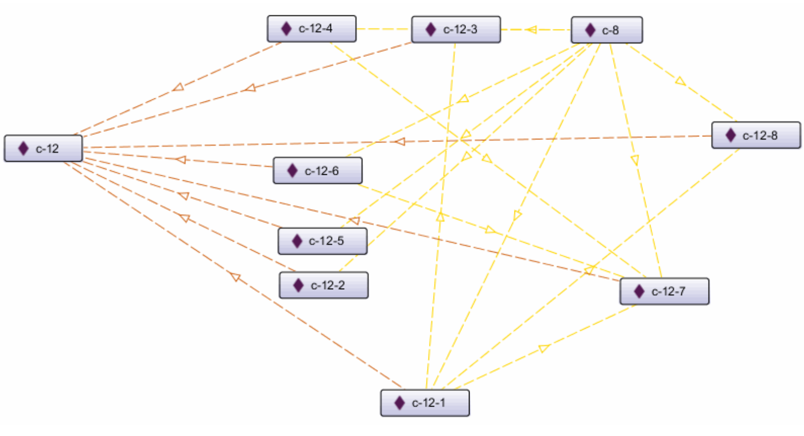

Fig. 2. Relationships between the concepts in the "Iterations" Sub-domain

Step 3. Since the learner has not learned any concept before, all the concepts obtained in Step 2 constitute the set of the Main Concepts. For each Main Concept, Steps 4 through Step 13 are followed. For example, for the "Loop Invariant" concept these steps are as follows:

Step 4. According to Table 2 which shows the values of the input parameters for the each of the Main Concepts, $x_1 = 2$.

Step 5. According to Table 2, $x_2 = 0$.

Step 6. According to Table 2, $x_3 = 0$.

Step 7. According to Step 4 through Step 6, $h(x_1) = 2, h(x_2) = 0, h(x_3) = 0$.

Step 8. According to the assumption of Eq. 1, the values of $h(x_i)$ are sorted ascendingly such that $(h'(x_1) = 0) \leq (h'(x_2) = 0) \leq (h'(x_3) = 2)$.

Step 9. The values of g_i for the "Loop Invariant" is calculated by Eq. 8:

$$g_1 = \frac{1}{1 + h'(1)} = 1.00$$

$$g_2 = \frac{1}{1 + h'(2)} = 1.00$$

$$g_3 = \frac{1}{1 + h'(3)} = 0.33$$

Step 10. The value of λ is calculated using Eq. 9 and is $\lambda = -0.99$.

Step 11. The value of $g(A_i)$ for the "Loop Invariant" concept is calculated by using Eq. 6 and Eq. 7 recursively. The final result is:

$$g(A_1) = g_1 + g_2 + g_3 + \lambda g_1 g_2 + \lambda g_1 g_3 + \lambda g_2 g_3 + \lambda^2 g_1 g_2 g_3 = 1.01$$
$$g(A_2) = g_2 + g_3 + \lambda g_2 g_3 = 1.00$$
$$g(A_3) = g_3 = 0.33$$

Step 12. The values of $h(x_i)$ and g are used in Eq. 10 to calculate the difficulty of the "Loop Invariant" concept for the learner:

$$E_g(h) = 0 \times 1.01 + (0 - 0) \times 1.00 + (2 - 0) \times 0.33 = 0.66$$

Step 13. The difficulty level of the "Loop Invariant" for the learner is 0.66.

Table 3 and Table 4 show the values of g and difficulty level for all of the Main Concepts in the "Iterations" Sub-domain respectively. The two values are the same for all the Main Concepts that has the same values for the input parameters shown in Table 2.

Table 2. The value of input parameters for the concepts in the "Iterations" Sub-domain

	Concept	x_1	x_2	x_3
1	While Loop	1	1	0
2	Loop Boundaries	1	0	0
3	Conditional Loop Strategies	2	0	0
4	For Loop	1	1	1
5	Nested Loops	1	0	0
6	Do-While Loop	1	1	1
7	Choosing a Loop Control Structure	4	1	0
8	Loop Invariants	2	0	0

Table 3. The values of fuzzy measure for the concepts in the "Iterations" Sub-domain

	Concept	g_1	g_2	g_3	$g(A_1)$	$g(A_2)$	$g(A_3)$
1	While Loop	1.00	0.50	0.50	1.01	0.75	0.50
2	Loop Boundaries	1.00	1.00	0.50	1.01	1.00	0.50
3	Conditional Loop Strategies	1.00	1.00	0.33	1.01	1.00	0.33
4	For Loop	0.50	0.50	0.50	0.88	0.75	0.50
5	Nested Loops	1.00	1.00	0.50	1.01	1.00	0.50
6	Do-While Loop	0.50	0.50	0.50	0.88	0.75	0.50
7	Choosing a Loop Control Structure	1.00	0.50	0.20	1.01	0.60	0.20
8	Loop Invariants	1.00	1.00	0.33	1.01	1.00	0.33

Table 4. Difficulty level for the concepts in the "Iterations" Sub-domain

	Concept	Difficulty Level
1	While Loop	0.75
2	Loop Boundaries	0.50
3	Conditional Loop Strategies	0.66
4	For Loop	0.88
5	Nested Loops	0.50
6	Do-While Loop	0.88
7	Choosing a Loop Control Structure	1.20
8	Loop Invariants	0.66

The highlighted concept in row 7 of Table 2 has the greatest value of x_1, x_2, and x_3 among all the other concepts. Hence, this concept is expected to be the most difficult concept in the "Iterations" Sub-domain for the learner. The highlighted row in Table 4 confirms this fact and shows the greatest difficulty level of this concept.

5 Conclusion

In this research a method is proposed to calculate the difficulty level for the concepts of the certain knowledge domain. To this end, the information in the learner model and Choquet Fuzzy Integral are used. This method provides the means to calculate the difficulty level of concepts which resides in the contents for learner. The content sequencing and presentation can be then adapted to the learner's knowledge and this in turn leads to the efficiency of the learning process. Choquet Fuzzy Integral considers the dependency among the input parameters, while simultaneously combines the information in the learner model with the relationship between the concepts of the knowledge domain. Therefore, the proposed method is more precise than the static methods which consider equal concept difficulty for all the learners with different levels of knowledge. The results of numerical analysis show that this method is quite hopeful in determining the difficulty level of the knowledge domain concepts. Compare to the work done in [13], the proposed method has less complexity by using Constraint-Independent models for both the knowledge domain and the learner model and hence, can be easily applied to different knowledge domains.

As it is shown in this research, the results obtained by this method can be used by an adaptive learning system for recommendation of the contents that are matched to the learner's knowledge. Our future works aims to implement the proposed system in this paper and evaluate the obtained results. Besides, a method will be proposed for providing adaptive content presentation based on dynamic calculation of the difficulty level of learning concepts. Moreover, the impact of adding or removing the influential parameters can be evaluated and the precision of the method can be improved accordingly.

Acknowledgement. This paper is provided as a part of a research performed in the Advanced E-learning Technologies Laboratory in Amirkabir University of Technology and financially supported by Iran Telecommunication Research Center (ITRC). Therefore, the authors would like to express their thanks to the ITRC for its significant supporting help.

References

1. Alexander, S.: E-learning Developments and Experiences. Education +Training 43, 240–248 (2001)
2. Brusilovsky, P.: Adaptive Hypermedia. User Modeling and User-Adapted Interaction 11, 87–110 (2001)
3. Zhu, F., Yao, N.: Ontology-Based Learning Activity Sequencing in Personalized Education System. In: 2009 International Conference on Information Technology and Computer Science, itcs, vol. 1, pp. 285–288 (2009)
4. Baylari, A., Montazer, G.A.: Design a Personalized E-learning System Based on Item Response Theory and Artificial Neural Network Approach. Expert Systems with Applications 36, 8013–8021 (2009)
5. Chen, C.-M., Liu, C.-Y., Chang, M.-H.: Personalized Curriculum Sequencing Utilizing Modified Item Response Theory for Web-Based Instruction. Expert Systems with Applications 30, 378–396 (2006)

6. Brusilovsky, P.: Adaptive and Intelligent Technologies for Web-Based Education. Künstliche Intelligenz 4, 19–25 (1999)
7. Melis, E., Andrès, E., Büdenbender, J., Frischauf, A., Goguadze, G., Libbrecht, P., Pollet, M., Ullrich, C.: ActiveMath: A Generic and Adaptive Web-Based Learning Environment. International Journal of Artificial Intelligence in Education 12, 385–407 (2001)
8. Dolog, P., Henze, N., Nejdl, W., Sintek, M.: The Personal Reader: Personalizing and Enriching Learning Resources Using Semantic Web Technologies. In: De Bra, P.M.E., Nejdl, W. (eds.) AH 2004. LNCS, vol. 3137, pp. 85–94. Springer, Heidelberg (2004)
9. Brady, A., O'Keeffe, I., Conlan, O., Wade, V.: Just-in-time Generation of Pedagogically Sound, Context Sensitive Personalized Learning Experiences. International Journal on E-learning 5, 113–127 (2006)
10. Chen, C.-M., Hsu, S.-H.: Personalized Intelligent Mobile Learning System for Supporting Effective English Learning. Educational Technology & Society 11, 153–180 (2008)
11. Chen, C.-M.: Intelligent Web-Based Learning System with Personalized Learning Path Guidance. Computers & Education 51, 787–814 (2008)
12. Weber, G., Brusilovsky, P.: ELM-ART: An Adaptive Versatile System for Web-Based Instruction. International Journal of Artificial Intelligence in Education 12, 351–384 (2001)
13. Mitrović, A., Martin, B.: Evaluating Adaptive Problem Selection. In: De Bra, P.M.E., Nejdl, W. (eds.) AH 2004. LNCS, vol. 3137, pp. 185–194. Springer, Heidelberg (2004)
14. Wei, T.: An Intelligent Tutoring System for Thai Writing Using Constraint Based Modeling. Master of Science Thesis, School of Computing, National University of Singapore (2005)
15. Harmelen, F., McGuiness, D.: OWL Web Ontology Language Overview, http://www.w3.org/TR/2004/REC-owl-features-20040210
16. Detyniecki, M.: Fundamentals on Aggregation Operators. Manuscript, Computer Science Division. University of California, Berkeley (2001)
17. Shieh, J.-I., Wu, H.-H., Liu, H.-C.: Applying a Complexity-Based Choquet Integral to Evaluate Students' Performance. Expert Systems with Applications 36, 5100–5106 (2009)
18. Narukawa, Y., Torra, V.: Fuzzy Measures and Integrals in Evaluation of Strategies. Information Sciences 177, 4686–4695 (2007)
19. Chiang, J.-H.: Aggregating Membership Values by a Choquet-Fuzzy-Integral Based Operator. Fuzzy Sets and Systems 114, 367–375 (2000)
20. Medasani, S., Kim, J., Krishnapuram, R.: An Overview of Membership Function Generation Techniques for Pattern Recognition. International Journal of Approximate Reasoning 19, 391–417 (1999)
21. Advanced Placement Program. Introduction of Java in (2003-2004) The College Board, http://www.collegeboard.org/ap/computer-science
22. Protégé. Developed by Stanford Medical Informatics at the Stanford University School of Medicine, http://protege.stanford.edu