Expert Systems with Applications 39 (2012) 7471–7478

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

A genetic fuzzy expert system for automatic question classification in a competitive learning environment

Elena Verdú, María J. Verdú *, Luisa M. Regueras, Juan P. de Castro, Ricardo García

School of Telecommunications Engineering, University of Valladolid, Paseo Belén, 15, 47011 Valladolid, Spain

ARTICLE INFO

Keywords:
Intelligent tutoring systems
Educational technology
Automatic question classification
Competitive learning
Genetic algorithms
Fuzzy systems

ABSTRACT

Intelligent tutoring systems are efficient tools to automatically adapt the learning process to the student's progress and needs. One of the possible adaptations is to apply an adaptive question sequencing system, which matches the difficulty of the questions to the student's knowledge level. In this context, it is important to correctly classify the questions to be presented to students according to their difficulty level. Many systems have been developed for estimating the difficulty of questions. However the variety in the application environments makes difficult to apply the existing solutions directly to other applications. Therefore, a specific solution has been designed in order to determine the difficulty level of open questions in an automatic and objective way. This solution can be applied to activities with special temporal and running features, as the contests developed through QUESTOURnament, which is a tool integrated into the e-learning platform Moodle. The proposed solution is a fuzzy expert system that uses a genetic algorithm in order to characterize each difficulty level. From the output of the algorithm, it defines the fuzzy rules that are used to classify the questions. Data registered from a competitive activity in a Telecommunications Engineering course have been used in order to validate the system against a group of experts. Results show that the system performs successfully. Therefore, it can be concluded that the system is able to do the questions classification labour in a competitive learning environment.

1. Introduction

During the last years, the learning process is changing substantially in order to be centred on the students and adapted to their needs and features. Different studies have shown the effectiveness of the new adaptive learning systems (Verdú, Regueras, Verdú, de Castro, & Pérez, 2008). Many of these systems attempt to be more adaptive by offering students questions with difficulty levels according to their skills and capabilities. The aim is to increase the efficiency and the level of interaction and motivation of students (Lilley, Barker, & Britton, 2004). Too difficult or too easy questions can frustrate and decrease students' motivation, while adaptive question sequencing provides a more efficient and effective learning (Wauters, Desmet, & Van den Noortgate, 2010). Moreover, according to (Lee & Heyworth, 2000), students should be able to score higher if the items or problems are arranged according to their difficulty level, since after solving easier problems, they feel more motivated to solve the harder ones.

On the other hand, the competitive learning systems, as the QUESTOURnament system, are an effective technique to capture students' interest, motivation and engagement by arousing their competitive instincts (Anderson, 2006; Philpot, Hall, Hubing & Flori, 2005). Moreover, competitive learning reduces procrastination, a common cause for students failing to complete assignments (Lawrence, 2004) and improves the learning process (Regueras et al., 2009).

QUESTOURnament is a telematic tool integrated into the e-learning platform Moodle that allows teachers to organize dynamic contests in any knowledge domain (Regueras et al., 2009). Students compete for getting the highest marks and being at the top in the ranking. They must solve exercises (known as *challenges* in QUESTOURnament) within a time limit and as soon as possible, since the scoring function varies with time.

The competitive nature of QUESTOURnament motivates students but also can provoke stress and discouragement in the worst classified students. To assign the adequate opponents and questions to a student may be an effective strategy to reduce these negative effects (Wu et al., 2007). Therefore the system should group students by knowledge level so that students with similar skills compete together and answer questions with a difficulty level suitable for them.

In this context, it is very important to correctly classify questions by difficulty level. However, it is difficult for teachers to accurately estimate the difficulty level according to the students' level of competence (Watering & Rijt, 2006). Experience helps teachers to better estimate the difficulty level of the questions, but even senior teachers sometimes fail and have to rectify when they

* Corresponding author. Address: ETSI Telecomunicación, Paseo Belén 15, 47011 Valladolid, Spain. Tel.: +34 983423707; fax: +34 983423667.

E-mail addresses: elever@tel.uva.es (E. Verdú), marver@tel.uva.es (M.J. Verdú), luireg@tel.uva.es (L.M. Regueras), jpdecastro@tel.uva.es (J.P. de Castro), ricgar@tel.uva.es (R. García).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2012.01.115

analyze the answers given by their students. An automatic estimation system could be the basis for an effective adaptation process.

A lot of systems that automatically estimate the difficulty level of items can be found in the literature (Burghof, 2001; Cheng, Shen, & Basu, 2008; Jong, Chan, Wu, & Lin, 2006; Lee, 1996; Wauters et al., 2010). However, the variety in the nature of the application environments makes difficult to apply the existing solutions directly to other applications. Therefore, a specific solution has been designed in order to turn the competitive e-learning system QUESTOURnament into an intelligent system. The objective is to make learning more effective and to mitigate some of the practical drawbacks of competitive learning.

This paper discusses the validity of an expert system that automatically estimates the difficulty level of the questions posed in the QUESTOURnament competitive learning system. Section 2 introduces the major issues about teachers' perception of difficulty and summarizes the search towards the solution. The expert system is described in Section 3. Section 4 starts with a description of the experiment developed in order to validate the system. Next, a study that analyzes the accuracy of the estimations of difficulty obtained by the intelligent system is presented. Finally, the main conclusions are stated.

2. Background

2.1. Teachers' perception of difficulty

The correct estimation of the difficulty level of learning material (questions, items...) is very important in the design and definition of assessment processes, adaptive learning systems or standard setting methods. However, there are not too many studies about the perception and estimation of difficulty level by teachers.

Estimating the difficulty level of questions is not an easy job. Several studies (Alexandrou-Leonidou & Philippou, 2005; Hadjidemetriou & Williams 2002; Lee & Heyworth, 2000; Watering & Rijt, 2006) question the ability of teachers to make accurate difficulty level estimations of learning material since teachers usually fail to identify the correct difficulty level according to the students' ability. In general terms, students' performance tends to be overestimated by teachers (Goodwin, 1999; Impara & Plake, 1998; Verhoeven, Verwijnen, Muijtjens, Scherpbier, & Van der Vleuten, 2002). Moreover, according to Watering and Rijt (2006), if the accuracy of teachers' perception of difficulty is analysed by categories, teachers tend to overestimate the difficulty of easy items and underestimate the difficulty of hard items. Impara and Plake (1998) also suggest that estimating item difficulty accurately is quite difficult; however, they do not think that teachers systematically underestimate the difficulty of hard items and overestimate the difficulty of easy items. In this respect, other contradictory results are found too. For example, Mattar (2000) states that teachers are less successful at rating very difficult or very easy items, while Zhou (2009) indicates that teachers classify better the hardest items.

In short, although there are not conclusive studies about the tendency of teachers when they classify questions by difficulty level, all researchers agree on the difficulty of doing this classification. Therefore, an automatic system that adjusts the difficulty level of questions according to the students' behaviour would be a very useful support tool and a key component for a truly adaptive learning environment.

2.2. In search of an intelligent solution for a competitive tool

There are many domain-dependent intelligent tutoring systems (ITSs) that provide students an adequate learning path through the different topics of a subject, according to the previously learnt topics. These systems are based on techniques such as Bayesian Networks (Hibou & Labat, 2004; Nouh, Karthikeyani, & Nadarajan, 2006; Vomlel, 2004) and require the previous definition of knowledge domains by using, for example, domain-specific ontologies (Colace & De Santo, 2006). Modelling these networks of knowledge components and their dependencies, generalizing them for every student, is not an easy task (Noguez, Sucar, & Ramos, 2005), especially for domain-independent systems like QUESTOURnament, which can be used for diverse subjects and levels of education.

Many domain-independent ITSs focus on presenting questions and problems adapted to the students' knowledge level. They often apply the Item Response Theory (IRT) to estimate both the characteristics of the questions, such as difficulty or guessing probability, and the knowledge level of students (Chen, Lee, & Chen, 2005; Lilley et al., 2004), independently of the knowledge domain. However, the correct application of traditional theories for tests implies some assumptions, which are not met by many examination contexts, especially when telematic tools are used for distance learning. Moreover, some of the characteristics of more specific tools, such as the competitive nature of QUESTOURnament, make the application of these theories difficult for the environment under study.

The typically used IRT models are one-dimensional, that is, they assume that the response to a question depends on a single trait, usually the knowledge level. Besides, it is also supposed that the response a student gives to a specific question does not depend on the responses given to other questions (Embretson & Reise, 2000). Therefore, using IRT entails carefully designing the tests so that these both conditions are fulfilled. Moreover, conventional IRT models only the response accuracy, ignoring response time; since it was thought to be used in pure power tests (Roskam, 1997), which assume that students have unlimited time to solve a question. Even if limited time could be assumed, at least the requirement should be that time is not a factor that affects the students' response. However, in a competitive environment as QUESTOURnament, time is very important, since only the first student who answers a challenge correctly will be able to obtain the highest score for that challenge. Therefore, there are different factors that could distort the results obtained by the IRT methods when applied to the QUESTOURnament system.

Students can apply different strategies during competition and even different personality factors can determine the students' final response to an item. Several challenges can be posed at the same time and students have to select one of them to be solved first. Many students tend to read all the different questions and select the one that seems the easiest to be solved first. Difficult challenges are usually read several times and solved after the easiest ones have been answered. On the other hand, two students with exactly the same knowledge level could respond to a same question differently, as one can be more persistent and devote more time to solve the question while another one can be more anxious with the competition and quickly respond to be the first one. Consequently, time and number of readings are important factors that should be taken into account in the model, but its modelling is dependent on the actual students' behaviour.

Moreover, when teachers pose challenges to QUESTOURnament, they do not have any restriction related to time, type of questions or skills needed to solve them. They are free to use any configuration of the system in any context. Then, there are some important factors that can vary:

- Maximum time available to submit an answer to a challenge.
- Type of questions (open questions, multiple choice questions, true/false questions, short response questions, problems, etc).
- Context surrounding students when they solve the questions: a contest may be developed in classroom or on distance during one or several days.

- Personality of the students (e.g. the stress of the student faced with a competitive situation can influence on the response).

There are different models adapted from the classic IRT that cover different partial aspects of the searched solution but there is not a model that covers all aspects required by the specific features of the QUESTOURnament system. Roskam (1997) presents a model based on IRT for speed tests with time limit where correctness and response time are integrated. Van der Linden (2007) proposes a flexible hierarchical solution that basically has an IRT model, a time-response distribution model and a higher level structure that has into account the dependencies between the items and the students' parameters in those models. For each of these components, the most suitable model can be used.

Anyway, a model based on IRT, which took into account all possible factors that influence the response a student gives to a challenge within QUESTOURnament, according to the so many different contexts of application, would be vastly complex. There are other solutions to determine the difficulty level of the learning material. However, most of these proposals are too simplistic – like the solution used in Jong et al. (2006), where the difficulty is estimated as the ratio between the number of times that a question is incorrectly answered and the total number of answers – or are too focused on the target subject – such as the solution described in Kunichika, Urushima, Hirashima, and Takeuchi (2002), which estimates the difficulty level of questions about English language sentences.

After analysing classical and specific solutions, it was decided to design an ad-hoc solution for the system, whose fundamentals could be applied to other systems used in open contexts. This solution is based on the definition of a fuzzy genetic expert system, which classifies the questions in several difficulty levels.

There are examples of successful application of this kind of systems to e-learning environments such as the one described by Romero, Gonzalez, Ventura, del Jesus, and Herrera (2009). They use an evolutionary algorithm to learn fuzzy rules, which describe relationships between the students' interactions with the e-learning system Moodle and the final marks obtained in the course. Typically, genetic learning of rules assumes a predefined set of fuzzy membership functions generated by human domain experts (Cordón, 2004). However, as aforementioned, the different nature of the challenges that can be posed through QUESTOURnament, as well as the varied students' profiles, makes it very difficult to define and generalize fuzzy sets and rules. Teachers can use QUESTOURnament for multiple-choice questions or for laborious exercises or problems. Since, for example, ten minutes can be a very short time for a complex problem but a long time for a true/false question, it is very difficult to predefine the fuzzy membership function for the time parameter. Moreover, the contests with QUESTOURnament can be developed in very different contexts, for example, during face-to-face classes or on distance, even lasting several weeks. All these elements (nature of the questions, application contexts of the system, profiles and behaviours of the students...) make it necessary to define fuzzy sets and fuzzy rules each time a group of questions are classified. Doing it by hand should be very laborious and impractical, so an automatic system is needed. Besides, according to Nebot, Mugica, Castro, and Acosta (2010), learning the *fuzzification* process parameters by genetic algorithms instead of using the expert's criteria provides better results.

Then, the proposed system starts from scratch. Taking some data about the interaction of the students with QUESTOURnament and the initial difficulty level estimated by the teacher, it learns both the adequate membership functions with their linguistic values as well as the fuzzy rules. In the next section, the complete system is detailed. Along this description of the genetic fuzzy expert system some real case examples are included to facilitate comprehension. The details of the real case and the corresponding experiment results are set out in Section 4.

3. The expert system

A genetic fuzzy expert system has been designed that generates fuzzy sets and rules appropriate for each specific case. The knowledge base is provided by a Fuzzy Model Generator that includes a genetic system capable of identifying the characteristics of the questions for each difficulty level.

The estimation of the difficulty level then takes place in two phases. During a first phase the Fuzzy Model Generator learns from the Facts Base (formed by the students' response patterns) and dynamically creates the classification rules and the fuzzy sets of the input variables for the specific data. During a second phase, the fuzzy expert system infers the difficulty level of each question.

The components of all the system are shown in Fig. 1. From Moodle and QUESTOURnament logs, three parameters are considered in the response patterns: time in minutes from the last reading of the question until the submission of the answer, grade obtained for that answer and number of accesses or readings before submitting the answer. All these factors depend on the students' behaviour when answering a question and are related to the difficulty level of each challenge (as aforementioned). All these data make up a set of context-dependent and noisy usage patterns that are stored in the Facts Base and feed the intelligent system.

For each difficulty level, the genetic system uses the response patterns of all the questions belonging to that level (according to the initial classification made by the teacher) and obtains a characterization of their responses as crisp sets. From these crisp sets the Fuzzy Model Generator creates the fuzzy sets and rules of the Knowledge Base. Once the fuzzy sets and the rules of a group of questions have been generated, the Inference Engine can infer the difficulty level of the patterns in the Facts Base. Finally, the difficulty level of each question is calculated as the median of the difficulty level of its response patterns and the challenges repository is updated with the new difficulty level.

Thus, the system combines the students' behaviour and the teachers' perception in order to objectively estimate the real difficulty level of each challenge.

3.1. The genetic system

The objective of the genetic algorithm for the proposed system is to generate groups of crisp sets that characterize the students' responses for three difficulty levels: easy, moderate and hard.

The system groups challenges by difficulty level according to the initial classification made by the teacher. The genetic algorithm then uses the responses for all the questions belonging to a specific difficulty level in order to obtain its characterization. As above mentioned, the input of the genetic algorithm is a set of response patterns with the structure <*time*, *grade*, *accesses*>. The crisp sets then are ranges of *time*, *grade* and *number of accesses* that together include the highest number of response patterns for a specific difficulty level. Therefore, a possible solution to the problem is represented with the following coded chromosome or individual [$t1$, $t2$, $g1$, $g2$, $a1$, $a2$], being $t1$ and $t2$ the lower and upper limits of a *time* range, $g1$ and $g2$, the lower and upper limits of a *grade* range, and $a1$ and $a2$, the lower and upper values of a *number of accesses* range.

The genetic algorithm implements the crossover operator BLX-α, the uniform mutation operator, the roulette wheel as selection method, and a fitness function based on the support measure typically used to evaluate inferred rules. In addition, due to the fact that the response patterns for a question do not depend only on the question itself but also on the behaviour of the students answering (e.g. knowledge level, persistence, etc.), the algorithm also incorporates niching methods, such as *sharing*, in order to promote the diversity and to be able to characterize each difficulty level by several groups

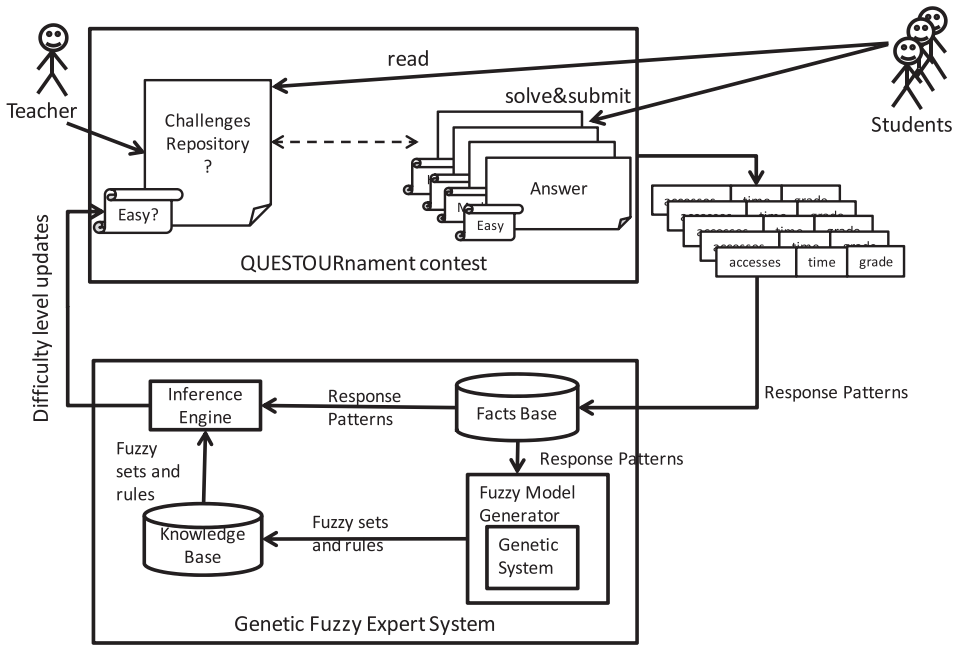

Fig. 1. Architecture of the genetic fuzzy expert system.

Table 1
Groups of ranges delivered by the genetic system (where Acc. = number of accesses).

Hard level				Moderate level				Easy level			
Time	Grade	Acc.	Fitness	Time	Grade	Acc.	Fitness	Time	Grade	Acc.	Fitness
[1,48]	[0,5]	[3,4]	0.143	[4,28]	[30,50]	[1,2]	0.301				
[13,36]	[44,60]	[1,2]	0.107	[7,35]	[0,18]	[1,2]	0.251	[0,25]	[80,100]	[1,2]	0.541
				[27,74]	[84,100]	[1,2]	0.167				

of ranges. More details about the fitness function, selection method, crossover and mutation operators and diversity methods are available in Verdú, Regueras, Verdú, and de Castro (2010a,2010b).

3.2. Generation of the Fuzzy Model

For each difficulty level, the genetic algorithm obtains groups of ranges of the input variables that characterize the response patterns of that difficulty level. Table 1 shows the ranges obtained by the genetic algorithm in the experiment (which is described in detail in Section 4.1). Later, from these groups of ranges, the Fuzzy Model Generator obtains the membership functions and the classification rules of the Fuzzy Model.

The fuzzy sets for the input variables *grade*, *time* and *number of accesses* corresponding to the data of Table 1 are represented in Fig. 2. In principle, a fuzzy set is defined for each range found by the genetic algorithm. For example the fuzzy set of the input variable *grade* with linguistic value "Very Low (VL)" corresponds to the grade range [0,5], found by the genetic system for the hard level questions. However, when two ranges are very similar, such as the grade range [80,100] found in easy questions and the grade range [84,100] found in moderate ones, the algorithm assigns only one fuzzy set for both ranges, "Very High (VH)" in this specific example. On the other hand, when the system finds a range that is not similar to another range but includes or overlaps this range, the system creates several fuzzy sets. For example, the grade range [0,18] includes the grade range [0,5]. In this case the system generates two different linguistic values, "Very Low (VL)" and "Low (L)".

The number of linguistic values of the input variables (Very Low, Low, Medium, High, etc.) is not fixed as it depends on the number of fuzzy sets determined by the Fuzzy Model Generator each time a group of questions are classified. In the given example, five fuzzy sets have been defined for the input variables *grade* and *time* and then, they take the linguistic values "Very Low (VL)", "Low (L)", "Medium (M)", "High (H)" and "Very High (VH)". However, only two fuzzy sets have been defined for the input variable *number of accesses* and then, two linguistic values are used: "Low (L)" and "High (H)".

The membership functions of the output variable *Difficulty* are not dynamically set, unlike those of the input variables, and always take the trapezoidal shapes shown in Fig. 2.

Once the fuzzy sets have been automatically created, the Fuzzy Model Generator defines the fuzzy rules from these fuzzy sets and the results of the Genetic Algorithm (see Table 1). For example, for the easy difficulty level the following fuzzy rules have been automatically defined:

IF GRADE IS VH AND TIME IS VL AND ACCESSES IS L THEN DIFFICULTY IS EASY
IF GRADE IS VH AND TIME IS L AND ACCESSES IS L THEN DIFFICULTY IS EASY

Two rules have been created from an only group of ranges found by the genetic system since the *time* range was split during the fuzzy sets creation phase. This same procedure is followed in order to define all the fuzzy rules from each group of ranges delivered by the genetic algorithm.

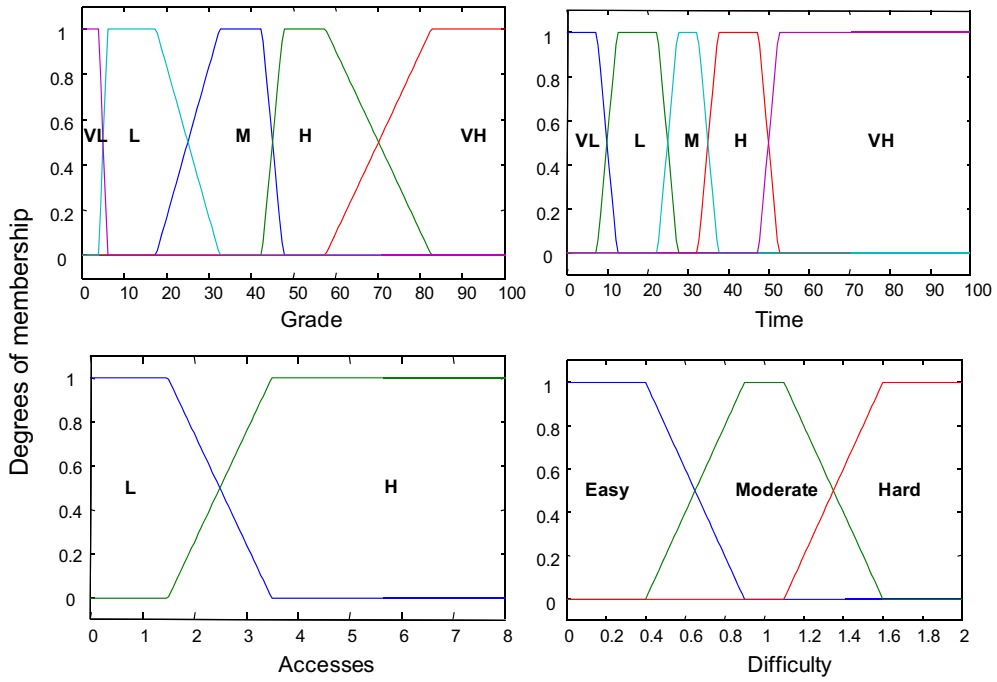

Fig. 2. Fuzzy sets of the input and the output variables.

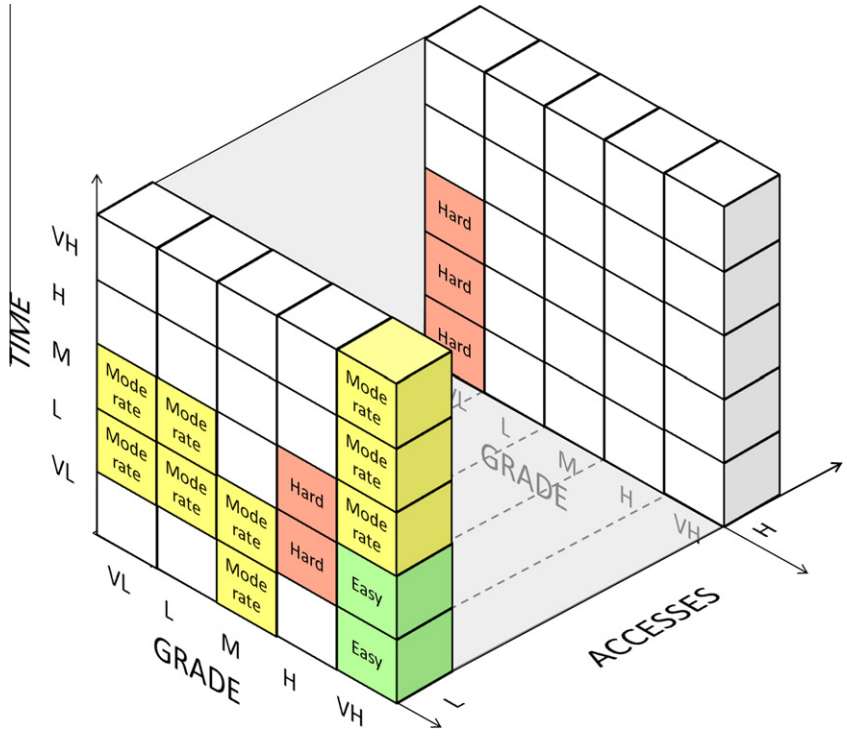

Fig. 3. Sliced-cube FAM representation of the set of rules.

The fuzzy rules can be graphically represented by using a cube form called *Fuzzy Associative Memory* (FAM) as shown in Fig. 3, where the 16 rules that describe the given data are represented.

At the end of this process, a set of fuzzy rules as well as the membership functions of the input and output variables describing the problem have been defined and stored in the Knowledge Base.

3.3. The inference engine

The inference engine uses the Mamdani method to infer the difficulty level corresponding to a response pattern from its three crisp input variables: *time*, *grade* and *number of accesses*.

The Matlab Fuzzy Logic Toolbox has been used to simulate the operation of this component. Again, a concrete example is used to show this operation. Fig. 4 shows the fuzzy inference of the

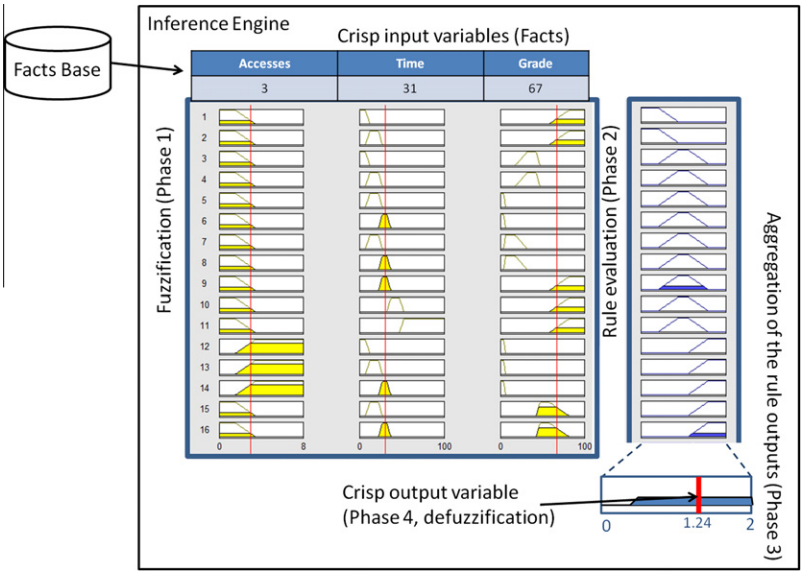

Fig. 4. Fuzzy inference of the difficulty of the input pattern <31,67,3>.

difficulty for the response pattern with *time*, *grade* and *number of accesses* equal to 31, 67 and 3, respectively.

Fuzzy inference takes place in four steps. First, the crisp input variables are *fuzzified* during the *fuzzification* phase. The rule evaluation phase then takes place. In the example, 16 rules describe the behaviour of the system but this input pattern only fulfils the three antecedent conditions of two rules. The classical Min method has been applied for the fuzzy operator AND. During the third step, the rule consequents are aggregated into a single fuzzy set using the Max composition method. Last, *defuzzification* obtains a crisp output using the MOM (Mean of Maximum) method. The pattern of the example has been assigned a difficulty value equal to 1.27, which corresponds to a difficulty level halfway between moderate and hard, but closer to moderate, as it can be seen in Fig. 2.

4. Results

The hypothesis to be tested in this paper is that the designed expert system performs as a human expert, that is, an expert teacher that is able to reclassify questions by difficulty by means of a thorough analysis of the students' behaviour while answering.

In order to analyze and validate the performance of this expert system, this one has been tested with real data from a contest developed with the QUESTOURnament tool in an undergraduate course of Diploma in Telecommunications Engineering at the University of Valladolid (Spain).

4.1. The experiment

The study was carried out from February until June 2010 (during three weeks with a 2-hour laboratory session in each week), with 38 enrolled students. All these students participated in the contest and 12 challenges (exercises on IP addressing and routing), were posed by the teacher. The system asked the teacher, upon creating the challenges, to classify them according to the estimated difficulty level: easy, moderate or hard.

According to the teacher's initial estimation; 4 challenges were classified as easy, 3 as moderate and 5 as hard. The total number of available answers for the easy, moderate and hard levels was 134, 103 and 169, respectively. For each answer, one response pattern is recorded with the grade obtained, the number of accesses and the time from reading to answering. All these records made up the input patterns to the system.

In the previous sections, it has been explained the design and operation of the genetic fuzzy expert system with the help of some examples related to the real case presented now. Therefore, Table 1 corresponds to the output given by the genetic algorithm with the input data corresponding to the 12 challenges posed in this course. These groups of ranges found by the genetic system were used by the Fuzzy Model Generator to create the membership functions of the input variables shown in Fig. 2 as explained in Section 3.2. In the same way, the output of the genetic system and the generated fuzzy sets were used to define the 16 classification rules shown in Fig. 3, which describe the behaviour of the students when answering these challenges of different difficulty levels. For example, the system found that answers with Very High grade and Low number of accesses correspond to the easy difficulty level if the time is Low or Very Low, or to the moderate difficulty level if the time is Medium or Higher.

On the other hand, as it was expected, easy questions are characterized by Very High grades and Very Low or low time. There are also moderate questions with answers graded Very High but the time in this case is higher than the time required for the easy questions before mentioned, indicating that time and difficulty are inversely related, which is consistent with the study presented in Mason, Zollman, Bramble, and O'Brien (1992).

Applying the rules to the students' response patterns, the expert system obtains the new difficulty level for each challenge. Thus, next step is to update the challenges repository consequently. According to the classification done by the system (see the third column in Table 2), three challenges should be reclassified (questions number 2, 8 and 10) as their initial difficulty does not match the difficulty assigned by the system.

Once the questions have been reclassified, it is necessary to validate the intelligent system.

4.2. Validation of the intelligent system

Since expert systems are addressed to perform at close to human expert levels and to solve problems without a defined correct solution, they should typically be validated against human experts (O'Keefe, Balci & Smith, 1987). Therefore, the chosen method for the system validation has been a "validation against a group of

Table 2
Data for validation of the expert system.

Id	Initial classification by teacher	Expert system (crisp value)	Expert system (linguistic value)	Human expert 1	Human expert 2	Human expert 3
1	Moderate	1.08	Moderate	Moderate	Moderate	Moderate
2	Moderate	1.51	Hard bordering on moderate	Hard bordering on moderate	Hard	Hard
3	Easy	0.38	Easy	Easy	Easy	Easy
4	Easy	0.52	Easy bordering on moderate	Easy	Easy	Easy bordering on moderate
5	Moderate	1.07	Moderate	Moderate	Moderate	Moderate
6	Hard	1.48	Hard bordering on moderate	Hard	Hard	Hard bordering on moderate
7	Easy	0.37	Easy	Easy	Easy	Easy
8	Easy	1.08	Moderate	Moderate	Moderate	Moderate
9	Hard	1.68	Hard	Hard	Hard	Hard
10	Hard	1.32	Moderate bordering on hard	Moderate	Moderate	Moderate bordering on hard
11	Hard	1.62	Hard	Hard	Hard	Hard
12	Hard	1.55	Hard bordering on moderate	Moderate bordering on hard	Hard	Moderate bordering on hard

experts" based on the method described in Mosqueira-Rey, Moret-Bonillo, and Fernández-Leal (2008). This method provides a measure of agreement between the human experts and verifies if the expert system performs as one of them and, therefore, it can be then incorporated into the group of experts without making the agreement level worse.

In the experiment described the group of experts consisted of the teacher of the course and other two teachers who are also experts on the subject. Table 2 shows the difficulty levels estimated by the genetic fuzzy expert system and the group of experts for each of the 12 challenges. First column is simply a challenge identifier. Second column shows the initial classification done by the teacher. Third column shows a crisp number that represents the difficulty of a challenge obtained by the system, which ranges from 0 to 2 (see Fig. 2). Fourth column represents the corresponding linguistic value for this crisp output. Last three columns show the classification done by the human experts, who carefully assigned a difficulty level to all the questions after analyzing the actual results and behaviour of students who answered them.

The level of agreement between each pair of human experts has been measured through the *weighted kappa* (Mosqueira-Rey et al., 2008). The results of the measure of *kappa* (see Table 3) show a strong agreement between pairs of experts; since values of *kappa* higher than 0.80 indicate an almost perfect agreement whereas values in the range 0.61–0.80 indicate a significant agreement (Viera & Garrett, 2005). The level of agreement between the expert system and each human expert varies from a significant agreement (*kappa* = 0.747) to an almost perfect agreement (*kappa* = 0.947).

Next, *Williams index* (Mosqueira-Rey et al., 2008) has been used as a measure to verify that introducing the system into the group of experts does not decrease the agreement level of the group. Values equal or higher than 1 indicate a good agreement whereas values lower than 1 imply that the agreement in the group of experts with the expert system included is worse than the agreement among only human experts. The *Williams index* has been calculated from the values for *kappa* of Table 3, and a value of 1.005 has been obtained. Therefore, it can be concluded that the system has performed successfully and is able to do the reclassification labour on behalf of the teacher satisfactorily.

Table 3
Values of *weighted kappa*.

	Human expert 1	Human expert 2	Human expert 3	Expert system
Human expert 1	–	0.901	0.805	0.837
Human expert 2	0.901	–	0.813	0.747
Human expert 3	0.805	0.813	–	0.947
Expert system	0.837	0.747	0.947	–

5. Discussion and conclusion

An expert system that satisfactorily classifies the challenges posed in the competitive learning system QUESTOURnament according to their real difficulty level has been designed. Teachers can insert challenges into the QUESTOURnament tool and the genetic fuzzy expert system will readjust the initial difficulty level estimated by the teacher according to the real behaviour of students when facing up to the challenges. The system has been tested with real data and the results have been successfully validated against human experts.

Once the system has been validated, its results can also be used to study and analyze the accuracy of estimations done by teachers. When compared with the difficulty level obtained by the expert system, the teacher's estimation (Initial classification by teacher in Table 2) is quite accurate, having into account that the teacher uses a three-scale method, whereas the values calculated from the data of the system include intermediate levels. The teacher's estimation does not match the difficulty assigned by the system in three cases. Specifically, the teacher overestimates the difficulty of one question and underestimates the difficulty of other two questions without following any rule. Thus, no conclusion can be obtained about the tendency of the teacher. The results of Table 2 only show that teachers estimate better the difficulty of hard questions than of easy or moderate questions.

For future work it is intended to study the possible inclusion of more parameters, related to students' profile and behaviour, in the response patterns. Besides, the system assumes that the initial classification of questions done by the teacher is good enough. It could be interesting to study in depth how dependant is the approach on the initial difficulty levels assigned by teachers.

Finally, the tests show that a same difficulty level is characterized by different fuzzy sets. This is probably due to the different behaviour of students when solving a challenge (for example, some of them can be more resolute while other ones can be more persistent) and, of course, to their different knowledge level. Then, the output of the system could also be used for students clustering because typical behaviours of students can be detected from the rules shown in the FAM representation of Fig. 3. For example, there are students who, when they do not know how to answer a challenge, read it several times and, finally, submit a quick response just in case they get it right by chance. This corresponds to the hard level questions characterized by a high number of accesses, Very Low to medium time for solving and Very Low grade. There are other rules for hard level questions that correspond to those more promising students who get a high grade even in hard questions. Therefore, each rule for the same difficulty level may correspond to students with similar knowledge level and/or behaviour when answering

questions; this result can be used to effectively classify students and to refine the model by taking into account their competences and profiles. Thus, it is also planned to study the possibility of using the different fuzzy sets obtained by the Fuzzy Model Generator to detect and classify groups of students according to their knowledge level and behaviour profile.

References

Alexandrou-Leonidou, V., & Philippou, G. N. (2005). Teachers' beliefs about students' development of the pre-algebraic concepto of equation. In *Proceedings of the 29th Conference of the International Group for the Psychology of Mathematics Education* (pp. 41–48). Melbourne: University of Melbourne.

Anderson, J. R. (2006). On cooperative and competitive learning in the Management Classroom. *Mountain Plains Journal of Business and Economics - Pedagogy, 7*, 1–10.

Burghof, K. L. (2001). Assembling an item-bank for computerised linear and adaptive testing in Geography. *International Education Journal, 2*(4), 74–83.

Chen, C.-M., Lee, H.-M., & Chen, Y.-H. (2005). Personalized e-learning system using item response theory. *Computers & Education, 44*(3), 237–255.

Cheng, I., Shen, R., & Basu, A. (2008). An algorithm for automatic difficulty level estimation of multimedia mathematical test items. In *Proceedings of the 8th IEEE International Conference on Advanced Learning Technologies* (pp. 175–179). Los Alamitos, CA: IEEE Computer Society.

Colace, F., & De Santo, M. A. (2006). A tutoring tool based on bayesian approach. In *Proceedings of Sixth International Conference on Advanced Learning Technologies* (pp. 109–113). Washington DC: IEEE Computer Society.

Cordón, O. (2004). Ten years of genetic fuzzy systems: Current framework and new trends. *Fuzzy Sets and Systems, 141*(1), 5–31.

Embretson, S. E., & Reise, S. P. (2000). *Item response theory for psychologists*. Mahwah, NJ: Lawrence Erlbaum Associates.

Goodwin, L. D. (1999). Relations between observed item difficulty levels and Angoff minimum passing levels for a group of borderline examinees. *Applied Measurement in Education, 12*, 13–28.

Hadjidemetriou, C., & Williams, J. S. (2002) Teachers' pedagogical content knowledge: graphs, from a cognitivist to a situated perspective. In *Proceedings of the 26th Conference of the International Group for the Psychology of Mathematics Education* (pp. 57–64). Norwich: University of East Anglia.

Hibou, M., & Labat, J.-M. (2004). Embedded Bayesian network student models. In *Proceedings of the Fifth International Conference on Information Technology Based Higher Education and Training* (pp 468–472). Istanbul: IEEE.

Impara, J. C., & Plake, B. S. (1998). Teachers' ability to estimate item difficulty: A test of the assumptions in the Angoff standard setting method. *Journal of Educational Measurement, 35*(1), 69–81.

Jong, B.-S., Chan, T.-Y., Wu, Y.-L., & Lin, T.-W. (2006). Applying the adaptive learning material producing strategy to group learning. *Lecture Notes in Computer Science, 3942*, 39–49.

Kunichika, H., Urushima, M., Hirashima, T., & Takeuchi, A. (2002). A computational method of complexity of questions on contents of English sentences and its evaluation. In *Proceedings of the International Conference on Computers in Education* (pp. 97–101). Auckland, New Zealand: IEEE Computer Society.

Lawrence, R. (2004). Teaching Data Structures Using Competitive Games. *IEEE Transactions on Education, 47*(4), 459–466.

Lee, F. L. (1996). Electronic Homework: An Intelligent Tutoring System in Mathematics. PhD Thesis. The Chinese University of Hong Kong.

Lee, F. L., & Heyworth, R. M. (2000). Problem complexity: A measure of problem difficulty in algebra by using computer. *Education Journal, 28*(1), 85–107.

Lilley, M., Barker, T., & Britton, C. (2004). The development and evaluation of a software prototype for computer-adaptive testing. *Computers & Education, 43*(1), 109–123.

Mason, E., Zollman, A., Bramble, W. J., & O'Brien, J. (1992). Response time and item difficulty in a computer-based high school mathematics course. *Focus on Learning Problems in Mathematics, 14*(3), 41–51.

Mattar, J. D. (2000). Investigation of the validity of the Angoff standard setting procedure for multiple-choice items. Ph.D. dissertation. University of Massachusetts.

Mosqueira-Rey, E., Moret-Bonillo, V., & Fernández-Leal, Á. (2008). An expert system to achieve fuzzy interpretations of validation data. *Expert Systems with Applications, 35*(4), 2089–2106.

Nebot, A., Mugica, F., Castro, F. & Acosta, J. (2010). Genetic fuzzy system for predictive and decision support modelling in e-learning. In *Proceedings of the 2010 IEEE International Conference on Fuzzy Systems* (pp. 1804–1811). IEEE Computer Society.

Noguez, J., Sucar, E., & Ramos, F. (2005). A probabilistic relational student model for virtual laboratories. In *Proceedings of the Sixth Mexican International Conference on Computer Science* (pp. 2–9). Puebla, Mexico. doi:10.1109/ENC.2005.7. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1592194>.

Nouh, Y., Karthikeyani, P., & Nadarajan, R. (2006). Intelligent tutoring system-Bayesian student model. In *Proceedings of the 1st International Conference on Digital Information Management* (pp 257–262). Bangalore: IEEE.

O'Keefe, R., Balci, O., & Smith, E. P. (1987). Validation of expert system performance. *IEEE Transactions on Expert Systems, 2*(4), 81–90.

Philpot, T. A., Hall, R. H., Hubing, N., & Flori, R. E. (2005). Using games to teach statics calculation procedures: Application and assessment. *Computer Applications in Engineering Education, 13*(3), 222–232.

Regueras, L. M., Verdú, E., Muñoz, M. F., Pérez, M. A., de Castro, J. P., & Verdú, M. J. (2009). Effects of competitive e-learning tools on higher education students: A case study. *IEEE Transactions on Education, 52*(2), 279–285.

Romero, C., Gonzalez, P., Ventura, S., del Jesus, M. J., & Herrera, F. (2009). Evolutionary algorithms for subgroup discovery in e-learning: A practical application using Moodle data. *Expert Systems with Applications, 36*(2), 1632–1644.

Roskam, E. E. (1997). Models for speed and time-limit tests. In W. J. van der Linden & R. K. Hambleton (Eds.), *Handbook of Modern Item Response Theory* (pp. 188–207). New York: Springer-Verlag.

Van der Linden, W. J. (2007). A hierarchical framework for modeling speed and accuracy on test items. *Psychometrika, 72*(3), 287–308.

Verhoeven, B. H., Verwijnen, G. M., Muijtjens, A. M. M., Scherpbier, A. J. J. A., & Van der Vleuten, C. P. M. (2002). Panel expertise for an Angoff standard setting procedure in progress testing: Item writers compared to recently graduated students. *Medical Education, 36*, 860–867.

Verdú, E., Regueras, L. M., Verdú, M. J., de Castro, J. P., & Pérez, M. A. (2008). An analysis of the research on adaptive learning: The next generation of e-learning. *WSEAS Transactions on Information Science and Applications, 5*(6), 859–868.

Verdú, E., Regueras, L. M., Verdú, M. J., & de Castro, J. P. (2010a). Estimating the difficulty level of the challenges proposed in a competitive e-learning environment. *Lecture Notes in Artificial Intelligence, 6096*, 225–234.

Verdú, E., Verdú, M. J., Regueras, L. M., & de Castro, J. P. (2010b). A diversity-enhanced genetic algorithm to characterize the questions of a competitive e-learning system. In *Proceedings of IEEE International Conference on Advanced Learning Technologies* (pp. 25–29). Los Alamitos, CA: IEEE Computer Society.

Viera, A. J., & Garrett, J. M. (2005). Understanding interobserver agreement: The Kappa statistic. *Family Medicine, 37*(5), 360–363.

Vomlel, J. (2004). Building adaptive tests using Bayesian Networks. *Kybernetika, 40*, 333–348.

Watering, G. V. D., & Rijt, J. V. D. (2006). Teachers' and students' perceptions of assessments: A review and a study into the ability and accuracy of estimating the difficulty levels of assessment items. *Educational Research Review, 1*, 133–147.

Wauters, K., Desmet, P., & Van den Noortgate, W. (2010). Adaptive item-based learning environments based on the item response theory: Possibilities and challenges. *Journal of Computer Assisted Learning, 26*, 549–562.

Wu, W. M. C., Cheng, H. N. H., Chiang, M.-C., Deng, Y.-C., Chou, C.-Y., Tsai, C.-C., & Chan, T.-W. (2007). Answer matching: A competitive learning game with uneven chance tactic. In *Proceedings of the First IEEE International Workshop on Digital Game and Intelligent Toy Enhanced Learning* (pp.89–96). Los Alamitos, CA: IEEE Computer Society.

Zhou, W. (2009). Teachers' estimation of item difficulty: What contributes to their accuracy? In *Proceedings of the 31st Annual Meeting of the North American Chapter of the International Group for the Psychology of Mathematics Education* (pp. 261–264). Atlanta: Georgia State University.