Applied Soft Computing 11 (2011) 2142–2150

Contents lists available at ScienceDirect

Applied Soft Computing

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

Fuzzy and MultiAgent Instructional Planner for an Intelligent Tutorial System

R. Aguilar[a], V. Muñoz[a,*], E.J. González[a], M. Noda[b], A. Bruno[b], L. Moreno[a]

[a] Dpto. Ingeniería de Sistemas y Automática y Arquitectura y Tecnología de Computadores, Universidad de La Laguna, Spain
[b] Dpto. Análisis Matemático, Universidad de La Laguna, Spain

ARTICLE INFO

Article history:
Received 31 March 2008
Received in revised form 20 July 2010
Accepted 25 July 2010
Available online 3 August 2010

Keywords:
Intelligent Tutoring System (ITS)
Expert system
Fuzzy system
MultiAgent System
Decision-support systems

ABSTRACT

This article presents some aspects in our research into the design of a Fuzzy and MultiAgent Instructional Planner belonging to an Intelligent Tutoring System (ITS), which has been designed as a tool for the reinforcement of the addition operation. The authors propose the combined use of both fuzzy and Multi-Agent Systems. The fuzzy logic methodology is used to model the student's knowledge and the teaching strategy. Furthermore, the MultiAgent System implemented determines the learning objectives so as to provide the student with an efficient learning process. The fuzzy and MultiAgent Systems comprising the instructional planner were verified with the collaboration of experts in mathematics and in other areas of knowledge. The results obtained by the primary school children who used the ITS are also presented.

1. Introduction

The topic of this paper is the design and implementation of an instructional planner for an Intelligent Tutorial System (ITS) intended to reinforce the logical concepts of numbers, addition and subtraction as part of a primary school curriculum, though it can easily be generalized to other application domains.

Intelligent Tutoring Systems [1–3] were first used in the 70s as a way to provide greater flexibility to the learning strategy and to achieve better interaction with the user [4]. The aim of ITS is to capture the knowledge of experts to create dynamic interactions with the users, allowing them to make decisions, even those that may not have been anticipated by the experts.

The main advantage of ITS versus traditional tutoring systems is that they are more flexible in both their approach to the learning domain as well as in their adaptation to the student. In traditional systems, which contain a large amount of rules, and therefore of information, the student can find herself lost and improperly guided by the tutorial. It can also happen that since traditional tutorials do not adapt to the student, if she already has knowledge of the subject she will become bored if forced to inflexibly adhere to a sequence of activities whose concepts are already familiar to her. The opposite could also happen whereby the tutorial advances too quickly for a student who has not fully assimilated the basic concepts. The solution offered by ITS is the inclusion of several modules containing information about the student and the domain. In this way the tutorial, thanks to its flexibility and adaptability, can offer solutions to a wide range of users and benefit from the use of audiovisual resources (video, audio, animation, etc.) to motivate the student.

Keeping in mind that statement, In this paper we propose a sophisticated planning process at different levels of the ITS processing by decomposing the tutor module into two different components through the combined use of fuzzy logic and multiagent techniques, taking advantage of the benefits provided by each. Fuzzy logic is employed due to the usefulness of having a tool in the ITS for handling imprecision and that allows for knowledge to flow freely from the expert. There are various ways of modeling imprecision, but in most cases that imprecision is used to a small extent. The majority of methods for handling imprecision are probabilistic, though it is interesting to bear in mind that experts do not normally think in terms of probability, but rather in terms such as much, little, good, average, etc. An expert fuzzy system means that the system incorporates fuzzy sets and/or fuzzy logic into the reasoning process and/or into the representation of knowledge. The theories on fuzzy sets and logic are well established, having existed now for 25 years and applied in multiple control applications. Expert systems that feature fuzzy techniques have also been developed [4,5]. In the ITS, the fuzzy logic methodology is employed to model the uncertainty in the student's knowledge base and in the teaching strategy [4,6]. The use of a MultiAgent System, on the other hand, offers a great advantage in applications that are able to use distributed computing since the ability to divide tasks provides modularity and flexibility while reducing computing time [7,8]. Incorporating this technique into the ITS improves the objective learning strategy

* Corresponding author.
E-mail address: vanessa@isaatc.ull.es (V. Muñoz).

1568-4946/$ – see front matter
doi:10.1016/j.asoc.2010.07.013

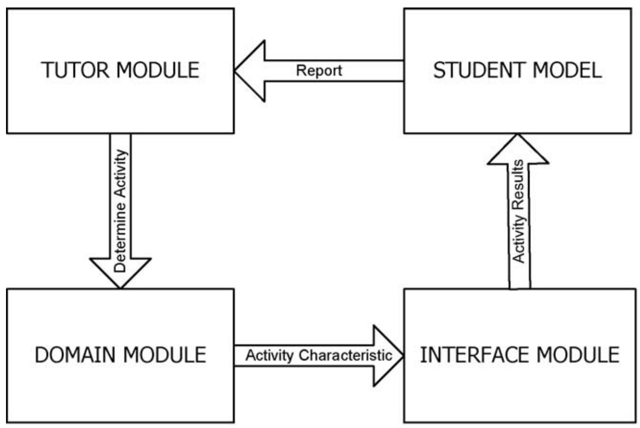

Fig. 1. Architecture of an ITS.

while at the same time decreasing the wait time in interacting with the user [9].

This paper is organized as follows. In Section 2 we describe the ITS architecture and domain. In Section 3 different methods for the implementation of instructional planners are enumerated. In Sections 4 and 5 we describe the two fundamental parts of the instructional planner: the fuzzy system and the MultiAgent System. Section 6 shows the results of the ITS validation. Lastly, the conclusions are presented in Section 7.

2. Architecture and domain of Intelligent Tutoring System

The tutorial will follow an individualized teaching process which consists of determining what the learning objectives are based on the students' predetermined characteristics. To this end, a series of activities will be devised for the students to perform so as to enable them to acquire the specified skills. The set of activities, therefore, will not be the same for all the students, but rather will depend on the characteristics of each.

The ITS uses a modular structure that lends itself to being reused in other application domains. Although mathematics was used as the application domain in this implementation of the ITS, it has been designed so as to enable its generalization to any other domain. Two key aspects make this generalization possible: the database-centered approach, which uses a database (DB) to store all of the system's static and dynamic information; and the design of the fuzzy and MultiAgent Systems used in the implementation of the Instructional Planner.

2.1. ITS architecture

It is possible to consider Intelligent Tutorial Systems as being comprised of following four modules (domain module, student model, tutorial module and interface module) [10] (Fig. 1):

- Domain module: This module contains the knowledge about the area of study. It contains the specific and detailed knowledge of the application domain as obtained from human experts. Every concept related to the learning objectives is described. In addition, the mechanisms for learning said concepts are specified. Therefore, for each student several tasks are specified. These tasks are chosen based on the learning objectives for this student. The student characteristics are obtained from the student's profile. Then, the tasks (for example motivation, presentation, evaluation or reinforcement) will be displayed to the student through the interface module.
- Student model: This module monitors the student's progress. It contains all the data and information on the student. These data are used to choose both the appropriate subsequent subject and the educational methodology or strategies. Basically it is possible to divide the model of the student into two fundamental components, the profile and the history. The profile indicates the personal characteristics of the student as related to their motor function, cognitive and psycho-social development. Variables that are considered in the profile are: chronological age, cognitive age, physical capacities (fine motor function, ocular contact, auditory capacity), cognitive development (short-term memory, attention, capacity to form concepts, capacity to group objects into significant categories, language and vocabulary development, understanding of words, capacity to react and initiative), interaction with the surroundings (connection with the surroundings, level of adaptation, participation in group activities, obtaining of information from the surroundings), personality (fear of failure, dynamics, hyperactivity or passivity), behavioral patterns (repetitive conduct, restricted conduct, stereotyped activities), preferences (colors, personal interest). The history covers those variables related to previous concepts that the student knows, as well as the progress exhibited (repeated exercises, omitted tasks, rate of progress between activities, learning style, etc.).
- Tutorial module or instructional planning: It controls the system. This module determines the education strategies so the system can adapt and improve the tutorial strategies based on the student. It must detect the level of the student, select the next activity to be carried out by the student, select examples, correct errors, etc.
- Interface module: Its objective is to display the subjects to the students. The interface can, depending on its design, make the user's interaction with the system more or less understandable. This can affect the acceptance level that the student has of the ITS.

2.2. Application of the proposed architecture

The educational objective of the ITS presented in this paper is the number concept and the objectives of sum and subtraction in Spain's primary education curriculum. Some objectives are presented simultaneously, while others are a prerequisite for their successors (Fig. 2). The result is four phases that the student has to cover successively. It is possible to advance or to backtrack based on the results of the execution of the activities by the student. Each of these phases will consist of objectives to be covered in parallel. These objectives are carried out through related activities. The activities of a single objective can be divided into several difficulty levels. When the student carries out the activities corresponding to each objective with a suitable passing grade, the student can progress to the following phase.

The phase-1 activities to be completed by the student are grouped into two difficulty levels, while those for the objectives corresponding to phases 2, 3 and 4 are grouped into three difficulty levels.

3. Tutoring module

The tutorial module or Instructional Planner makes the decisions in the ITS. The process of individualized education consists of determining the learning objectives considering the characteristics of each student. A set of tasks to be carried out by the student is devised so as to allow him to acquire the concepts. There will not be a standard set of tasks to be carried by every student, as these will depend on the characteristics of each. For each concrete stu-

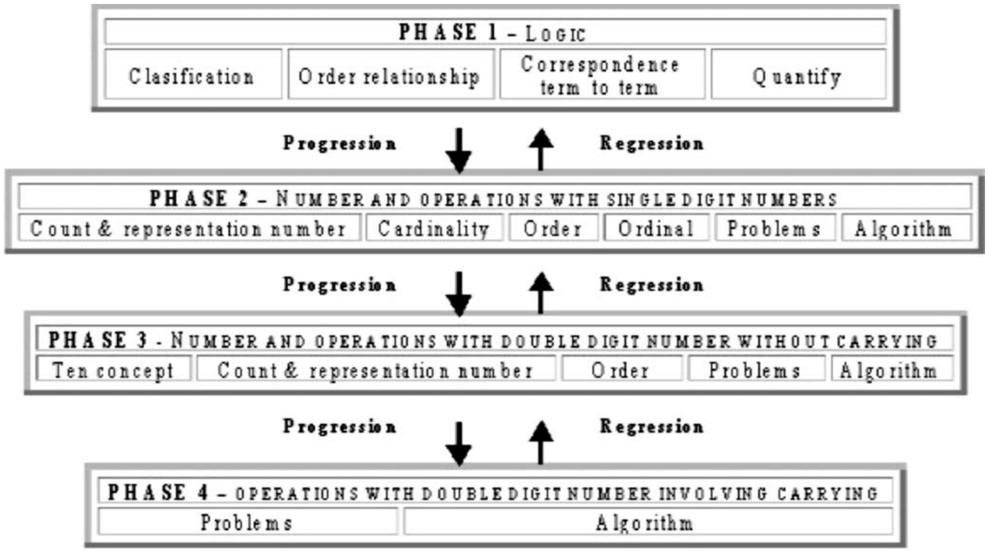

Fig. 2. Phases in the addition operation.

dent, learning objectives are established and a sequence of actions is specified for attaining those objectives. A search is conducted of the space state for the tasks to be carried out by the student in achieving the objective. Therefore, the Problem-Solving Method used by the instructors consists of a planning process, in which the nodes of the search tree represent student situations (extent to which objectives are achieved), and the path from one node to another is the plan for progressing from one situation to another (where a plan is the actions to be carried out by the student). The objective of this plan is to design the exercises to be completed by the student. The student then moves along the situation tree in the general direction of the objective node. Elements to consider in the design of the student exercises (plans) are as follows:

- Student characteristics.
- Learning objectives.
- Resources available.

Searching the state space backwards (from the objective to the initial state) is more efficient because, in general, the objectives have few ramifications. Nevertheless, in an ITS it is not possible to use this method because the situation of the student cannot start off from the final state. It is for that reason that the authors propose the combined use of fuzzy and MultiAgent Systems for the implementation of the instructional planner. Various authors have proposed implementations of the tutoring module based on fuzzy systems or multiagents systems like Nkambou and Kabanza that proposed multiagent planning approach at different levels of the ITS processing [7], Ganjanasuwan and Sanrach that proposed an integrated multi-agent for automation of courseware production planning [8], Hospers et al. (an agent-based Intelligent Tutoring System for nurse education [11]) and Choi and Yang (a Fuzzy Set Based Tutoring System for Adaptive Learning [12]).

Furthermore various authors have proposed other different implementations of the tutoring module: [3,13,14]

- Case-based reasoning: the process of solving new problems based on the solutions to similar problems in the past. One problem with case-based reasoning is that a set of previous cases is necessary in order to initiate the reasoning process. A drawback of this technique is that the most appropriate case may not be selected when choosing a solution from among the stored cases.
- Markov decision processes: Markov decision processes provide a mathematical framework for modeling decision making, planning, and control in situations where outcomes are partly random and partly under the control of the decision maker. Disadvantages of this method are the difficult modeling of complex systems and that the systems do not have memory.
- Neuro-fuzzy is a hybrid intelligent system that synergizes the human-like reasoning style of fuzzy systems with the learning and connectionist structure of neural networks. A disadvantage in a Fuzzy Neural Network is that many parameters must be defined, thus complicating their interpretation.
- A Bayesian network (or a belief network) is a probabilistic graphical model that represents a set of variables and their probabilistic independencies. Bayesian networks present two basic disadvantages: the inability to correlate the variables, considering their behavior in time, and the difficulty of establishing the optimum combination of states for the variables.
- Blackboard system: a type of Artificial Intelligence application based on the blackboard architectural model. A blackboard system enables a flexible brainstorming style of interaction between diverse specialists (can be agents). They use the blackboard as the workplace for cooperatively developing the solution. In the blackboard coordination scheme, a severe bottleneck may result if there are many agents.
- Rule-based system: represents knowledge in terms of a set of rules that specify what to do or what to conclude in different situations. Rule-based systems have many disadvantages: difficulty in deriving rules, no degree of truth, difficulty in designing a program applicable to teaching that considers every imaginable rule, hard to maintain a complex rule-based system.

In this article the authors propose the combined use of fuzzy logic and MultiAgent Systems since they allow for the knowledge of the human experts to be acquired immediately. In addition, that facilitates its generalization to other domains of application.

The Instructional Planner proceeds as shown in Fig. 3:

1. Once the student solves an objective's activities, the manager activates the fuzzy system and sends it the student's information.
2. The fuzzy system decides the next level to be presented to the student for that objective.
3. Once the level is determined, the manager activates the MultiAgent System.

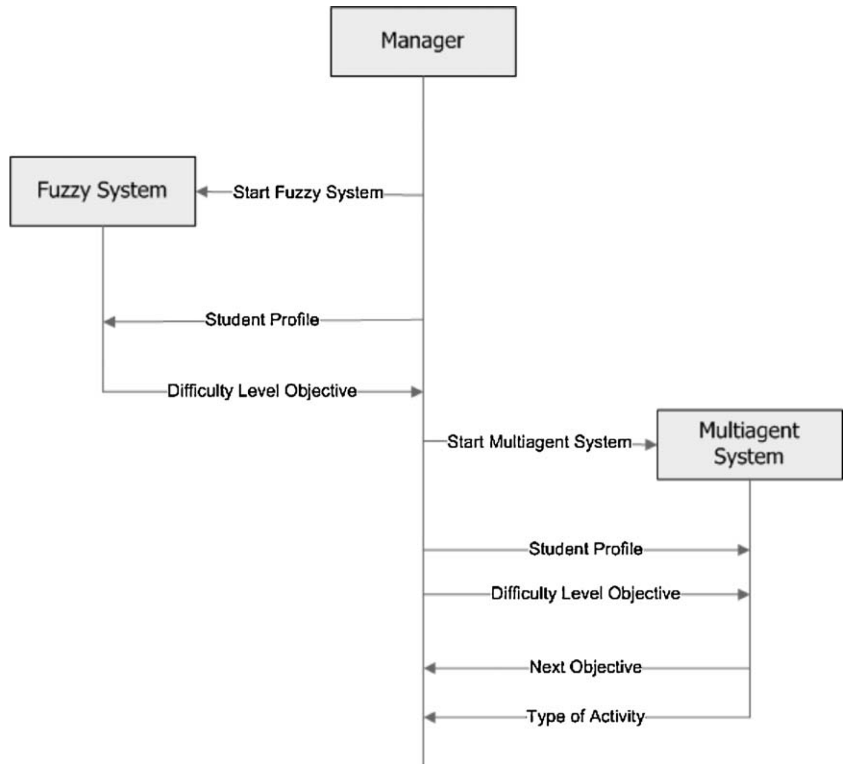

Fig. 3. Sequence of the Fuzzy and MultiAgent Instructional Planner.

4. Said system is tasked with deciding the next objective and activity type within that objective to be presented to the student, based on the student's data and record.

Next, in Sections 4 and 5, we describe the design and development of each of the systems that comprise the Instructional Planner: the fuzzy system and the MultiAgent System.

4. Instructional Planner: fuzzy system

The use of fuzzy logic techniques [15,16] in systems like the instructional planner is immediately applicable given the planner's behavior based on imprecisely defined rules. This imprecision stems from the system's own complexity. This type of problem is addressed by reducing the complexity and increasing the uncertainty of the variables. The planner behaves according to a set of rules that are often imprecise, or that use linguistic terms laden with uncertainty. This results in rules of the type "If the student is progressing well, then increase the activity's complexity level". Said rules are provided by the *pedagogical expert*, the person who knows how best to teach each student based on their particular characteristics as determined from experience gained in the classroom.

The Fuzzy Instructional Planner consists of a set of fuzzy systems that will infer the sequence of activities to be performed by the student. This sequence is based on the results obtained by the student. Since the teaching of the number, addition and subtraction concepts entails covering a series of objectives, as many fuzzy systems must be built as there are learning objectives.

The two inputs to the fuzzy system are: percentage of right answers provided by a student (ASR), and a variable to indicate the student's progress and record (ST) (Fig. 4). ASR is a specific input whose value is between 0 and 1 (for example, if the student correctly answers 75% of the activities, this input will have a value of 0.75). ST is a fuzzy input corresponding to the output of the fuzzy system prior to defuzzification (system feedback) [17]. Based on the rules implemented, the system is capable of determining the complexity level of the activities to be shown to the student (NCAO). This value, once defuzzified, is the output of our system.

As mentioned previously, the learning objectives are grouped into phases. The transition between phases (progression, regression and prolongation) is handled by a fuzzy system like that shown in Fig. 4. The inputs to this fuzzy system are the unfuzzified outputs of each of the objectives in the phase being worked on by the student (that is, the student's progress in each objective). The defuzzified output yields an exact number that indicates whether the student remains in the same phase, advances to the next or goes back to the previous one.

A convenient way of representing the set of rules is by using a table, where each square represents the linguistic value of the consequent of a rule, and the column to the left and the top row contain the linguistic values of the variables of the antecedent.

One example of the instructional planner rules for students who are afraid of failure, or who are hyperactive or motivated is shown in Tables 1–3. These are students who have difficulty focusing when faced with a stimulus, and who have to be given a greater variety of activities to keep them interested.

As examples we have:

(1) If 'ASR' is Very Low and 'ST' is Go back, then 'NCAO' is Go back.

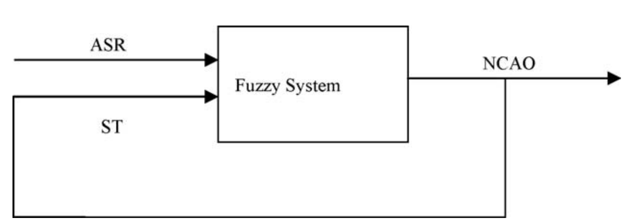

Fig. 4. Fuzzy system for teaching objective N.

Table 1
Rule base for a Fuzzy Instructional Planner for a student with a fear of failure.

ASR\ST	Very Low	Low	Average	High	Very High
Go back	Go back	Go back	Go back	Stay	Stay
Stay	Go back	Go back	Stay	Stay	Advance
Advance	Go back	Stay	Stay	Advance	Advance

Table 2
Rule base for a Fuzzy Instructional Planner for a motivated student.

ASR\ST	Very Low	Low	Average	High	Very High
Go back	Go back	Go back	Stay	Stay	Stay
Stay	Go back	Stay	Stay	Stay	Advance
Advance	Stay	Stay	Stay	Advance	Advance

Table 3
Rule base for a Fuzzy Instructional Planner for a hyperactive student.

ASR\ST	Very Low	Low	Average	High	Very High
Go back	Go back	Go back	Stay	Stay	Advance
Stay	Go back	Stay	Stay	Advance	Advance
Advance	Stay	Stay	Advance	Advance	Advance

This rule quantifies the situation in which the student does poorly with the activities presented, as well as in previous interactions. The strategy in this case is to lower the complexity of the activities proposed. This rule, for example, is the same for the three student types.

(2) If 'ASR' is High and 'ST' is Stay, then 'NCAO' is Stay for students who are afraid of failure or motivated and Advance for hyperactive students.

This rule quantifies the situation in which the student correctly answers the activities proposed, but given her previous average performance, is kept at the same difficulty level. If the student is hyperactive, however, the complexity of the new levels proposed should be on a higher level so as to vary the types of activities and to ensure the student is not distracted or loses interest in the tutorial.

The fuzzy partitions for the input variable ASR for each fuzzy system that handles the learning objectives, then, will consist of 5 Gaussian diffuse sets distributed in a normalized universe of discourse in the range [0,1], Fig. 5. In the case of the ST input variable and of the NCAO output variable, the fuzzy partition will be the same since the ST variable is the feedback from the fuzzy system output. The fuzzy partitions will consist of 3 Gaussian diffuse sets distributed in a normalized universe of discourse in the range [0,1] (Fig. 6).

5. Instructional Planner: MultiAgent System

Another important element in planning is determining the set of activities to be presented to the student once the difficulty level of said activities has been decided. To determine which objective and which activity within that objective to show next, we opted to use a MultiAgent System (MAS).

An initial approach for the instructional planner was developed by randomly determining the objective and type of activity shown to the student once the difficulty level had been obtained using fuzzy logic. The experiments conducted using this prototype showed that some objectives were never presented to the student, while others were repeated often. That is why it was decided to use a MAS to determine the objective to be presented and the type of activity within that objective to be carried out by the student.

A MultiAgent System (MAS) is a system in which coexist a set of agents that interact with each other. Each agent has its own objectives and must cooperate with or compete against other agents in order to carry out these objectives [3]. The decision to use a Multi-

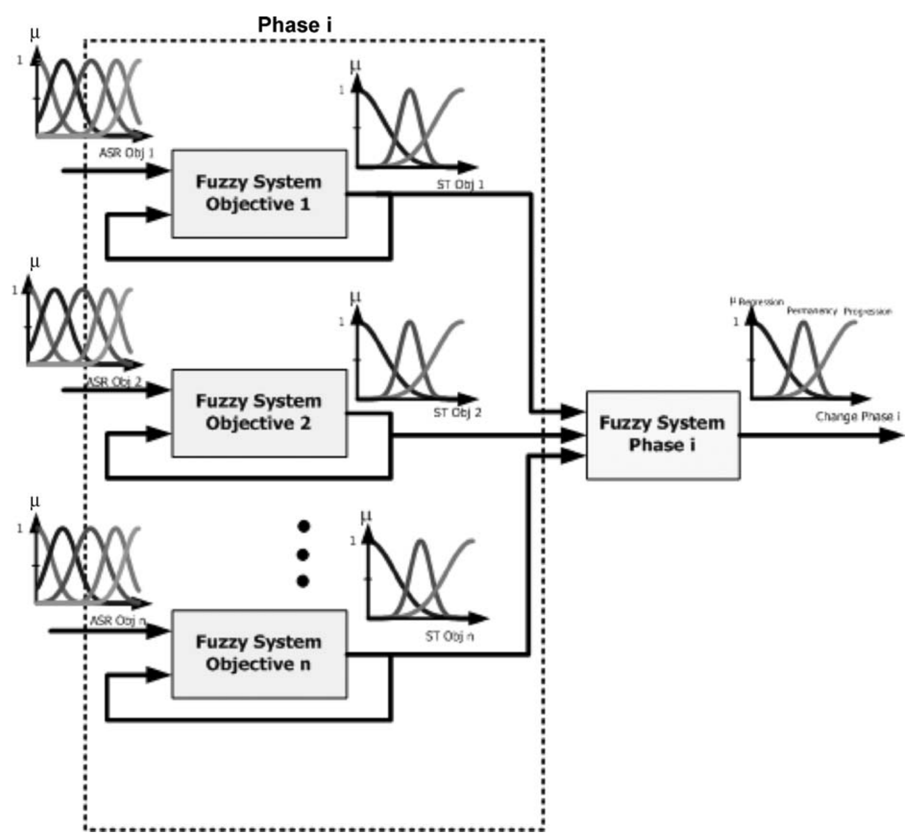

Fig. 5. Fuzzy system used to infer the student's current phase (set of working objectives).

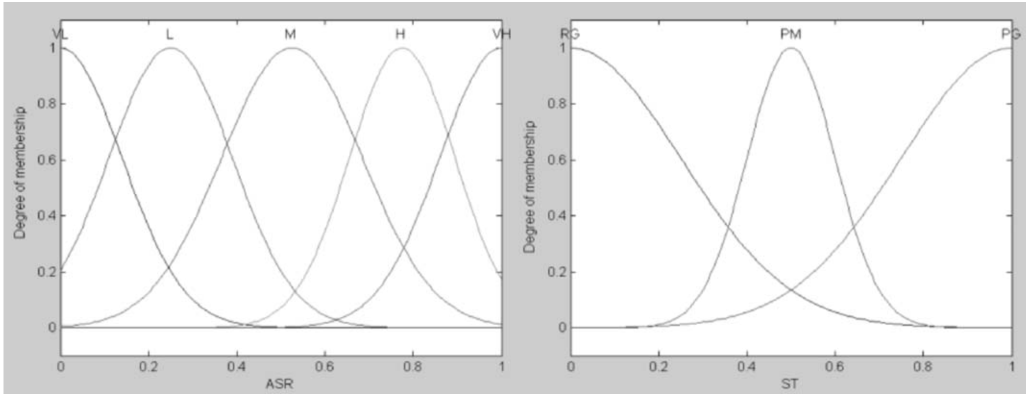

Fig. 6. Fuzzy partitions of input variable ASR and ST.

Agent System is important to the creation of an intelligent teaching platform because of its great ability to react [18], thanks to the distribution of tasks to agents working concurrently. This means that while the student does the final exercises, the agents can be deciding on the education strategy to be followed in the next objective. This provides for a shorter delay in the user's interaction with the ITS.

Since in a MultiAgent System no agent has direct control over other agents, a method for coordinating agents and resolving conflicts is necessary. This implies the introduction of a negotiation concept, which can be defined as a process by means of which a common decision is made by two or more parties. The parties involved first enunciate their contradictory demands and eventually move towards an agreement by means of a process of concessions or by searching for new alternatives. Different negotiation mechanisms exist. The instructional planner presented in this paper uses an auction-based mechanism, which offers a greater speed in attaining a solution. We rely on a first-price auction (each bidder submits a bid in a sealed envelope, with the highest bidder receiving the good and paying the amount bid) using intelligent agents who modify the bid value adaptively.

Determining, then, the next learning objective to be worked on by the student is the task of the Objective Supervisor Agent, which is in charge of negotiating with several Objective Agents. The number of these varies depending on the phase being worked on by the student, there being one Objective Agent for each learning objective defined for each phase. So, in phase 1 there are four Objective Agents, six for phase 2, and so on. Once the objective is decided upon, the next step is to decide the activity type within said objective to which the exercises to be shown to the student belong. This is done by the Activity Supervisor Agent, which is informed of the winning objective by the Objective Supervisor Agent. There are different activity types for each objective, meaning that, as is the case with Objective Agents, a variable number of Activity Agents will be generated in the system. These will negotiate among themselves under the oversight of the Activity Supervisor Agent (Fig. 6).

Each objective agent wants for the objective that it manages to be the next one to be carried out by the student. To that end, each objective agent evaluates the following objective function:

$$\begin{aligned}\text{Funtion\_Obj}_i &= p_1 * \text{Num\_activ\_obj}_i + p_2 * \text{Degree\_ful\_obj}_i \\ &\quad + p_3 * \text{Ant\_obj}_i\end{aligned}$$

where:

- Num_activ_obj_i: number of activities done of objective *i* versus the number of activities done for the entire phase.
- Degree_ful_obj_i: the degree of fulfillment of objective *i* (percentage of right answers by the student for objective *i*).
- Ant_obj_i: seniority of objective *i*, a measure that indicates the time elapsed since attaining objective *i*.
- p_1, p_2, p_3: weight coefficients.

This is the value that the Objective Agents will use to bid in the auction. As noted previously, a sealed-bid auction is used, so each Objective Agent sends its bid value to the Objective Supervisor Agent as it calculates the Fobj_i value, ignoring the values of the other Objective Agents. Once all the calculations are completed and the Objective Supervisor Agent has all the bid values, the Objective Agent having submitted the highest bid (highest Fobj_i value) is chosen as the winner. The timing diagram is shown in Fig. 7.

Several tests were conducted using different weight coefficients for Fobj_i. Initially the seniority parameter for the objectives was not included in the function. But among the requisites given by the experts for the system to work properly was that not only do the objectives with the worst results have to be emphasized, but also that the activities have to vary so as to further motivate the students and hold their attention. In the initial tests that were run without using this parameter, the same objective could be output several times in a row if the results obtained by the student in that objective were very low. Nevertheless, the expert indicated that the objectives had to be varied because insisting on a same objective could bore the student. Once this parameter was included, an adaptive simulation process was run, with the parameters $p_1 = 0.6$, $p_2 = 0.3$, $p_3 = 0.1$, which yielded the results desired by the experts.

Once the winning objective is determined, we have to choose the type of activity within that objective that is going to be presented next to the student. To do this, the Objective Supervisor Agent communicates this result to the Activity Supervisor Agent and yields control to it. The Activity Supervisor Agent instances as many Activity Agents as there are types of activities comprising the winning objective and indicates the beginning of the auction. Just as in the case of the Objective Agents, each of the Activity Agents is tasked with obtaining the student data from the database and calculating the following function:

$$\text{Funtion\_Act}_i = p_1 * \text{Num\_activ}_i + p_2 * \text{Degree\_ful\_act}_i$$

where:

- Num_activ_i: number of activities done of type *i* versus the number of activities done for that objective.
- Degree_ful_act_i: the degree of fulfillment of activity *i* (percentage of right answers by the student for the type of activity *i*).
- p_1, p_2, p_3: weight coefficients.

As in the previous case, the weight coefficients are adjusted so that, given two activities of the same importance level, the

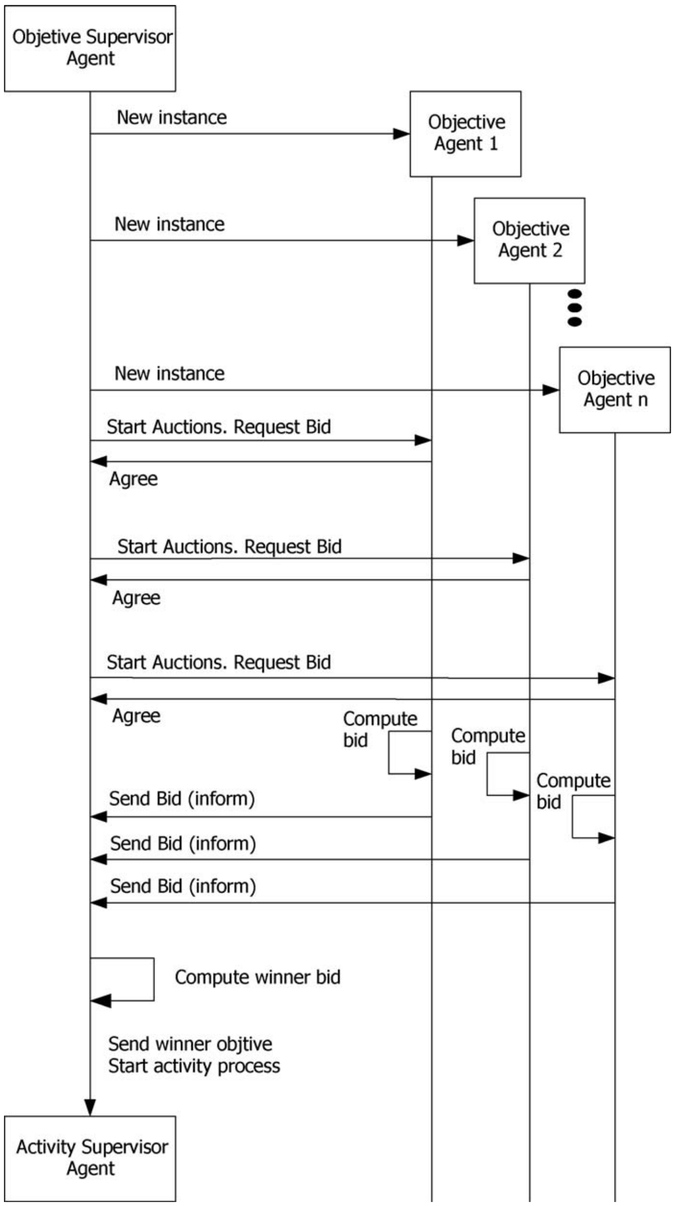

Fig. 7. Intelligent agents sequence.

activities are varied so as to improve the student's motivation. In addition, if the student has negative results in an activity, it is necessary to show more activities of that type so that she can practice and improve her level in that activity. The parameters used were $p_1 = 0.7$, $p_2 = 0.3$.

Each objective consists of several activity types. Some of the activities within the objective have a higher priority than others. For that reason the adjustment of the weight coefficients depends on whether the activity has a high priority or not. The different simulations conducted taking into account high-priority and non-high-priority activities determined that the non-high-priority activity never won the auction, Fig. 8.

As a result, the authors decided to multiply Funtion_Act by the priority function, which is generated in the following way:

$$\text{Priority\_Function}_i = \frac{100 * \text{Quantum\_activ}_i}{\text{Num\_activ}_i}$$

In order to calculate the quantum bases of activity, we used a value of 100 for objective and calculated the value corresponding to

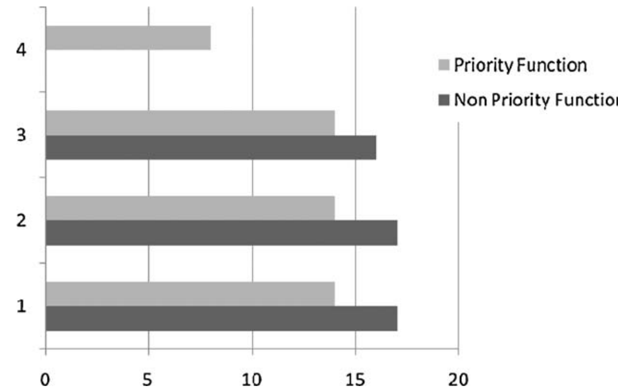

Fig. 8. Number of each type of activity.

each activity by considering the fact that the high-priority activities weigh 3 times more than the non-high-priority ones. Quantums for different cases are calculated as shown in Table 4.

The weight coefficients are adjusted so that given two activities of the same importance level, the activities are varied so as to improve the student's motivation. In addition, if the student has negative results in an activity it is necessary to show more activities of that type so that he can practice and improve the level in that activity. The parameters used were $p_1 = 0.7$, $p_2 = 0.3$.

6. Verification and validation

The qualitative validation relied on a series of cases (observations) for both the fuzzy and MultiAgent Systems that allowed for the results obtained from the tutorial simulation to be compared with what the experts would propose for said cases. The series of cases complied with the two essential requirements for the sample to be significant: quantity (that the number of cases be significant) and representativeness (that the cases exhibit variety within the application domain). These cases were validated by a group of experts, which offered the advantage of having the opinions of several experts (as opposed to that of a single expert or a consensus opinion). This results in fewer errors and also allows for the degree of agreement or similarity between the experts' responses to be compared.

One sample case for validating the fuzzy system is as follows: consider a hyperactive student doing medium-level activities with a bad record of progressing through said activities. If, when doing new activities, his correct answer percentage is 50%, what should be the next difficulty level assigned to the student? Should he be kept at the same level, dropped to the low level or raised to the high level?

The statistical measures used for the quantitative validation are measures of agreement that compare the results of the tutorial with each expert's interpretation for the cases proposed.

Fig. 5 shows a summary of the validation results for the fuzzy system that is used to determine the level of the next activity to be shown to the student. It shows the results of the validation as conducted by 15 experts, both those who participated in the development of the project and those not involved in it. Fig. 9 shows the value for the agreement index, weighted kappa, tau b, gamma and Spearman's rho [19]. It is interesting to note that all of the values obtained are, on average, above 0.80. The Gamma statistical value is very close to 1, which indicates a strong correlation between the results of the experts and those proposed by the fuzzy planner.

For the MAS, as with the fuzzy system validation, we show the results of the validation conducted by 15 experts for both the objectives and the activities. The validation values are shown for the agreement index, weighted kappa, lambda and Goodman–Kruskal

Table 4
Quantum bases for each activity.

Example	# Priority activities	# Non-priority activities	Quantum bases
1	1	1	100 = 3y + y QB Act1 = 75, QB Act2 = 25
2	3	1	100 = 3y + 3y + 3y + y QB Act1 = QB Act2 = QB Act3 = 30 QB Act 4 = 10

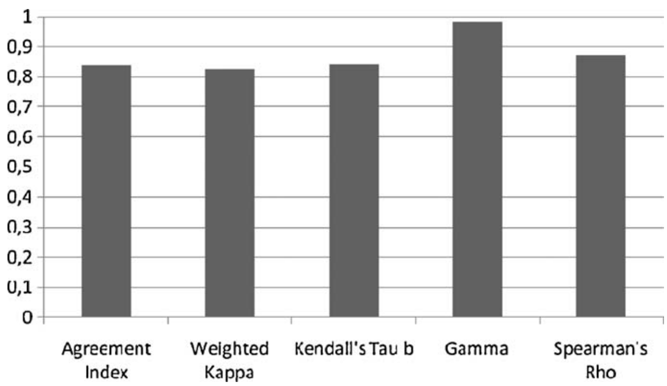

Fig. 9. Results of expert validation for the fuzzy system.

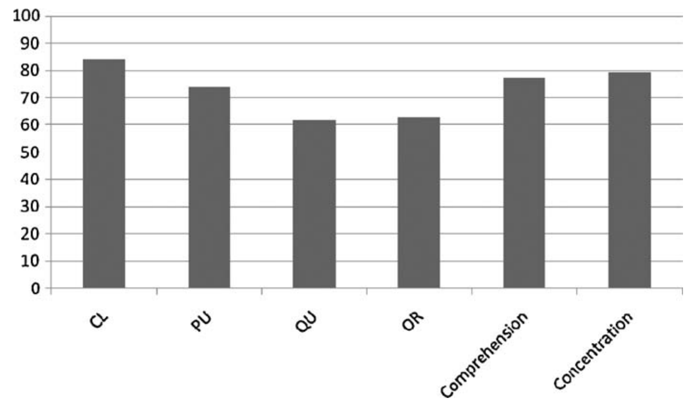

Fig. 11. Percentage of right answers to activities, student understanding and concentration.

tau (Fig. 10) [19]. Note that the agreement index yielded a value above 0.80, this result being somewhat smaller for the predictive association measures, though the results, on average, are above 70%.

At this moment the system is being used in some schools in Spain and Brazil for the purpose of validating and measuring its actual benefits.

The qualitative validation of phase 1 is complete and was conducted in three stages. First, the students worked on the activities separately, then with set sequences of activities so that a comparison could be made of the students' performance independently of the level of difficulty of the activities, and subsequently with the ITS. The validation was carried out in cooperation with the faculty of the Mathematical Analysis Department (area of Mathematics Education) of the University of La Laguna [20]. The data analyzed were obtained from the answers to the classification, ordering, pair up and quantifier activities. In total, the students worked with 174 activities: 31 on classification (CL), 56 on ordering (RO), 36 on pair up (CO) and 77 on quantifiers (CU). Since the tutorial shows the students different activities based on their answers, the analysis takes into account the overall results.

The percentage of right answers given by the students for these activities is shown in Fig. 11. Every concept resulted in percentages above 60%, though the ordering activities posed the greatest challenge to the students [20]. The students' understanding of the action to be performed in the ITS was also evaluated for almost every activity presented, revealing more doubts regarding the mathematical than the technological concepts (Fig. 11). Also analyzed was the students' concentration when working with the ITS, since the interest and attention shown by the students when doing the activities has an obvious bearing on the correctness of their answers. Fig. 11 shows the percentages for those activities during which the students exhibited concentration and attention throughout the performance of the activity. The interest and level of attention when engaged in ITS activities is proof of the significant motivation on the part of the students, as evidenced by the concentration when doing the activities and the interest in continuing with the activities at the conclusion of the work sessions.

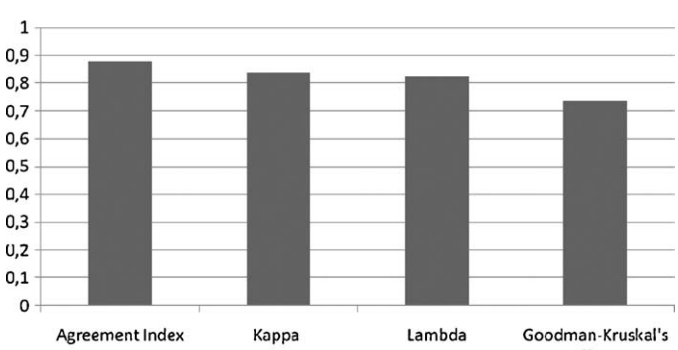

Fig. 10. Results of expert validation for the MultiAgent System.

Phase 2 of the ITS is currently undergoing validation.

7. Conclusions

In this paper we presented the Fuzzy and Multiagent Instructional Planner for an Intelligent Tutorial System. A fuzzy system was modeled to define the difficulty level of the activities to be carried out by the students. In addition, we designed a MAS to determine objectives and activities for each user. Our experience is that the teaching of the addition operation is satisfactorily modeled by the combined use of fuzzy and MultiAgent methods. Thanks to the incorporation of this combined technique, we were able to improve the strategy for learning objectives while at the same time decreasing the wait time involved in interacting with the user.

The ITS was verified and validated with help from both experts in mathematics and from instructors in other areas of knowledge. To conduct the verification, the Instructional Planner was divided into two subsystems (fuzzy and MAS), each of which was verified separately in preparation for a validation of the system as a whole and a check of the proper operation of its combined parts. The ITS is currently being validated with primary school children in Spain and Brazil. The results for the validation of phase 1 of the ITS have been good in terms of the success of the students when working with the activities as well as in their understanding of said activities. Moreover, the interest and attention shown when doing the Tutorial activities is proof of the students' high degree of motivation. We are currently engaged in validating Phase 2 of the ITS.

References

[1] R.M. Aguilar, V. Muñoz, M. Noda, A. Bruno, L. Moreno, Teacher strategies simulation using fuzzy systems, Computer Applications in Engineering Education 18 (1) (2010) 183–192.

[2] J.A. Elgorriaga, I. Fernández, Using case-based reasoning in instructional planning. Towards a hybrid self-improving instructional planner, International Journal of Artificial Intelligence in Education 11 (2000) 416–449.

[3] G. Weiss, Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence, MIT Press, Cambridge, Mass, 2000.

[4] J. Liu, The use of fuzzy reasoning in intelligent computer aided instructional systems, in: Proceedings of the Eighteenth International Symposium on Multiple-Valued Logic, issue 24–26, 1988, pp. 346–351.

[5] S. Ammar, W. Duncombe, B. Jump, R. Wright, Constructing a fuzzy-knowledge-based-system: an application for assessing the financial condition of public schools, Expert System with Applications 27 (2004) 349–364.

[6] J. Liu, The use of fuzzy reasoning in intelligent computer aided instructional systems, in: Proceedings of the Eighteenth International Symposium on Multiple-Valued Logic, 1988, pp. 346–351.

[7] R. Nkambou, F. Kabanza, Designing Intelligent Tutoring System: a multiagent planning approach, ACM SIGCUE Outlook 27 (2) (2001) 46–60.

[8] T. Ganjanasuwan, C. Sanrach, Multi-agent instructional resource planning, in: IEEE Conference on Cybernetics and Intelligent Systems, 2006, pp. 1–7.

[9] L. Sheremetov, A. Arenas, EVA: an interactive web-based collaborative learning environment, Computers & Education 39 (2002) 161–182.

[10] T. Murray, Authoring intelligent tutoring systems: an analysis of the sate of the art, International Journal of Artificial Intelligence in Education 10 (1999) 98–129.

[11] M.E. Hospers, A. Kroezen, H.J.A. op den Akker Nijholt, D. Heylen, An agent-based intelligent tutoring system for nurse education, in: J. Nealon, A. Moreno (Eds.), Applications of Intelligent Agents in Health Care. ser. Whitestein Series in Software Agent Technologies, Birkhauser, Basel, Switzerland, 2003, pp. 143–159.

[12] S.-y. Choi, H.-j. Yang, A Fuzzy Set Based Tutoring System for Adaptive Learning, Lecture Notes in Computer Science 3137 (2004) 407–427.

[13] G.D. Magoulas, K.A. Papanikolaou, M. Grigoriadou, Neuro-fuzzy synergism for planning the content in a web-based course, Informatica 25 (2001) 39–48.

[14] J. Vassileva, B. Wasson, Instructional planning approaches: from tutoring towards free learning 1, in: Proceedings of Euro-AIED'96, Lisbon, 1996, 30.9.-2.10, pp. 1–8.

[15] J.S.R. Jang, C.T. Sun, E. Mizutani, Neuro-fuzzy and Soft Computing, A Computational Approach to Learning and Machine Intelligence, Prentice Hall, 1997.

[16] K.M. Passivo, S. Yurkovich, Fuzzy Control, Addison Wesley, 1998.

[17] L. Moreno, R.M. Aguilar, C. González, V. Muñoz, A. Bruno, A. Noda, The design of an Intelligent Tutorial for the addition operation in children with learning difficulties. Part I: The fuzzy instructional planner, in: Proceedings of E-Learn, Phoenix, 2003.

[18] M. Yousoof, M. Sapiyan, K. Kamaludin, Applications of Intelligent Tutorial Systems in education, in: International Conference on Knowledge Management (ICKM), 2005.

[19] E. Mosquera, V. Moret, Intelligent interpretation of validation data, Expert Systems with Applications 23 (2002).

[20] R.M. Aguilar, V. Muñoz, C.S. González, M.A. Noda, A. Bruno, L. Moreno, Pensamiento Lógico Matemático en alumnos con Síndrome de Down, in: Actas de las XII Jornadas sobre el Aprendizaje y Enseñanza de las Matemáticas, Albacete, 2005.