

Mamdani's Fuzzy Inference eMathTeacher: a Tutorial for Active Learning*

M. GLORIA SÁNCHEZ-TORRUBIA, CARMEN TORRES-BLANC, SANJAY KRISHNANKUTTY

Applied Mathematics Department (DMA)

Universidad Politécnica of Madrid (UPM)

Campus de Montegancedo s.n. Boadilla del Monte (Madrid)

SPAIN

gsanchez@fi.upm.es, ctorres@fi.upm.es, sankrishnan@hotmail.com

<http://www.dma.fi.upm.es>

Abstract: – An eMathTeacher [16] is an eLearning on-line self-assessment tool that helps users to active learning math concepts and algorithms by themselves, correcting their mistakes and providing them with clues to find the right solution. This paper introduces an example of a new concept on Computer Aided Instruction (CAI) resources, i.e. a tutorial designed under eMathTeacher philosophy for active eLearning Mamdani's Direct Method, and presents a brief survey on available CAI resources discussing what their influence over students' behaviour is. It also describes the minimum and complementary requirements an eLearning tool must fulfil to be considered an eMathTeacher as well as the main contributions of this kind of tutorials to the learning processes. Needless to say that, such features as interactivity, visualization and simplicity turn these tools into great value pedagogical instruments.

Key-Words: – eMathTeacher, eLearning, Active learning, Interactive Java applets, Fuzzy Inference Systems (FIS), Fuzzy Logic, Computer Assisted Instruction (CAI).

1 Introduction

Education systems cannot keep out of the changes that new technologies bring up in our society and technological advances should generate a substantial transformation of our current didactical methodologies. Computer Aided Instruction (CAI) provides new exciting tools for teaching and learning and it is necessary to consider them. Blended courses, taking advantage of computational aids, are turning into a must, especially in computer and engineering education. This way, the power and effectiveness of face to face teaching are enhanced with the flexibility and technical capabilities of eLearning, turning out the students in the protagonists of their own learning progression. Thus, Web based learning technologies play a relevant role in the modern education process.

CAI systems have demonstrated to be useful for every education area. An application of eLearning to a video/audio course can be found on [10]. The authors assert: "the research result shows that the eLearning tool did change the learning attitudes of students and increase the learning effects"

The mentioned technologies have been applied, not only to higher education, but also to primary

school. In [1] the authors present MINDA, a system for supporting mathematical communication in word problem solving. The system has been used by 7th grade students and their results "indicate that MINDA has a positive effect on the students' achievements"

Some investigations have shown that Computer Assisted Instruction has been more effective than traditional methodologies [14, 20, 24]; particularly, visualization technologies have been proved as a very positive aid to the learning task [8, 12, 16]. However, some authors report reluctance to accept a real value of visualization as a pedagogical tool. Such position might be based on the passive attitude some students adopt when using these tools. This drawback and the ways to avoid it are covered in section 4.

Mathematical concepts as well as algorithmic procedures use to be difficult subjects for learners to be caught and CAI usually helps them on the learning task. However, comprehensive research is required to determine the best kind of methodology to be applied for the design and development of computer-assisted training, as well as the efficiency of the teaching/learning processes based on this particular method of instruction [8].

*This work has been partially supported by UPM under Project No. IE07 1010-029 and by CICYT (Spain) under Project TIN 2005-08943-C02-01

It is widely accepted that visualization enhances comprehension, engagement and satisfaction among students. Visualization technologies can be used to illustrate many mathematical concepts and a good graphic interface environment would surely become helpful for a better understanding of the ideas or how the methods may be implemented.

We have been working in this field, developing and testing Interactive Java Tutorials both for being used by teachers on classroom lectures and students when learning by themselves. Those tools are available in our Department website [27, 28 and 30] and we are presenting a brief summary of some of their features in section 2.3.

In this paper we introduce a new tutorial including a set of java applets designed for self-assessment while learning the Mamdani's Fuzzy Inference direct method. Fuzzy logic, introduced for Lofti A. Zadeh in 1965 [26], is a multivalued logic, as its truth values are defined within the $[0, 1]$ interval. It was defined as an alternative to bivalued classic logic which has only two truth values: 0 or 1. Fuzzy Logic has undergone a huge improvement during the last decades as its methods are suitable for being used to solve problems that are difficult to address using traditional techniques. This fact can be ascribed to the main characteristic of fuzzy logic: it is multivalued instead of bivalued which provides flexibility and simplicity when implementing mathematical formalizations based on natural language or working with imprecise data.

The pioneer industrial application of this concept arose in mid-70s, with the first controller based on fuzzy logic, designed by E. Mamdani [11]. Many industries are implementing control systems and knowledge management, by means of Fuzzy Inference techniques, and applying them for different purposes ever since.

Many applications of Fuzzy logic can be found in disparate fields such as modelling the dynamics of spread of epidemics [6], automated cow status monitoring [5], motor speed control system [4], human resources management [3], calculus of Incurred But Not Referred (IBNR) loss reserves in insurance companies [2], database design [22], Website structures selection [18] or expert systems for supporting students and specialists [23]. That's why many people are becoming aware of its importance and need to learn its fundamentals.

The current paper is organized as follows: Within section 2 we define what we have called an eMathTeacher, describing the minimum and complementary requirements we demand these tools should accomplish. We also present some exciting examples of this kind of learning materials and

summarize the benefits these applications provide to education processes together with a brief evaluation of their impact on students' rates.

Section 3 is devoted to Mamdani's Fuzzy Inference eMathTeacher. We briefly explain Mamdani's Inference Direct Method and describe the tool features.

As mentioned above, different web learning resources can be considered depending on their interactivity degree and the impact caused on the students attitude. In section 4 we classify them according to these parameters, based on different examples collected from the technical literature, and discuss their influence on cognitive learning effectiveness.

However, when we started to test the eMathTeachers presented within this paper, we found some weakness related to their functionality and flexibility capabilities. In section 5 we identify and evaluate them and consider what we have done in terms of adding some complementary features to our new designs with the scope of fixing those issues.

2 eMathTeacher tools

An eMathTeacher [16] is an eLearning tool which acts as a virtual maths teacher. In other words: they are on-line self-assessment tools that help students to active learn math concepts and algorithms by themselves, correcting their mistakes and providing students with clues to find the right solution. They can also be applied as complementary material for bLearning [15] both for being used by teachers on classroom lectures and by students when learning maths by themselves. Nonetheless, the most important feature of an eMathTeacher is the feasibility of being used for practicing with math methods and algorithms while the system guides the user towards the right answer.

2.1 Minimum requirements for an eMathTeacher

These are the minimum conditions we consider an eMathTeacher must fulfil:

- Step by step inquiring: for every process step, the student should provide the solution while the application waits in a stand by mode, expecting the user's input.
- Step by step evaluation: just after the user's entry occurs, the eMathTeacher evaluates it, providing a tip for finding the proper answer if it is wrong, or executing it if ok.

- Visualization of the step's changes.
- Easy to use interface: the students' effort would be focused on learning the topic, not the tool.
- Flexible and reliable: allowing the user to introduce and modify the example and to repeat the process if desired.
- Nice and friendly graphic environment, helping insight.
- Clear presentation: focusing the student's attention on the essential concepts instead of getting spread on minor ideas.
- Platform independency: reaching the widest possible audience and keeping right performance when the platform is updated.
- Continuous availability, simplifying users' utilization (*anytime, anywhere*).

2.2 Complementary requirements for an eMathTeacher

The requirements listed above are mandatory for an application to be considered an eMathTeacher. In addition, there are some other desirable conditions that these tools should meet:

- Being included into a complete system covering a whole topic, so that the student can apply different procedures and methods on a unique data set.
- Integrating an algorithm visualization window, able to display the current algorithm step.
- Allowing users to save their work for later review. This is a challenging requirement as java applets are not able to write on the client's hard disk.
- Integrating a library where the teacher may store examples to present during lectures and also to propose practice exercises for the students.
- Including a language menu, so that the tool is translated to several languages, especially English, to broad the targeted users.
- Including an optional automatic execution, especially designed for extremely complex exercises.
- Providing alternative solutions, if existing, once the problem has been successfully completed.
- Adding a theoretical introductory part, containing concepts and results to be used by the procedure.
- Requiring no installation or maintenance tasks: suitable for users belonging to non computer areas and/or freshmen.
- Light downloading weight to avoid heavy transfer processes over the Network both for users and server.

2.3 The first eMathTeachers

The challenge of discovering new ways to motivate students in active learning encouraged us to start developing this new kind of web based tools: the eMathTeachers. Our main goal has always been to get students involved, as actively as possible, in their learning process in order to enhance their comprehension and learning assets.

With this objective in mind, we have developed three different eMathTeachers for active learning graph algorithms, corresponding to DFS & BFS [27], Fleury's [28] and Prim & Kruskal's [30] algorithms. These tools have been implemented as Java applets and are available in our website¹ (Applied Mathematics Department, UPM). When running those applets, once the graph has been drawn and the algorithm has been selected, the applet is able to check whether it can be executed or not. In case not, an error message indicating why the algorithm cannot be applied is displayed, otherwise the algorithm starts. Then, the user should click on the next vertex to be reached or on the next edge to be added. Again, if the selected node or edge is not right, the applet displays an error message showing the problem and providing hints for solving it. If it is right, the application changes the node and/or edge's colour and waits for the user's next entry.

2.4 eMathTeacher tools on active learning

Computer Aided Instruction may be a significant aid for learning as a comparison study [13], performed in Arcadia University (Canada), between a control section and a computer-based studio section of calculus-based Introductory Physics has shown. It proved that there was a statistically significant difference between both groups, where the studio class clearly outperformed the traditional lecture class in all cases. Collectively, their results clearly demonstrated that the studio model was significantly more effective than the lecture model.

According to the conclusions of the working team "Improving the Educational Impact on Algorithm Visualization" [12], when a group of teachers were asked about the benefits they had experienced when using visualization, they got, among others, the following results: 90% stated that teaching experience was more enjoyable, 86% noticed an improved level of student participation, 83% had anecdotal evidence that the class was more exciting for students, 76% thought that visualization provided a powerful basis for discussing conceptual

¹ <http://www.dma.fi.upm.es/java>

foundations of algorithm, and 52% got objective evidence of improved student learning.

Recent approaches to this topic have confirmed that visualization aided interactive tools provide a very positive support to the learning task [8, 12, 16, 17, 21]. The visualization features of eMathTeachers play a relevant role, significantly increasing insight –an image is worth thousand words–, engagement, memorization and satisfaction on the students.

In agreement with the above mentioned studies, we have also experienced that visualization seems to be a very important attribute when learning maths, but, in our opinion, the main characteristic of an eMathTeacher is its full interactivity. Therefore, our major objective when pursuing this feature has been the student's learning task while he/she is required to actively participate in the process. It means, as will be described in section 3, that the user must execute each method step while the tool evaluates and corrects the input.

eMathTeacher tools do enhance teacher's lectures, but their main contribution lies on the possibilities they reveal when used by the students themselves. In Deakin University (Australia) [20], it was demonstrated that students who used interactive tools learned 60% faster, and after 30 days the knowledge kept was from 25% to 50% higher than those who did not use them.

In general terms, it is widely accepted that graphical and dynamical visualizations are more appealing for learners than traditional exercises or text books. However, if the students are not required to give some kind of response or predict what is happening next, they might adopt a passive attitude that is not beneficial at all and might even be harmful to their training. In that sense, it has been verified that learners spend much more study time when visualization is involved, but those who are actively engaged with it have consistently outperformed the other ones who passively viewed it [8]. That is why, during the execution, the program should, in some way, ask the user which must be the next step to be done, instead of just showing it. Here lays the main difference between an eMathTeacher, like which is introduced in this paper, and a not fully interactive (see section 4) or a simple demonstrative visualization tool.

As mentioned above, we have designed several eMathTeachers, mainly implementing graph algorithms, and measured their impact on students' learning [16]. It has been evaluated by comparing the marks obtained by two different groups of students in the graph exercise of Discrete Mathematics final exam (course 06/07). In this

exercise, the students had to apply two algorithms on a particular graph. The exercise was graded on a 0–10 basis, considering passed those who got more than 5 points. The study group used the applets for bLearning while the control group did not use them. Figure 1 and table 1 represent the quantity of students in number and in percentage respectively, obtaining good mark (8–10), passed mark (5–7), failed (2–4) and very poor (0–1).

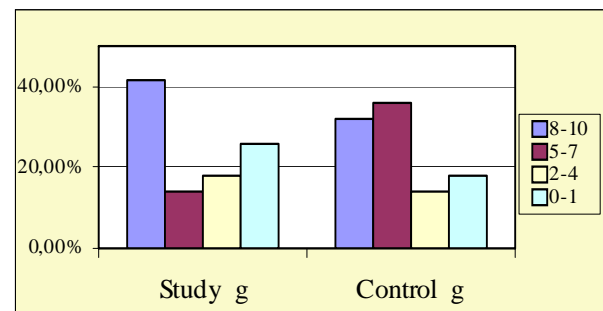


Fig.1: Percentages of students' rates in the graph exercise

	Study g	Control g	Study g	Control g
8–10	32	9	41,56%	32,14%
5–7	11	10	14,29%	35,71%
2–4	14	4	18,18%	14,29%
0–1	20	5	25,97%	17,86%
Total	77	28		

Table 1: Analysis of students' rates related to the graph exercise

Data show (see figure 1 and table 1) a deeper understanding of algorithms process in the study group, as the highest marks percentage is clearly bigger. Moreover, the study group was much larger than the control group, what could explain the higher failure rate.

3 Mamdani's Method eMathTeacher

Keeping in mind the necessities discussed in section 2, we decided to design and implement a comprehensible tutorial (see figure 2) for learning Fuzzy Inference Systems (FIS) that has been created under eMathTeacher philosophy [29].

This tutorial meets all the minimum requirements listed in section 2.1 as well as few other complementary features:

- Including a theoretical introductory part, that covers different concepts and results to be used by the method.
- No installation or maintenance requirements, suitable for learners belonging to non computer areas.

- Extremely low downloading size in order to avoid heavy transfer processes over the Network for both users and server.

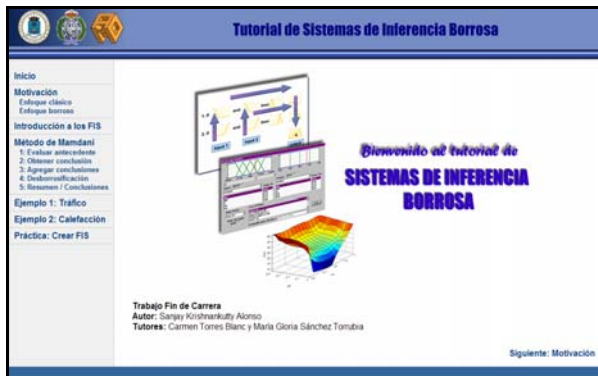


Fig.2: Mamdani's eMathTeacher Home Page

Many tutorials are available in the Internet for learning FIS. Most of them are demonstrative [32, 19], where data are pre defined and the applet shows, usually step by step, the system process. Others are not placed in an open website [9] or/and are implemented on expensive programs such as Matlab Fuzzy Logic Toolbox tutorial [33].

Unlike the tools already referenced, our tutorial carries out, as a differentiator, the following capabilities: fully interaction, easy to use interface and free availability.

3.1 Fuzzy inference systems

Fuzzy inference is the process of transforming an input space into an output space by means of fuzzy logic. This method is able to interpret values in an input vector and, based on if-then rules, assign new ones to an output vector. Fuzzy inference systems try to formalize human language reasoning by using multivalued logic, whose truth values are within the $[0, 1]$ interval, to solve a decision problem.

One of the advantages of using fuzzy (multivalued) instead of classic bivalued logic is its flexibility and simplicity when implementing mathematical formalizations as it is based on natural language and can work with imprecise data.

This explains why fuzzy inference systems have been successfully applied in fields such as automatic control, data classification, decision analysis, expert systems, and computer vision.

3.2 Mamdani's inference method

Mamdani's direct method is the easiest, oldest and most popular fuzzy inference methodology. It was proposed in 1974 by Ebrahim Mamdani [11] when attempting to design a control system, formalizing

the expert's knowledge by means of synthesizing a set of linguistic control rules. Mamdani's work was based on the fuzzy algorithms for complex systems and decision processes already introduced by Zadeh in [25]. This Mamdani's method has a simple structure of min-max operations and is, therefore, very commonly used in applications.

The fuzzy inference process comprises four subsequent steps: evaluating the antecedent for each rule, obtaining a conclusion for each rule, aggregating all conclusions, and finally defuzzifying.

3.2.1 Evaluating the antecedents

This step consists of two parts. First, given the provided inputs (always numerical values) we have to determine their membership degree, i.e. the degree to which they belong to each of the given fuzzy sets, via membership functions (input fuzzification). The membership degree must be always a value belonging to $[0, 1]$ interval.

Then, if the antecedent of a given rule has more than one part, a fuzzy operator (*t-norm*, *t-conorm*, *negation* or *modifier*) is applied to obtain a single value that represents the result of the antecedent for that rule.

3.2.2 Obtaining conclusions

In this step, for each rule, an implication operator is applied both to the single value of the antecedent obtained in the previous step, and to the consequent of the rule (a fuzzy set), obtaining a new fuzzy set which is the conclusion of the rule.

Two of the most commonly used operators are the *minimum*, which truncates the consequent membership function, and the *product*, which scales it.

3.2.3 Aggregating all conclusions

Now, the outputs of the rules, obtained in step two, are combined into a single fuzzy set by using an aggregation operator. The operator implemented in this tool is the *maximum* although some others, such as the *sum* or the *probabilistic or*, are also commonly used.

3.2.4 Defuzzifying

In the way to solve a decision problem, when applying the Mamdani's method, the final target is to get, instead of a fuzzy set, a unique numerical value as the output of the inference process. Therefore, we need to transform the fuzzy set obtained in step three into a single number. This process is called defuzzification, and it is supported in this tool via the *centroid method*, which returns

the centre of the area under the curve, defined by the membership function of the fuzzy set.

3.3 Tool's description

This eMathTeacher includes three differentiated parts. The first one is a theoretical part, split into three sections:

- An overview of the motivation behind the use of fuzzy logic in contrast to classical logic, by means of showing an easy example.
- A general description of fuzzy inference systems and if-then rules.
- A detailed explanation of Mamdani's inference direct method.

For instance, figure 3a illustrates the theoretical description of the centroid defuzzifying method and figure 3b displays a scheme representing the whole Mamdani's direct method.

The second part of the tool consists of two fully interactive examples comprising a couple of series of applets, each one implementing one of the four steps of Mamdani's method, explained in section 3.2. These applets meet all the requirements of an eMathTeacher, providing full interactivity, a user friendly environment and a clear presentation.

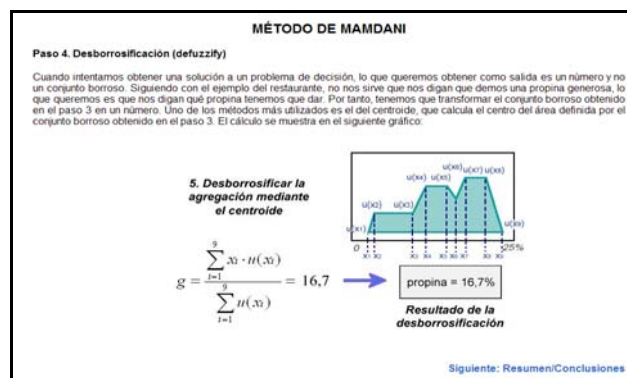


Fig.3a: Theory. Step4: Defuzzifying

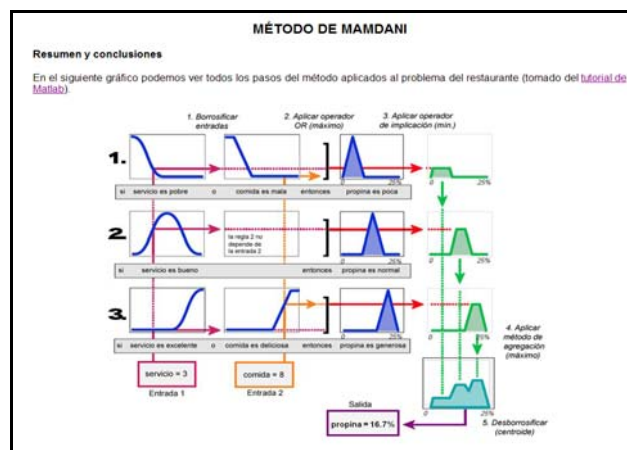


Fig.3b: Theory. Summary/conclusion scheme

The tutorial third part is an applet that executes the whole fuzzy inference process. It lets the user enter different membership functions, define the if-then rules and choose among various aggregating methods in order to implement a complete fuzzy inference system. The applet computes the fuzzy conclusion and calculates the defuzzified output. It also allows selecting from two defuzzifying methods: *centroid* and *maximum*.

The fuzzy inference system presented in the first example is a road traffic management system (by controlling a traffic light timer), whose main purpose is to minimize the waiting time cars spend in a crossroad, as well as the length of the vehicles line. This system is defined as follows: two input variables (rate of car arrivals in a green light, and length of the cars line in a red light) assuming the range of 0 – 30 cars, and one output variable (duration of the traffic light's green phase) within the range 5 – 50 seconds.

The fuzzy inference system presented in the second example is an air conditioning (AC) controller, designed to regulate the air temperature within an area (cooling or heating the water), depending on the air properties and the user wish at a given time. This controller is defined as follows: two input variables, difference in temperature (between desired and current values) in the range [–50°C, 50°C] and time in the range [0, 24] hours; and one output variable, water temperature of the system in the range (0°C, 100°C).

Both examples include two rules: the first one (with two antecedents and one consequent) will be processed by the user, while the second one will be provided (automatically processed) by the system to be aggregated with the first one in the third step.

Within the following sections, we will move thru the detailed description of the application, i.e. how the user covers the Mamdani's inference method while executing the subsequent applets which configures the tool.

3.3.1 Applet 1: Evaluating the antecedents

This applet has been designed for practicing the first step of Mamdani's Inference Direct Method.

The user must choose the numerical values of the input variables, within the permitted range, and the membership function of both antecedents. Then, the user must evaluate the antecedents (with the selected membership functions and the numerical values), select an *OR/AND* operator and finally apply this operator to obtain a single value that represents the result of the antecedent (see figure 4).

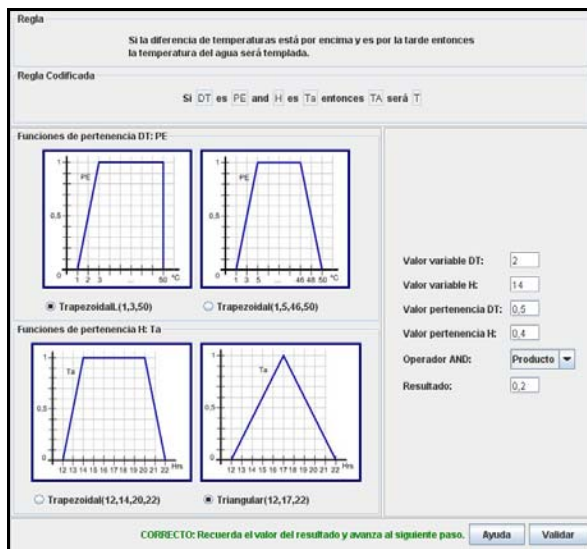


Fig.4: Applet for evaluating the antecedents

The application then checks if the results entered by the user are correct. In case one or more of the results are not the right ones, the application shows an error message indicating that one of the variables is out of range or suggesting to come back and review the theoretical section of the tutorial.

3.3.2 Applet 2: Obtaining conclusions

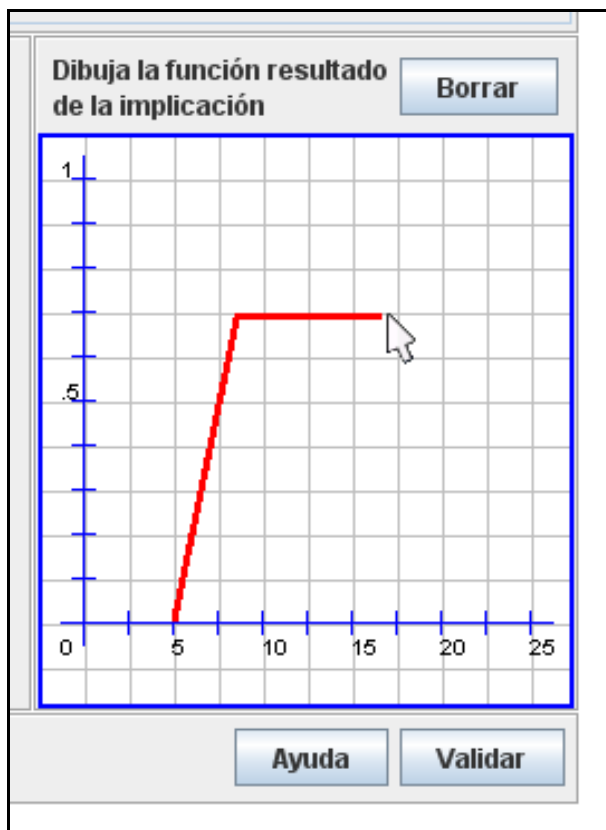


Fig.5a: Drawing panel

Once the user has completed the first step, applet 2 requests him/her to apply the result of the

antecedent in order to obtain the conclusion of the first rule.

It means that the user must enter the membership degree of the antecedent, obtained as a result in the previous applet, select the implication method and choose between both membership functions associated to the consequent of the rule.

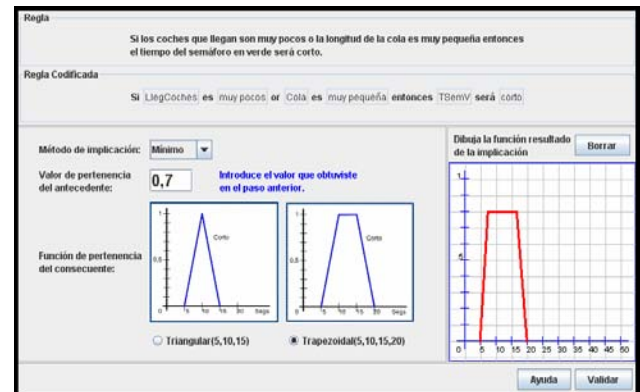


Fig.5b: Applet for obtaining conclusions (traffic)

A drawing panel is provided (see figures 5a and 5b), where the user must draw the resulting fuzzy set of applying the implication method to the membership function of the consequent and the membership degree of the antecedent; this fuzzy set is the conclusion of the rule. This drawing panel becomes the first engaging point for the students.

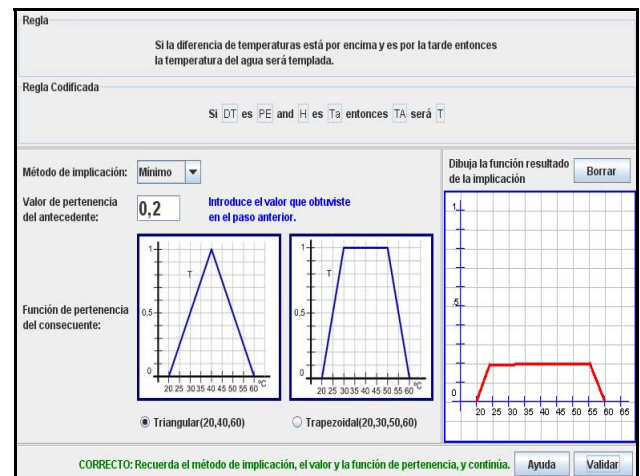


Fig.5c: Applet for obtaining conclusions (AC) showing a correct message

When drawing the fuzzy set, they can truly understand, for the first time, what is the real meaning of using Mamdani's method, as it provides a very intuitional graphic visualization of the implication operator as well as a full interactive interface for testing their knowledge.

The application then checks whether the drawing entered by the user is correct or not (see figure 5c). In case the result is not the right one, the application shows an error message (see figure 6).

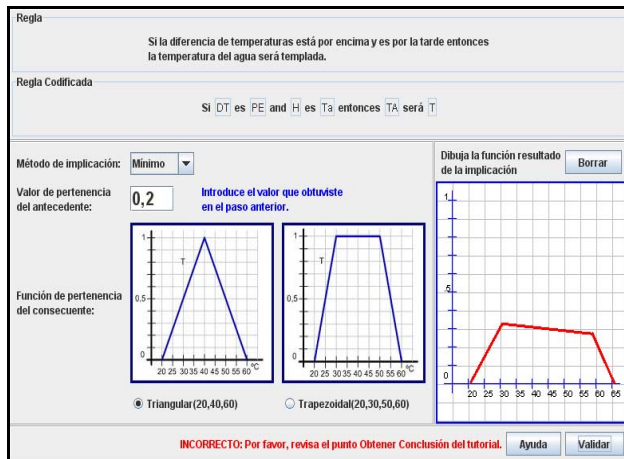


Fig.6: Applet for obtaining conclusions (AC) showing an error message

3.3.3 Applet 3: Aggregating conclusions

In this applet, the user aggregates the conclusion of the first rule, which has been processed in the two previous applets, together with a second rule, whose associated conclusion fuzzy set is also given in the example.

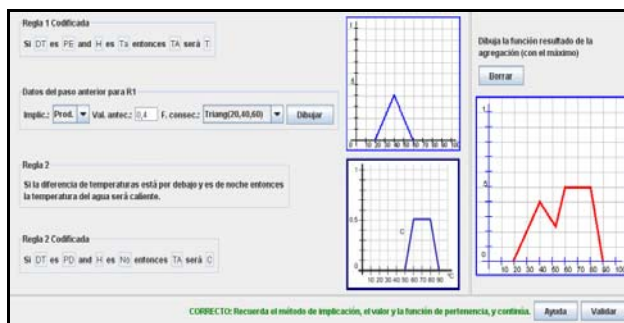


Fig.7: Applet for aggregating conclusions (AC) showing a correct message

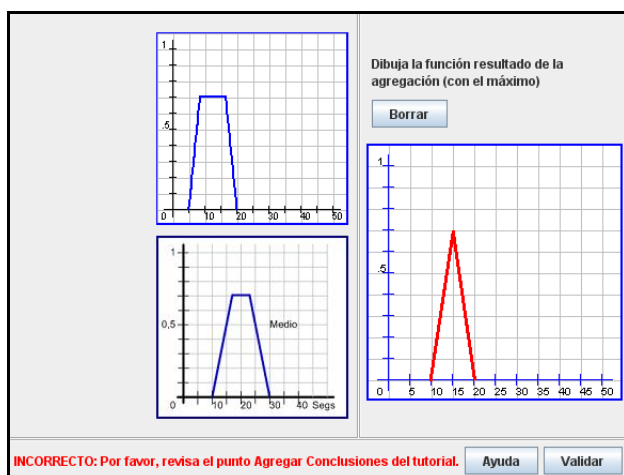


Fig.8: Applet for aggregating conclusions (traffic) showing an error message

Then, the user must enter the membership degree of the antecedent obtained in the first applet, and both the implication method and the membership

function of the consequent, selected in the second applet. The application will then draw the conclusion fuzzy set for the first rule.

At this point, a second drawing panel is provided (see figure 7), so that the user can draw the fuzzy set obtained as a result of applying the aggregation method (*maximum*) to both previous fuzzy sets.

This set represents the final conclusion of the fuzzy inference system. The application then checks if the drawing entered by the user is correct. In other case it shows an error message (see figure 8), as it is designed for the previous applets.

3.3.4 Applet 4: Defuzzifying

This applet differs on the previous ones in the sense that it does not provide the same degree of interactivity, on the contrary, it just shows the calculations it has made to defuzzify the conclusion fuzzy set via the centroid method.

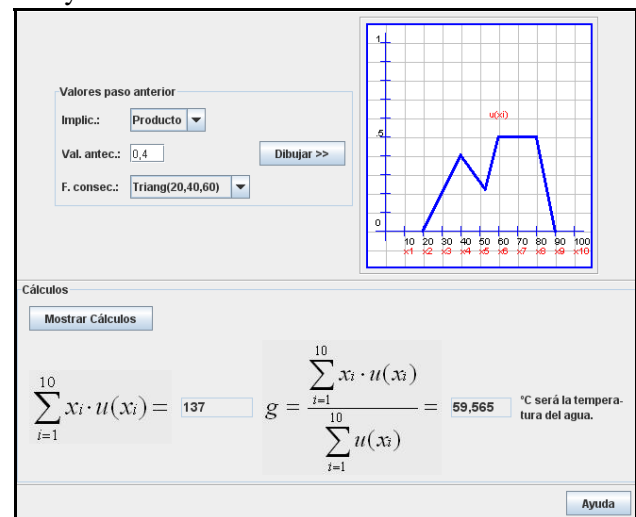


Fig.9: Applet for defuzzifying (AC)

The user must enter the membership degree of the antecedent obtained in the first applet, and both the implication method and the membership function of the consequent, selected in the second applet. Then, the application will draw the conclusion fuzzy set based on the aforementioned parameters and calculate the final result of the inference process with the centroid method (see figure 9).

3.3.5 Applet 5: Fuzzy Inference System

This applet is the third part of the tutorial and implements a whole fuzzy inference system by using Mamdani's Direct Method, practiced with the previous applets, this time in one step. It allows the user to define two input and one output variables with their respective ranges (see figure 10).

For each variable, the user can define three possible linguistic variables and there is a choice of

five predefined types of membership functions to be selected for each linguistic variable, with adjustable parameters.

To help the user when selecting the parameters for the membership functions, there are links to several drawing function applets contained in the Introduction to Fuzzy Logic tutorial, available in DMA website [31] (see figure 11).

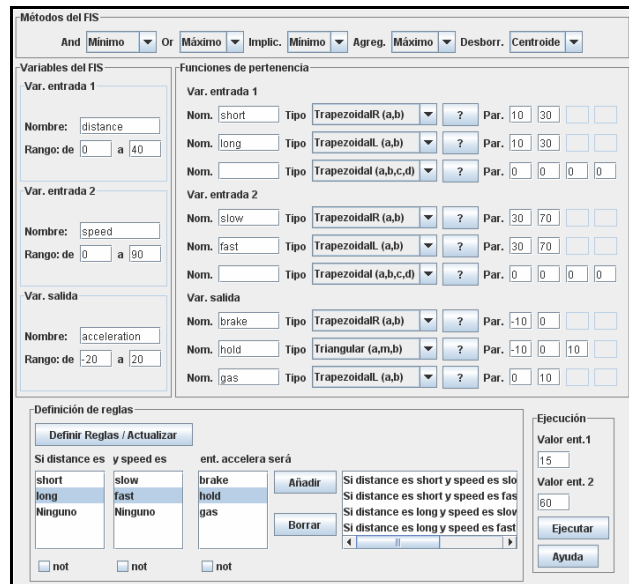


Fig.10: Fuzzy Inference System Applet

Once all the variables have been defined and their membership functions selected, the learner should create the rules by clicking on the linguistic variables. Then, the applet will provide a numeric output corresponding to the numeric values of the two input variables (that have been introduced by the user), obtained after applying Mamdani's Direct Fuzzy Inference Method (see figure 12).

4 Analysis and Discussion of eLearning resources

In [14], Reeves and Hedberg assert that "Cognitive tools empower learners to design their own representations of knowledge rather than absorbing knowledge representations preconceived by others" and that "Well-designed interactive learning systems may be the ideal vehicle for this transformation. The learner should be responsible for recognizing and judging patterns of information, organizing data, constructing alternative perspectives, and representing new knowledge in meaningful ways, while the computer should perform calculations, store information, and retrieve it upon the learner's command. For example, when multimedia is used by learners as a cognitive tool for representing their own knowledge, the teacher

can be a coach or even a collaborator in the knowledge construction process".

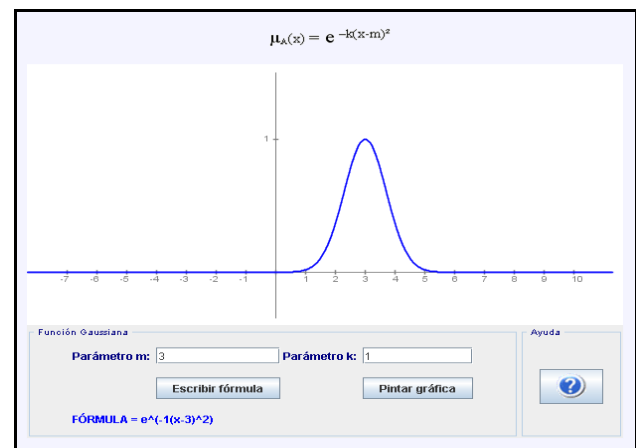


Fig.11: Drawing membership functions help

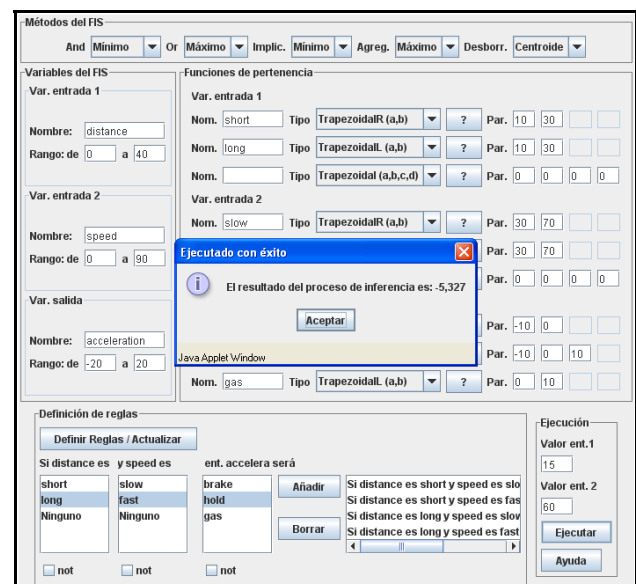


Fig.12: Fuzzy Inference System Applet showing the final output

That is why, in our opinion, a true virtual teacher should do nothing but expect the students' response and guide them towards the right solution. So, the only way for building up their knowledge, is participating actively in the learning process.

In this sense, the analysis presented by Hundhausen et al. [8] asserts that "how students use AV² technology, rather than what students see, appears to have the greatest impact on educational effectiveness" and their study "suggests that the most successful educational uses of AV technology are those in which the technology is used as a vehicle for actively engaging students in the process of learning algorithms". They concluded that, those who are actively involved with the visualization

² Algorithm Visualization

have consistently outperformed the other ones who passively viewed them. Thus, in order to avoid a “lazy” attitude when using an eLearning tool, the program should request a continuous interaction from the users’ side, forcing them to predict the following step.

Many applications and complete systems for Computer Aided Instruction can be found in the web although not all of them take into account the necessity of actively engage students. So, they present disparate interactivity degrees and thus, obtain extremely different results. As a consequence of our research, we have classified math learning tools into three groups, depending on their interactivity degree [16]:

- Demonstrative. It just requires a passive observation while the application is being executed.
- Interactive. It allows users to interact with the system in some way, mostly introducing input data in the application.
- eMathTeacher. It is fully interactive, which means that it is not possible to get a result without the user full engagement.

As stated in section 2.4, learners prefer graphical and dynamical tools better than traditional learning materials but, with the demonstrative tools, the students often adopt a dangerous passive attitude.

This situation can be partially prevented by allowing users to introduce the data into the application. Interactive tools encourage learners to experiment with data variations, providing good understanding of algorithm’s results. The problem with these applications lies in the unfeasibility to control students’ comprehension and acquisition of the required skills. As the system provides the execution path, the student tends to get the feeling *‘it is so easy that I would do it if I tried’*. And, of course, this is a very dangerous belief as they will never overcome it if they do not try several times and face the problems by themselves.

To be an eMathTeacher, the tool should do nothing but expect (and execute or correct) users’ entry. In our research, we have not been able to find any math learning web resource fulfilling all the requirements listed in section 2.1. However, we have seen an eLearning tool designed for programming code learning (not for math learning), with quite similar features: AulaWeb Self-Assessment Module [7]. The application consists in a java applet that emulates a Virtual Programming Environment where the student introduces a specified piece of code. The environment also includes a menu bar with options to compile and run

the complete program. Those compilation and run options return customized error messages, indicating the student whether the code was or not correct and which the error was. This feature makes the tool quite similar to what we have developed.

In summary, the main contributions of eMathTeachers for visualization and learning of maths concepts are:

- A change in the students’ method of learning, providing them with adapted powerful tools.
- Enhancement of concepts and processes comprehension by means of using graphical visualizations as a way to facilitate theory insight.
- Improvement of experimentation and test opportunities, reaching full interactivity and thus enabling users’ participation.
- Higher effectiveness on students’ understanding, involvement, performance and satisfaction.

In [8] the authors state that “AV technology has been successfully used to actively engage students in such activities as what-if analysis of algorithmic behavior, prediction exercises and programming exercises”. The eMathTeacher philosophy has been mainly inspired by getting the students involved in the two first activities: what-if analysis of algorithmic behavior (what means deep understanding of the algorithm design) as well as prediction of the algorithm outcomes (that entails a good comprehension of the algorithm process). In our opinion, as a consequence of the above described improvements, eMathTeachers get the users even more engaged in the two first activities, obtaining as a result a stimulating active learning and thus, hopefully, successful educational results.

5 Next steps

We have described the didactical benefits of eMathTeachers; however, as we are implementing them by means of Java applets, there are some features we lack and are currently trying to include. One of the Java applets’ handicaps is that, due to security reasons, they are not allowed to write in the client’s hard disk. It means that users cannot save their work for later review. The problem is being solved by using Java Web Start.

Another feature we missed is related to language capabilities. Currently, the applets are all just implemented in Spanish. Therefore, we are working on adding a language menu, starting with Spanish and English and expecting to get some support for being translated to other languages.

We are also thinking of creating new eMathTeachers able to implement other Fuzzy Inference Systems, such as Sugeno's method. In the field of graph algorithms we are in the process of designing new tools, as well as updating and improving the oldest ones, such as Dijkstra's algorithm applet (which is not an eMathTeacher at this moment). Dijkstra's algorithm new application is already being tested and it will meet all the minimum requirements and most of the complementary ones, necessary for being an eMathTeacher. Actually, the highlighting new feature of this application will be an animated algorithm visualization panel, able to display, on the code, the current step and the user's mistake. Other relevant new attribute included will be an active framework area for the algorithm data, where the user can modify the algorithm structures while the application verifies the correctness of the input. Finally, the application will offer the feasibility of saving and retrieving graphs.

Currently, we are designing and developing a whole suite of programs which actually integrates different types of graphs and graph algorithms fitted in it. The new design includes the panels described above and will feature the mentioned functions together with the possibility of saving the user's interaction history for later analysis and/or assessment.

We are also working on the design of tools dedicated to other learning areas of mathematics, such as Karnaugh maps and Quine McCluskey's algorithm oriented to simplification techniques of Boolean functions or Gauss method focused on solving linear equation systems.

6 Conclusions

The tutorial presented in this paper is a good example of eMathTeacher as it meets all the minimum requirements and has been designed for supporting active learning. Full interactivity, easy to use, intuitive and visual are the main characteristics kept in mind during design and implementation phases. Actually, these qualities have demonstrated to help achieving increased insight and engagement for the students when working by themselves. In our experience, eMathTeacher tools happen to be a very good aid for learning as they improve comprehension, memorization and satisfaction for the students.

As mentioned in section 2.4 the evaluation of impact on the learning results showed a deeper understanding of algorithmic processes in the study

group, as the highest marks percentage is clearly bigger.

eMathTeacher active learning tools introduce a new concept in mathematics Computer Aided Instruction and represent a revolution in this field. These tools pursue a new goal on CAI, so as to acting as genuine virtual trainers extending teacher's hand through the Web, promoting active learning, increasing effectiveness and offering the enhanced insight and appeal provided by graphical and dynamical tools.

References:

- [1] Ahmad, A., Salim, S.S., Zainuddin, R. A Cognitive Tool to Support Mathematical Communication in Fraction Word Problem Solving. *WSEAS Transactions on Computers*. Vol. 7, Issue 4, 2008.
- [2] Andrés Sánchez, J. Determinación de las IBNR con lógica borrosa. *Anales del Instituto de Actuarios Españoles*, No. 9, 2003, pp. 11–40.
- [3] Canós Darós, L. Gestión de recursos humanos basada en la lógica borrosa. *Revista Electrónica de Comunicaciones y Trabajos de ASEPUMA*, No. 6, 2005, pp. 29–60.
- [4] Dannenberg, A. et al. Fuzzy Logic Motor Control with MSP430x14x. *Texas Instruments: Application Report*. SLAA235–February 2005.
- [5] de Mol, R.M., Woldt, W.E. Application of fuzzy logic in automated cow status monitoring. *Journal of Dairy Science* Vol. 84 No. 2, 2001, pp. 400–410.
- [6] Di Stefano, B., Fuks, H., Lawniczak, A. Application of fuzzy logic in CA/LGCA models as a way of dealing with imprecise and vague data., *2000 Canadian Conference on Electrical and Computer Engineering*. Vol. 1, pp. 212–217.
- [7] García-Beltrán, A., Tapia, S., Martínez, R., González, M. Automatic Assessment of Multi-language Programming Questions using AulaWeb. *WSEAS Transactions on Computers*. Vol. 6, Issue 3, 2007, pp. 449–454.
- [8] Hundhausen, C.D., Douglas, S.A. And Stasko, J.T. A Meta-Study of Algorithm Visualization Effectiveness. *Journal of Visual Languages and Computing*, 13, 2002, pp. 259–290.
- [9] Jantzen, J. Fuzzy Control (Internet Course). <http://fuzzy.iau.dtu.dk/lspace3/central.nsf> (last visited October 2007).
- [10] Lien, W.-P., Chang, M., Chu, K.-K. Using Internet Technology to Make Students Learning Better in Video/Audio Design Course. *WSEAS Transactions on Computers* Vol. 6, Issue 3, 2007, pp. 433–440.

- [11] Mamdani, E.H. "Application of Fuzzy Algorithms for Control of Simple Dynamic Plant, *Proceedings of IEEE*, Vol. 121, No. 12, 1974, pp. 1585–1588.
- [12] Naps, T.L., Rößling, G., Almstrum, V., Dann, W., Fleischer, R., Hundhausen, C., Korhonen, A., Malmi, L., McNally, M., Rodger, S., Velazquez-Iturbide, J.A. Exploring the Role of Visualization and Engagement in Computer Science Education. *Inroads – Paving the Way towards Excellence in Computing Education*. Vol. 35, Issue 2, 2003, pp. 131–152.
- [13] Retson, D., Williams, P.J., Simmons, S. The Effectiveness of Computer-Based Studio Teaching of Physics. *Physics in Canada*, 59, 2003, pp. 201–204.
- [14] Reeves, T.C., Hedberg, J.G. *Interactive Learning Systems Evaluation*. Educational Technology Publication. Englewood Cliffs, 2003.
- [15] Sánchez-Torrubia M.G., Torres-Blanc, C. Giménez-Martínez, V. An eMathTeacher Tool for Active Learning Fleury's Algorithm. *International Journal Information Technologies and Knowledge (IJ ITK)* Vol. 2, 2008 (accepted).
- [16] Sánchez-Torrubia, M.G., Torres-Blanc, C., Castellanos, J. Defining eMathTeacher Tools and Comparing them with e&bLearning web based tools. *Proceedings of the International Conference on Engineering and Mathematics (ENMA 2007)*, Bilbao, Spain.
- [17] Sánchez-Torrubia M.G, Giménez-Martínez, V. Java Tutorials: a good tool for teaching and learning Graph Algorithms. *Proceedings of the 3rd International Conference on the Teaching of Mathematics (ICTM3)*, Istanbul, Turkey, 2006.
- [18] Saremi, H.Q., Montazer, G.A. An Application of Type-2 Fuzzy Notions in Website Structures Selection: Utilizing Extended TOPSIS Method. *WSEAS Transactions on Computers*, Vol. 7, Issue 1, 2008, pp. 8–15.
- [19] Schmid, C. Course on Dynamics of multidisciplinary and controlled Systems. (2005) <http://virtual.cvut.cz/dynlabmodules/ihhtml/dynlabmodules/syscontrol/main.html> (last visited October 2007).
- [20] Street, S., Goodman, A. Some experimental Evidence on the Educational Value of Interactive Java Applets in Web-based Tutorials. *Proceedings of the 3rd Australasian Conference on Computer Science Education*. ACM, 1998, pp. 94–100.
- [21] Torres-Blanc, C., Sánchez-Torrubia, M.G., Aguilar, J., Castiñeira-Holgado, E. Analyzing and Applying Computer Algebraic Systems to Engineering Education: an Interactive Digital Control course *WSEAS Transactions on Advances in Engineering Education* Vol. 3 N° 2, 2006, pp. 977–983.
- [22] Urrutia, A., Varas, M., Galindo, J. Diseño de una base de datos difusa modelada con UML, 6° *Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes Software, IDEAS 2003*, Asunción Paraguay, pp. 145–155
- [23] Urueta Vivanco, L., Valdés Barrios, H., Contreras Montes, J. La lógica difusa como apoyo a la enseñanza. *Actas del VIII Congreso Internacional de Informática en la Educación (INFOREDU 2002)*
- [24] Waldock, J., Gretton, H., Challis, N. Using the web to enhance student learning. *Proceedings of the 2nd International Conference on the Teaching of Mathematics*, 2002, Crete, Greece.
- [25] Zadeh, L.A. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 3, No. 1, 1973, pp. 28–44.
- [26] Zadeh, L.A. *Fuzzy Sets*, *Inf. Control* No. 8, 1965, pp. 338–353.
- [27] Depth and Breadth first search (DFS & BFS) algorithms. www.dma.fi.upm.es/java/matematicadiscreta/busqueda/
- [28] Fleury's algorithm. www.dma.fi.upm.es/java/matematicadiscreta/fleury/
- [29] Fuzzy Inference Systems Tutorial. www.dma.fi.upm.es/research/FundMatSoftComputing/FuzzyInf/
- [30] Prim and Kruskal algorithms www.dma.fi.upm.es/java/matematicadiscreta/Kruskal_prim/
- [31] Tutorial of Introduction to Fuzzy Logic <http://www.dma.fi.upm.es/research/FundMatSoftComputing/tutfuzzy/>
- [32] Tutorial of Fuzzy Logic Toolbox. The MathWorks. www.mathworks.com/access/helpdesk/help/toolbox/fuzzy/index.html (last visited October 2007).
- [33] Fuzzy Logic Toolbox for Matlab. The MathWorks.