# An Intelligent Learning System within a Learning Object

Lourdes Sánchez-Guerrero[1], Ana Lilia Laureano-Cruces[2], Martha Mora-Torres[3] and Javier Ramírez-Rodríguez[4]

[1,2 y 4] Universidad Autónoma Metropolitana
Azcapotzalco, Departamento de Sistemas
San Pablo 180, 02200 Mexico City, Mexico
{lsg, clc, jararo}@correo.azc.uam.mx

[3] Posgrado en Ciencia e Ingeniería de la Computación
IIMAS-Universidad Nacional Autónoma de México
Circuito escolar, 04510 Ciudad Universitaria. D.F., Mexico
kabhun@yahoo.com.mx

**Abstract:** The following paper describes the analysis and design of an Intelligent Learning System (ILS) whose domain is located in structured programming. The latter will be immersed in a Learning Object. The ILS includes artificial intelligence techniques that support the implementation of the teaching-learning process. Two representative scenarios are described. The interaction of the system is based on an inference engine created for general teaching. In this case, the scenarios adapt to the domain of the case studied. The project aims to use artificial intelligence techniques alongside new technologies in order to create systems that provide the user with both: knowledge of a specific domain, as well as the ability to count on self-regulatory learning strategies.

**Key words:** intelligent learning systems, instructional objectives, model student and learning objects.

## 1. Introduction

At the present time, with current progress in information technology and communications, there has been an increase in possibilities for the development of alternative educational methods, which in some way allow us to more effectively access more clients in the learning-teaching process.

Right now, a large variety of electronic media are used to send or receive support materials in order to provide Distance Education (DE). This has led to the birth of e-learning. This term refers to the use of new information and communication technologies for learning. One of these technologies is the internet, but we can also include the concept of multimedia and simulators. We could say that e-learning is a distance education method where both the teacher and the student use electronic media to carry out the teaching-learning process.

These new media are being used to express knowledge, to present information and to guide learning activities in relation to materials. The increase in the wealth of content no doubt represents a change in the educational context, but it will continue to be only a minor change if it is not accompanied by more in-depth transformations such as: 1) reorganizing educational content, 2) the way said content is accessed, and 3) its use in the teaching-learning process.

Likewise, distance education allows for the reduction of costs linked to training and for dealing with educational needs in groups located in different geographical areas.

The following article is organized as follows: *section two* briefly describes intelligent learning systems, with emphasis being placed on the tutorial model based on a general didactic model and also describes the proposal of this paper. *Section three* describes the cognitive didactics which includes instructional objectives. *Section four* describes the student model, errors and their classification. *Section five* provides two scenarios for the case studied. *Section six* briefly describes the learning objects, finishing with conclusions in *section seven*.

## 2. Intelligent Learning Systems

Intelligent learning systems (ILS) are made up of four components: 1) the expert module, 2) the student model, 3) the interface, and 4) the user. A brief description will be provided below. For more information, consult Laureano-Cruces, & de Arriaga (2000) and Laureano-Cruces (2000).

**The expert module** is where knowledge that the system tries to teach the student is collected. The implementation of this component is closely linked to the tutor module due to the fact that the tutor will teach the domain by placing emphasis on the organization of the expert module. Therefore, it is important that this module be organized in a pedagogical way.

**The student model** is a database that contains information about the student, thus allowing us to do the following: 1) adjust the system based on the student's ability in relation to a specific subject (teaching object), 2) prepare a report of material covered according to the curriculum, 3) select the correct level of intervention and explanation, 4) provide operational help and 5) provide feedback by the student.

**The Tutor Module** is responsible for deciding what actions to take in order to teach or correct a specific domain based on the design of the curriculum. In the case studied, we have based our paper on the proposal made by Laureano-Cruces, Sánchez-Guerrero, Mora-Torres & Ramírez-Rodríguez (2008*)*, coupled with the classification of errors in sub-classes (Table 2 and Figure 1). The above is carried out in relation to the objectives of the planner[1] regarding one or several specific subjects to be taught. The planner selects the problems to be resolved by the student, analyzes the answers, presents the solution to certain problems or decides to show examples. It manages didactic material and is responsible for selecting the most suitable material depending on reported situations. These situations are mainly established by the demands of the planner and the behavior of the student as perceived through the interface.

**The interface** can be considered to be a simulated environment since it is where input and output from the system are represented. Its basic responsibility is to communicate between the system and the student, although the fact that it is the output for ILS activities means that it also has a didactic responsibility (Velasco-Santos, Laureano-Cruces, Mora-Torres & Sánchez-Guerrero, 2008).

### 2.1. Our Proposal

In this case, we have designed an ILS for subsequent implementation. Implementation will be encapsulated in a learning object. In order to achieve this, an inference engine was used based on a *general didactics tutor*. This tutor is linked to the tutor module of the case studied and to the behavior perceived by the user (student model). The behavior of the user is represented by the student model. A relationship is established using the relationship between the tutor module and the student model which allows us to prepare different didactic strategies.

The general educational model used in this execution will take into consideration the nine elements proposed by Laureano-Cruces, Ramírez-Rodríguez, Mora-Torres & Escarela-Pérez (2008). Said model is inspired by human behavior. These elements are linked through causalities, which allow them to take into consideration the influence of each one on the rest. The form of representation selected in order to bring to life this inference engine is a technique known as fuzzy cognitive maps (FCM) and belongs to the area of cognitive engineering and heuristics methods. For more details, consult (Laureano-Cruces, Ramírez-Rodríguez, Mora-Torres & Escarela-Pérez, 2008).

The ILS's establish the learning process as the cooperation between an intelligent system and a human being. Based on the evaluation of the user's performance, the tutor is *constantly making decisions* in order to select the most

---

[1] The planner is understood to be the place where selection control methods and sequencing of teaching strategies are located. It can be implemented as a separate component or as part of the tutor module depending on the selected control mechanism.

appropriate teaching strategy. These strategies will be prepared on the basis of the perception of the performance of the user, using as evidence a series of parameters including: errors made, learning style, command of knowledge, emotional-motivational state, etc. These assessments shall establish: what to explain, the level of detail and at what time, as well as: when to interrupt the student and what information to provide during the interaction.

## 3. Cognitive Didactics of the ILS

In these systems, cognitive didactics are designed in accordance with instructional objectives (IO) that represent the sub-skills and cognitive abilities that the teacher (in this case the tutor module) wants to transmit to the student (Laureano et al., 2000), (Laureano-Cruces Terán-Gilmore, de Arriaga, El Alami, 2003), (Laureano-Cruces Terán-Gilmore, de Arriaga, 2004), (Laureano-Cruces, Terán-Gilmore, & Rodríguez-Aguilar, 2005). They are activated together with operative didactics.

### 3.1. Instructional Objectives

In this section, we will demonstrate the relationship that exists between each concept, or skill and the instructional objective (IO). Where an IO is defined by the cognitive skills and abilities that the tutor wishes to transmit to the student. IO's may be classified in accordance with the following taxonomy (Bloom, 1956): 1) knowledge, 2) understanding, 3) application, 4) analysis, 5) synthesis, 6) evaluation.

In the context of the domain in question and according to Gutierrez (1994), the first three will be used and emphasis will be placed on the fact that the analysis implies application, which in turn implies understanding, which in turn includes knowledge. Below we will define and link them to our domain:

**Knowledge**: This objective is linked to the acquiring of knowledge by the student.

The student needs to know the theoretical concepts divided into definitions of: 1) the different types of data, 2) control structures (sequence, iteration (conditional, non-conditional)), 3) selection (simple, multiple), 4) types of abstractions (procedural and functional) and 5) types of parameters (reference and value).

**Understanding:** be able to handle the detailed logic of each abstraction carried out. Be able to understand the control structures and their meaning in order to transfer the concept of each one to new concepts which allow them to be interpreted and compared.

**Application:** this objective is linked to the implementation of previously learned knowledge. According to (Laureano-Cruces et al. 2003), the student will correctly apply the procedure in order to be able to:

1) Understand the use of the different types of data according to the characteristics of the problem.
2) Understand the type of control structure, *ad-hoc* to the objective of the module being structured.
3) Understand the use of the different types of abstraction provided by the paradigm of the structured programming.
    a. Passage of parameters and their types.
    b. Types of abstraction: functional and procedural.

**Analysis:** is used to teach the student to understand the states of different settings and to analyze them using different types of reasoning (Laureano-Cruces et al. 2004), which involve:

*Prediction:* action or effect of announcing what will happen based on incomplete information in a possible future.

*Post-diction:* explanation of how we have arrived at a current concrete situation, equivalent to a retroactive prediction, which is why it is considered to be a non-deductive type of reasoning, also known as abductive logic. This type of reasoning is made up of two phases: the *first* is a set of possible explanations and the *second* is the construction of explanations and the selection of the best.

*Quantitative interpretation:* given the partial description of the peculiarities of a situation and observations of its behavior, conclude that other peculiarities exist and that more may occur.

*Causal reasoning:* This type of reasoning is a tool that gives credit to a hypothesis originating from an observed or postulated behavior. It is useful for the generation of: explanations, interpretation of measures, planning of experiments and, obviously, learning.

The student analyzes a specific state of the scenario and should be capable of understanding what is happening based on the values of the different parameters and their relationships with the other elements.

Once the instructional objectives have been established, the instructional plan needs to be refined in terms of the activities that need to be carried out both by the intelligent learning system, as well as the student. These activities will be known as *instructional strategies* and shall be responsible for: providing the student with exercises, for motivating the student, for sending system communications to the student (by means of explanations, comments, a graphic example, etc.), and for providing continuity to the instructional session.

Scenarios are constructed based on: the conceptual graph of the *structured programming* domain and *the proposed instructional objectives* in (Laureano-Cruces, Sánchez-Guerrero, Mora-Torres, Ramírez-Rodríguez, 2008). When constructing proposed scenarios, the academic experiences of co-writers were used.

Each scenario has evaluation parameters which are defined in the following section (3.2). These measures are necessary in order to establish: how to explain, the level of detail and at what time and how to interrupt the student. It is a process that also includes some emotional-motivational aspects (operative strategies) in order to maintain and control the continuous execution of tasks and activities required for the study.

### 3.2. Elements of the Student Model

The elements making up the student model are the nine elements of the general didactic tutor (Laureano-Cruces, Ramírez-Rodríguez, Mora-Torres & Escarela-Pérez, 2008), in addition to learning style and study objective (whether it is internal or external). They are as follows: 1) interest, 2) desire, 3) help, 4) cognitive and operative strategies, 5) interruption, 6) quit, 7) learning, 8) idle time, 9) error, 10) learning style, 11) motivation: internal or external.

Below we will structure the different elements that are assessed according to the different agents and their relationship with the errors that occur in the case studied.

## 4. Student Model

Elements to be taken into consideration (Reilly and Lewis, 1991) in order to assess the performance of a human teacher, as well as to establish the way in which the ILS *tutor module* works, include student motivation, materials available, instructional objectives, student skills and administrative or support capabilities of the community.

Furthermore, the design of an ILS needs to be sufficiently flexible in order to adapt to different needs and different students, as well as to provide a mechanism that makes students aware of its improvements. As a result, the nine elements in Table 1 are essential in order to simulate the teaching-learning process (Laureano-Cruces et al., 2000; Laureano-Cruces, Ramírez-Rodríguez, Mora-Torres & Escarela-Pérez, 2008) in an ILS.

| Elements | Description | Linked to |
|---|---|---|
| Interest | Interest in the subject of the proposed task. | Motivation, profile. |
| Desire | Desire to continue performing the proposed task. | Motivation, profile. |
| Help | Possibility of requesting help to carry out the proposed task. | Confidence in the teacher and the environment. |
| Cog/op strategies | Have cog/op strategies. | Motivation of the teacher. |
| Interruption | Need to interrupt the proposed task. | Ability to organize, to use cognitive and / or operative strategies, to undertake a constructive debate, to ask for time. |
| Abandonment | Possibility of leaving the system without completing the proposed task. | Latent possibility of the student who has no interest and / or previous knowledge. |
| Learning | Complete the proposed task; learn to carry out the proposed task by developing the necessary skills and abilities. | Latent possibility which is always aimed for and which is encouraged based on different strategies. |
| Inactive times | Possibility of not carrying out any tasks during prolonged periods of time. | Lack of understanding or interest in the example or concept due to exhaustion or frustration. |
| Errors | Possible errors while carrying out the proposed task. | Level has not been achieved regarding the handling of different skills. Due to distraction, lack of interest, exhaustion. |

**Table 1. Elements considered in the teaching-learning process.**

The interpretation of each of the elements involved in the teaching-learning process and their relationship with the other elements is detailed in Laureano-Cruces, Ramírez-Rodríguez, Mora-Torres & Escarela-Pérez (2008).

Based on the previous data, different teaching-learning strategies are developed. These are linked to the different types of errors handled by different agents. The latter represent micro-worlds of expertise in some of the instructional objectives.

We have a mental model which generically requires the use of different control structures. This point is extremely important and is located in the detailed logic stage of each of the modules (descending modular programming and structured programming). Taking this into consideration and based on the instructional objectives (defined in section 2.1), we have three agents: 1) sequence, 2) iteration: conditional (in the meantime and repeat) and non-conditional (arithmetic progression), and 3) selection (simple: IF_THEN_ELSE and multiple: CASE) as shown in Figure 1.

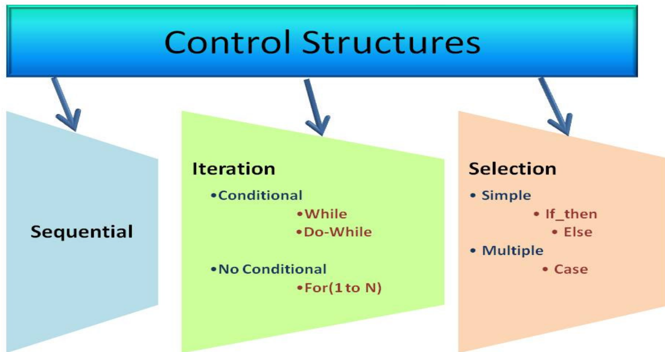


**Figure 1. Domain of agents for sub-classes.**

## 4.1. Errors

A classification has been developed for errors based on the experience of the writers. They are classified as: serious, minor, superficial and fatal based on the multi-agent architecture with dynamic intervention proposed by Laureano-Cruces et al. (2000) and Laureano-Cruces (2000). A specialist classification is made with the specialists being reactive agents that come into play as soon as a mistake is made in their area of expertise. Said classification is shown in Table 2.

***Serious Errors (S)*** are the result of an important lack of conceptualization that leads to the failure of the application of the control structures.

***Minor Errors (M)*** are the result of a lack of global attention rather than a lack of knowledge, i.e. they have the specific knowledge and may even have used it before but, due to a lack of attention, they get confused and do not complete part of the process.

***Fatal Errors (F)*** are the result of a complete lack of knowledge when it is considered impossible to continue. Remember that this is a training tutor and some initial knowledge of the resolution method, observed in class, is assumed.

| Sub Tutors | Control Structures | Control Structure Sequence | Control Structure Repetition | Control Structure Selection |
|---|---|---|---|---|
| Error 1 | Does not detect its use and needs to apply them **S** | Does not detect its use **S** | Does not detect its use **S** | Does not detect its use **S** |
| Error 2 | Confuses the structures **S** | Does not detect that it is not a sequence of instructions **S** | Confuses its use with the selection structure **S** | Confuses its use with a iteration structure **S** |
| Error 3 | Does not know in what case repetition and selection control structures are applied **S** | | Does not know how to use the structure condition **S** | Does not know the use of the structure condition **S** |
| Error 4 | | Does not know how to use the scope (beginning and end) of the structure **S** | Does not know how to use the scope (beginning and end) of the structure **S** | Does not know how to use the scope (beginning and end) of the structure **S** |

| | | | | |
|---|---|---|---|---|
| Error 5 | | Does not correctly define the type of variables of the structures **M** | Does not correctly define the type of variables of the structures **M** | Does not correctly define the type of variables of the structures **M** |
| Error 6 | | Does not correctly use the beginning of the instruction **M** | Does not understand the difference between asking for the condition at the beginning or at the end of the instruction **M** | Does not correctly use the beginning of the instruction **M** |
| Error 7 | | | Does not correctly use the decrement in the structure **M** | Does not correctly use the decrement in the structure **M** |
| Error 8 | | | Does not correctly define the use of the operators in the condition of the structure **F** | Does not correctly define the use of the operators in the condition of the structure **F** |

**Table 2. Critical errors for controlling the intervention of SUB-TUTORS (Obtained from the global development of the TC)**

## 5. Scenarios for the Case Studied

When evaluating a scenario, the value of each of the eleven elements that participate in the teaching-learning process (Section 4) is established. The detection of each element may be direct or indirect. Examples of the first are a request for help from the student, the learning style or objective through a questionnaire. As far as the indirect form is concerned, this is carried out by inference through causality relationships. The following scenarios are representative. In these examples, we aim to assess the types of assignment operators, arithmetic, relational and logic operators.

**Scenario 1:** The student will establish the execution order of the sentence, e.g. $a \leftarrow -2$; $q \leftarrow 4$; and $g \leftarrow 12q+(a*18 +14 - 12) / ( 22 * 3+ 4/2) *12$

What is the first operation executed? ______________
What is the second operation? ______________
Are operations established from left to right or from right to left? ________________________
So $g$ is equal to ______________

In this scenario, the student initially showed an interest in the task and a desire to continue. Furthermore, he/she requested help regarding the subject. Nevertheless, idle time is detected and there was a single error, the calculation of $g$. The above means that, in accordance with the teaching-learning process and the applying the causality matrix, we have the following entry vector *Vi* and the final vector *Vf*:

| | Interest | Desire | Help | Cog/op strategy | Interrup-tion | Quit | Learning | Idle time | Error |
|---|---|---|---|---|---|---|---|---|---|
| **$V_i$** | **1** | **1** | **1** | 0 | 0 | 0 | 0 | **1** | **1** |
| $V_1$ | 0.000045 | 0.000045 | 0.500000 | 1.000000 | 0.993307 | 0.993307 | 0.993307 | 0.500000 | 0.500000 |
| $V_2$ | 0.000572 | 0.000572 | 0.993078 | 1.000000 | 0.924110 | 0.924110 | 0.000591 | 0.993078 | 0.993078 |
| $V_3$ | 0.000000 | 0.000000 | 0.999999 | 1.000000 | 0.999999 | 0.999999 | 0.000101 | 0.999999 | 0.999999 |
| $V_4$ | 0.000000 | 0.000000 | 1.000000 | 1.000000 | 1.000000 | 1.000000 | 0.000045 | 1.000000 | 1.000000 |
| $V_5$ | 0.000000 | 0.000000 | 1.000000 | 1.000000 | 1.000000 | 1.000000 | 0.000045 | 1.000000 | 1.000000 |
| **$V_f$** | 0.000000 | 0.000000 | **1.000000** | **1.000000** | **1.000000** | **1.000000** | 0.000045 | **1.000000** | **1.000000** |

The final vector indicates that even when requesting help to solve the exercise, the student made mistakes and also had inactive times. As a result, an interruption is required by the tutor, which generates an increased possibility of abandonment. Therefore it is inferred that interest has been lost, as well as a desire to continue. Learning does not occur and, for this reason, a cognitive/operative strategy urgently needs to be applied.

**Scenario 2:** Taking as a basis the same type of exercise, the entry vector is *Vi*, almost the same vector as that presented in scenario 1, but unlike that vector, in scenario 2 a cognitive/operative strategy is applied in accordance with the learning style obtained when the user enter to the system. By applying the causality matrix for the teaching-learning process, we obtain the following final vector *Vf:*

| | Interest | Desire | Help | Cog/op strategy | Interrup-tion | Quit | Learning | Idle time | Error |
|---|---|---|---|---|---|---|---|---|---|
| **$V_i$** | **1** | **1** | **1** | **1** | 0 | 0 | 0 | **1** | **1** |
| $V_1$ | 0.006693 | 0.006693 | 0.006693 | 1.000000 | 0.500000 | 0.500000 | 0.999955 | 0.006693 | 0.006693 |
| $V_2$ | 0.992846 | 0.992846 | 0.006694 | 0.993943 | 0.000572 | 0.000572 | 0.508365 | 0.006694 | 0.006694 |
| $V_3$ | 0.999996 | 0.999996 | 0.000000 | 0.526508 | 0.000000 | 0.000000 | 1.000000 | 0.000000 | 0.000000 |
| $V_4$ | 0.999997 | 0.999997 | 0.000000 | 0.500000 | 0.000000 | 0.000000 | 0.999997 | 0.000000 | 0.000000 |
| $V_5$ | 0.999996 | 0.999996 | 0.000000 | 0.500000 | 0.000000 | 0.000000 | 0.999996 | 0.000000 | 0.000000 |
| **$V_f$** | **0.999996** | **0.999996** | 0.000000 | **0.500000** | 0.000000 | 0.000000 | **0.999996** | 0.000000 | 0.000000 |

The final vector indicates that, by including strategies, both interest and desire to continue, remain. And the possibility of abandonment, as well as inactive times and/or errors disappears. Furthermore, something very important is inferred: learning occurs this time.

In this section we have provided examples of two scenarios classified and detailed with the elements of the previous section and their possible didactic strategies.

The causal matrix of the general didactic tutor supports the evaluation of the development of the teaching-learning process, because they are part of the eleven elements of the student model. This allows us to prepare different educational tactics which bring to life the tutorial intervention.

In both examples, the use of Cog/Op Strategies is recommended. And in both cases the evaluation of the nine elements on Table 1 (section 2.3) is enriched by the learning style and the study motivation, as well as the type of error (agent will handle it). By integrating the above information, the dynamic, personalized didactic strategy is prepared.

## 6. Learning Objects

The Learning Objects (LO) Model provides a new way to organize content in a compositional hierarchy of granularity ranging from multimedia objectives, to informative objectives and learning objects, to more complex conglomerates of educational objectives such as sections, units, courses, study programs, etc. It provides a way of constructing educational content according to composition based on pieces of elements on lower levels. Likewise, it provides a way to find content objects, to locate them, recover them and integrate them via a collection of specifications and standards for their cataloging, requisition, exporting, transporting and importing. Finally, it provides each student with the opportunity to construct a personalized selection of educational content which provides an optimum context for their learning.

### 6.1. Design of Learning Objects

In order to design Learning Objects (LO), we will adopt the work methodology proposed by Muñoz-Arteaga, Osorio-Urrutia, Álvarez-Rodríguez y Cardona-Salas (2008) described in (Laureano-Cruces, Sánchez-Guerrero, Mora-Torres & Ramírez-Rodríguez, 2008).

The Analysis and Obtaining phase of the LO is where general data is identified, as well as the composition of didactic material for its construction. This is closely linked to instructional objectives which, in turn, are linked to the conceptual graph.

The ILS, defined throughout this paper, is encapsulated in the design phase and involves the identification of the objective, which in the studied case is: 1) personalized instruction, and 2) structured programming, among others.

### 6.2. Generation of Metadata

In order to generate metadata, the LO needs to be described in order to facilitate the management of cataloging, search and recovery. In order to generate this metadata, we will use the SCORM LO design standard. The SCORM metadata is based on TSC standard IEEE1484-12-12-1-2002L * Learning Object Meta-Data (LOM), which specifies the general characteristics of the metadata of an LO.

The assembly, packaging and storage of an LO in the repository takes place during the *Development* phase. In order to assemble an LO, we need to integrate each component in an XML template which contains general information relating to the LO. The packaging needs to be generated using the SCOR standard in order to be able to create and edit the metadata. The LO needs to be stored in an LO repository.

Finally, the Integration phase of the LO into an Administration System is required in order to efficiently manage LO's. The *Administration System* supports the integration of all activities such as: online courses, chats, exercises, exams, etc. The above can also be part of a collaborative environment.

## 7. Conclusions

In this project, the implementation of the inference engine based on a general didactic tutor will permit: 1) dynamic interaction regarding the decision-making process in order to select the best instructional strategy, 2) forecasting of the possible future state which allows us to personalize interactions through the eleven elements that constitute the student model, and 3) will allow us to carry out specialized processing of errors using reactive agents.

A methodology of analysis and design has been created which can be used to create other LO's that use *personalized teaching* by means of the model developed for the *general didactic tutor*. Furthermore, the fact that the LO's have development and access standards will allow for their increased use.

This system will support and benefit the *Individualized Learning System* of the Universidad Autónoma Metropolitana Azcapotzalco, campus by generating LO's that can be used to support students and professors. This can be achieved if these learning objects are designed using the SCORM (*Sharable Content Object Reference Model*) standard. Furthermore, it will allow content from different platforms located in other universities or organizations that share the same application domain to be shared. As a result, this will maximize the use of new technologies such as *e-learning* using learning objects and artificial intelligence techniques.

**Acknowledgments:**

This paper is part of the research being carried out by Lourdes Sánchez-Guerrero to obtain her Master's Degree in Computer Sciences at the Universidad Autónoma Metropolitana-Azcapotzalco. It is also part of the project: *E-Learning Knowledge: entornos virtuales para el aprovechamiento compartido del conocimiento. Modelo de generación de células (2007)* (E-Learning Knowledge: virtual environments for the shared use of knowledge. Model for the generation of cells (2007)), funded by same University.

## References

1. Bloom, B.S. (Ed.) (1956). *Taxonomy of educational objectives, the classification of educational goals - Handbook I: Cognitive Domain*. New York: McKay.

2. Gutiérrez, J. (1994). *INTZA: un Sistema Tutor Inteligente para Entrenamiento en Entornos Industriales*. Tesis Doctoral (Tercer Ciclo) de la Universidad del País Vasco, San Sebastián.

3. Laureano-Cruces, A (2000). *Interacción Dinámica en Sistemas de Enseñanza Inteligentes.* Tesis Doctoral. Instituto de Investigaciones Biomédicas/UNAM. http://delfosis.uam.mx/~ana/.

4. Laureano-Cruces, A., & de Arriaga, F., (2000). Reactive Agent Design for Intelligent Tutoring Systems. *Cybernetics and Systems* (an International Journal). 31(1), 1-47.

5. Laureano-Cruces, A., Terán-Gilmore, A. de Arriaga, F. y Alami, E. (2003). La Importancia De Las Estrategias Cognitivas en el Diseño del Curricula Didáctico. Vol. I, pp. 35–41. *XVI Congreso Nacional y II Congreso Internacional de Informática y Computación de la ANIEI*. 22-24 de octubre,Zacatecas http://delfosis.uam.mx/~ana/

6. Laureano-Cruces, A., Terán-Gilmore, A., J. & de Arriaga, F., (2004). A Learning Model Based on a Didactic Cognitive Approach: The Case Of Single-Degree-Of-Freedom Systems. *Computer Applications in Engineering Education*, 12(3), 152-164.

7. Laureano-Cruces, A., Teran-Gilmore, A. & Rodriguez-Aguilar, R.M. (2005). Cognitive and Affective Interaction in a Pedagogical Agent. *Memorias en CD, ISBN: 970-31-0528-9. En el XVIII Congreso Nacional y IV Congreso Internacional de Informática y Computación de la ANIEI*, 26-28 de octubre, Torreón, Coah., http://delfosis.uam.mx/~ana/ http://delfosis.uam.mx/~ana/

8. Laureano-Cruces, A., Ramírez-Rodríguez, J., Mora-Torres, M., de Arriaga, F. & Escarela-Pérez, R. (2008). Cognitive-Operative Model of Intelligent Learning Systems Behavior. *Interactive Learning Environments. ISSN: 0888-613X, Elsevier. UK. Publicado en Línea.* http://www.informaworld.com/smpp/content~content=a795273447~db=all~order=pubdate.

9. Laureano-Cruces, A., Sánchez-Guerrero, L., Mora-Torres, M. & Ramírez-Rodríguez, J. (2008). Learning Objects and Personalized Instruction. In G. Richards (Ed.), *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2008* (pp. 1728-1736). Chesapeake, VA: AACE. ISBN: 1-880094-66-5.

10. Velasco-Santos, P., Laureano-Cruces, A. Mora-Torres, M. y Sánchez-Guerrero, L. (2008). La Importancia del Diseño de una Interfaz en el Proceso de Enseñanza-Aprendizaje. *Memorias en CD, ISBN 978-970-15-14388-2, pp. 108-113. En el XXI Congreso Nacional y VII Congreso Internacional de Informática y Computación de la ANIEI.* 1-3 de octubre, monterrey. http://delfosis.uam.mx/~ana/.