# A Way to Learn How to Design Educational Software

**Dalia M. Gil**
Department of Electrical Engineering and Computer Engineering
University of Central Florida
P.O. Box 162450. Room EN407, FL 32816, USA
dgil@mail.ucf.edu

**Miguel A. Garay**
Department of Computer Science
Havana Polytechnic Institute
127 St., Marianao, Havana, Cuba

## ABSTRACT

The improvement in the quality of instructional process stimulates the development of active teaching methods based on “learning by doing” principles. This paper shows an experience developed on educational informatics course to engineering instructors students who learn to create a type of hypermedia intelligent tutoring system with a system approach. This experience has been developed for more of six years at Havana Polytechnic Institute.

**Keywords:** Educational Technology, Educational Software, and Teacher Education.

## 1. INTRODUCTION

The improvement in the quality of instructional process stimulates the development of active teaching methods based on “learning by doing” principles. This idea has been very effective in all teaching levels because the students can develop skills faster when they are able to solve by themselves different situations associated with the real world. For example, the acquisition of knowledge about physical phenomena by engineering students constitutes a complex and slow learning process. Students find it most difficult to identify a problem because they need to understand the verbal model (word problem) of reality. This phenomenon is evident when students need to understand the process of identification and representation of physical phenomena by means of mathematical model. This task has not been solved yet because it is impossible to provide the students with a checklist for selecting the corresponding interpretation of reality in different contexts. However, the authors considered that there were different ways to attenuate this problem. One of them was to explore the possibilities of technology and create teaching means that helped the teaching-learning process as a part of a pedagogical research. One interesting experience was to develop teaching means that have been used in a postgraduate educational informatics course to engineering instructor at Havana Polytechnic Institute for more of six years. Graduate students have learned how to design and create educational software applied to different subjects such as linear programming, mathematics, physics, chemistry, biology, and so forth. Some of these results are showed in the present paper.

## 2. RESEARCH RESULT

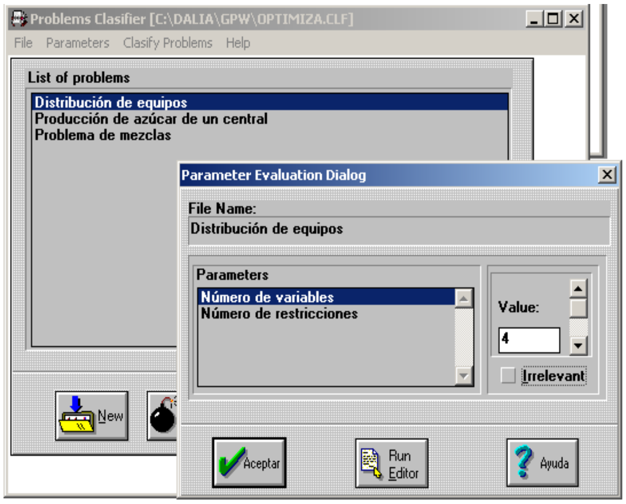
This research began when the authors decided to enhance mathematical modeling teaching using teaching means. The first step was to analyze several available software resources at the institute such as authoring systems, shells, programming languages, and linear programming applications developed by other authors. The goal was to know whether there is software that could be used as a mathematical modeling teaching means. As a result of this analysis, no applications were oriented to identify a problem of linear programming; neither recommended the most rational solution method of the problem. Because the identification of a problem is a process that starts from a set of data that appears on the word problem, and the student must work toward a conclusion, this mechanism can be considered analogous to forward reasoning. Then the decision was to develop a first version of an inference mechanism with this reasoning and a simple structure such as a binary tree representation. Beside that, it was explored how a multimedia environment could reduce the difference between the real problem and the word problem. The hypermedia representation was selected because it combines resources such video, photos, graphics, animations, sound, and text. Another important task was to find a classification method to organize into a bank of problems (data base) the word problems by means of different levels of complexity.

Based on the ideas stated above, the research group decided to create its own software tools focused on generating hypermedia intelligent tutoring systems that developed the student’s skills related to classification and identification of problems, this was chosen because this kind of activity is quite common in the engineering field. The tool was called Hypermedia Intelligent Tutoring Generator (HITG). This tool does not replace instructors; it merely generates tutors that help students improve the quality of their preparation or simply solves problems when the professor is not available. It was designed with a system approach that integrates different modules as a puzzle. The modules are: Hypermedia Problem Generator, Expert, Solver, Student, and Tutor modules. The entire tool is able to generate a specific kind of a tutoring system, and each module can be used as an independent tool. These modules are explained in the following sections.

### Hypermedia Problem Generator

The improvement of teaching process under given conditions must strengthen the relationship between reality and the subject being taught. It must guarantee the development of modeling skills, first in translating from reality into the verbal model, and second from the verbal model to the correct solution. These skills must be associated with their capabilities to apply the knowledge learned about the solution of the real problem, establish the main analogies and differences among different problem situations, and select the best or most rational solution. In relation to this, the selection of a bank of word problems with "different" difficulty levels was one of tasks. Problem classification is always subjective, but in all cases, it could be considered as the action, which sets into one class or level those problems that look alike. Different classification methods were analyzed. A statistical method developed by a Russian scientist Maxrov [4] was selected. This method classifies the objects (word problems) in different intervals (three or more), and it had been used with success to classify norms by the authors [2]. As a result, a software tool called Problem Classifier was programmed. This program employs a graphic environment (see figure 1). The user can type a name of each problem, and select how many complexity levels will be considered. Parameters and their weight can also be typed. Each parameter will receive a value, or it can be considered irrelevant (no value). Data can be modified at any time. After the information is typed, the system classifies and gives a report in a file .txt; another option of the classifier program is to access the hypermedia representation of each word problem to be analyzed.

After each word problem has been classified, the next step is to obtain its representation in a hypermedia format. This process can be developed by means commercial authoring systems that generate hypermedia applications. However, our research group decided to develop our own tool. As a result, a hypermedia generator system was created [9]. This software has two modules. The first module, called Editor has a window menu with different options to create a hypermedia application. The second module called Generator is used to run the application developed which format is a file .hlp. Before an application can be developed, it is necessary to create multimedia resources associated with it (video, photos, graphics, animations, sound, and text). These resources are generated by means different authoring systems available at the computer laboratory because our tool is compatible with the most resource file known. The hypermedia representation of each word problem permits the students to "touch" with their own hands the real word before trying to solve the problem. Once the bank of word problems has been created, the Expert module will be designed.


Problems Clasifier [C:\DALIA\GPW\OPTIMIZA.CLF]

List of problems

Distribución de equipos
Producción de azúcar de un central
Problema de mezclas

Parameter Evaluation Dialog

File Name:
Distribución de equipos

Parameters

Número de variables
Número de restricciones

Value: 4

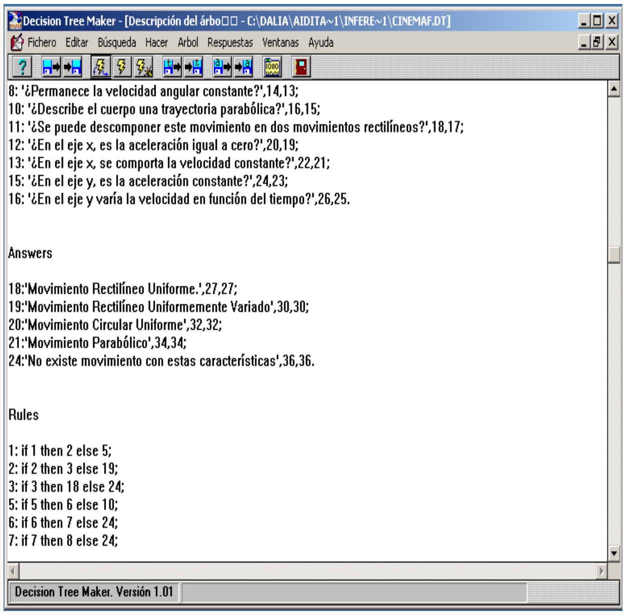
□ Irrelevant

Figure 1. Problem Classifier

### Expert

The expert module contains the knowledge about the specific subject of application. Drawing on this knowledge the system is able to answer the student's questions and give him/her guidance. This module uses a forward chaining mechanism for the inference engine. This mechanism responds to characteristics of the identification of the type problems selected. The proposed method provides questions about a problem and recommends the most rational solution for the given conditions. The knowledge base should be built through interactions with other experts. This body of knowledge is represented as a collection of production rules. Each application developed has its own knowledge base. This first version of the inference engine has a simple structure where the answers can be yes or no. However, the results have been very interesting. For example, an application developed to mathematical modeling teaching [9] provides questions about linear programming models (Simplex, transportation, and assignation), and it recommends the most rational solution for the given conditions. This knowledge base has 200 rules that identify problems to be solved by means Simplex methods, transportation, or assignment models. Figure 2 shows an example where appears questions about how to identify a physical movement. Some of questions are whether the velocity is constant, or whether the acceleration is zero. At the end, the inference mechanism guides the student to the conclusion about what is the mechanical movement associated with the problem analyzed.

Once the problem is identified and the solution method selected, the Solver module computes the solution.


Decision Tree Maker - [Descripción del árbo□□ - C:\DALIA\AIDITA~1\INFERE~1\CINEMAF.DT]
Fichero
Editar
Búsqueda
Hacer
Arbol
Respuestas
Ventanas
Ayuda
Decision Tree Maker. Versión 1.01

```
8: '¿Permanece la velocidad angular constante?',14,13;
10: '¿Describe el cuerpo una trayectoria parabólica?',16,15;
11: '¿Se puede descomponer este movimiento en dos movimientos rectilíneos?',18,17;
12: '¿En el eje x, es la aceleración igual a cero?',20,19;
13: '¿En el eje x, se comporta la velocidad constante?',22,21;
15: '¿En el eje y, es la aceleración constante?',24,23;
16: '¿En el eje y varía la velocidad en función del tiempo?',26,25.


Answers

18:'Movimiento Rectilíneo Uniforme.',27,27;
19:'Movimiento Rectilíneo Uniformemente Variado',30,30;
20:'Movimiento Circular Uniforme',32,32;
21:'Movimiento Parabólico',34,34;
24:'No existe movimiento con estas características',36,36.


Rules

1: if 1 then 2 else 5;
2: if 2 then 3 else 19;
3: if 3 then 18 else 24;
5: if 5 then 6 else 10;
6: if 6 then 7 else 24;
7: if 7 then 8 else 24;
```

Figure 2. Edition of knowledge base

## Solver

The modular conception of the tool permits the Solver module to be developed with any programming tools such as an authoring system or a programming language. Problem Solver works in correspondence with the Expert module, which determines the main characteristics of the formulated problem. Each word problem, a set of problem, or application, requires the corresponding solver module. This module can be less or more complex, it depends of the type of problem to solve. For example, the application about Linear Programming explained above, required its solver module had different algorithms such as Simplex, transportation, and assignment. That is, a student selects a problem (word problem with a hypermedia representation). If he/she has difficulty to identify what is the corresponding mathematical model, he/she invokes the Expert module, which helps the student to identify the mathematical model, then Solver is activated, and the student can enter data and obtain the solution. This application permits that student runs the algorithm many times. Besides that, a student has the option to see a graphical representation of linear problems with two and three variables.

## Student Model

The student module reflects the understanding of the subject matter by the student, his thinking and his strategies. This module gathers the student's behavior, and then the Tutor module uses this knowledge base to develop its pedagogical strategy. The information gathered is saved in a data file that can be accessed by instructor.

## Tutor

The tutor module contains the rules, strategies and processes that govern the way the system communicates with the students. It includes, for instance, solutions to questions about HOW to tutor, WHAT instructional tools to try, and WHY and HOW often to interrupt the student. Several possible strategies can be applied, ranging from rather intrusive tutoring (remarking immediately after mistake) to more liberal coaching. The tool allows the instructors to change the tutoring strategies according to their instructional goals. The solution of problems develops the student's logical thought. It demands a productive reasoning that permits him/her to be able to arrange his/her thoughts and generate adequate solutions. This reasoning implies an aggressive intellectual action. The pedagogical strategy of the tutor is base upon a productive reasoning. The selection of each problem is considered a function of student's learning. The student needs to feel that there is a real possibility to solve the problem.

The tool explained before has been used successfully in two important ways. The first has been to develop educational software related to mathematics, physic, chemistry, and biology applied to engineering teaching. The second has been as a teaching means in a postgraduate course to engineering instructors. The following section explains this last experience.

## 3. HOW THE TEACHING MEANS HAS BEEN USED

Furthermore for more than six years, our university has developed a Master of Science in Informatics. This program covers different modules such as software engineering, digital graphics, computer aided design, artificial intelligence, decision-making and educational technology. This last module is approaches to instructors or teachers from different educational levels that are interested in how to apply educational technology in the teaching learning process. Educational technology module integrates with an approach system a set of courses based on pedagogical and technologic criteria that permit active learning practices, and help students to face a variety of higher order challenges. The principle that supports the educational activities is “learning by doing”. As a result, students have different important assignments; one of them is a project to design and create a hypermedia intelligent tutoring system using the Hypermedia Intelligent Tutoring Generator (HITG) explained above and others. This project is carried out along of Intelligent Computer Aided Instruction course.

The objective of this project is to develop among students skills related to the design and implementation of educational software. In this case, they develop a hypermedia intelligent tutoring system. This project has demonstrated be learning participative environment where students enrolled can bring a wealth of teaching experience each to others offering many opportunities for faculty-student and student-to- student interaction. Thus students have opportunity to ask questions about and to criticize proposed approaches and answers. In addition, they are pressed to consider the validity of their evidences and reasoning identifying what one does and does not know.

The tutoring systems generated have the following characteristics:

1. Development of student’s skills in modeling by means of the isolation of subsystem (analysis) and its later integration (synthesis).
2. General and simple situations are introduced at the beginning. The level of complexity will be gradually increased later as a function of the student’s performance.
3. In all cases, the problems transition from static or simple situations to more complex ones. There are several models that represent the behavior of the problems situation under study.
4. All word problems have a friendly and exciting hypermedia format that combine different multimedia resources.
5. The teaching strategy in each tutor is a function of the student’s performance (student model).
6. Each application has an expert module with a knowledge base represented by production rules.

The project has eight stages that are carried out along the course. They are:

1. Selection of student’s subject and the proposal is introduced.
2. Selection and classification of a set of word problems.
3. Design and creation of a hypermedia format per word problem.
4. Design of the knowledge base.
5. Develop of the solver module.
6. Integration and compilation of all modules as a whole application.
7. Written and defense of the project.

**Selection of student’s subject and the proposal is introduced**

Students select a subject and topics that will be included in the software. These topics correspond with the class where they are instructors because they are considered experts in this field. Students must write a document introducing the theme, define its goals, and draw the initial ideas about this application, structure, behavior, and how they think the educational software could be applied in their classes. At the end of this stage, each proposal is analyzed by a group of students and the professor, who take this exercise as a brainstorm where interesting opinions are given. At this point, the experience of each student as an instructor plays an important role. The application of technology to teaching learning is actually a creative process if pedagogical knowledge and teacher’s expertise are considered.

Once each project has been approved, the task of developing the hypermedia intelligent tutoring system, is divided in different stages, where each is associated with and specific module of the program. Remember that the application will be created using the tool called Hypermedia Intelligent Tutor Generator (HITG) that has five important components: Hypermedia Problem Generator, Expert, Solver, Student, and Tutor modules.

**Selection and classification of a set of word problems**

Students should select no more than five exercises to problems because this is only a project and the goal is to create abilities in developing educational software of this type. The next step is to classify these problems in function of three levels of complexity (they could be more). The criteria of classification and the weight of parameters is a complex task because of the subjective factors among others. For example, how is the students who will use this system, the type of problem to solve, the goals, subjects, relationships with previous concepts, variables and parameters, type of formulas, presentation of data, the amount of effort to solve it, could be some of criteria. However, the experience has demonstrated that the same problem can be classified using different criteria. The level of complexity is a function of the context where it is analyzed and the viewpoint’s analyzer (expertise). Once the criteria and weights are defined, students use the Problem Classifier tool [2] explained above.

### Design and creation of a hypermedia format per word problem

Once the problems have been classified, students who have learned existing ways of organizing and presenting data and information should generate the corresponding word problems in hypermedia format. The translation from reality to a word problem with hypermedia format is called information interaction design interactive, which involves different stages such as information design, interaction design and sensorial design. In the first stage, students organize the structure and subject associated to each word problem. Because the second stage is story creating and telling, students should solve how the word problem is telling interactive and attractive way. The last stage, sensorial design involves the creation and presentation of media. To do this, students can select our Hypermedia Generation System or any other hypermedia generator, which file has the extension .hlp. Before students create different multimedia resources by means of authoring systems and other software tools explained in class. This stage is very creative because each student selects what multimedia resources will appear in each word problem that will be represented by a hypermedia format by means of video, photos, graphics, animations or sound. This representation of the word problem will permit the end users (students) can *"touch"* with their self own-hands the reality before trying to solve the problem.

### Design of the knowledge base

The next stage is associated with the intelligent part of the system. Here students should create a knowledge base about the specific subject defined at the beginning of the project. The tool used in this case is the Tree System [10] developed in our research group. The structure of the base is a binary tree represented as a collection of production rules, and the inference engine uses a forward chaining mechanism. This mechanism identifies the type of problem and recommends the best or most rational solution for the given conditions. Students must formalize the knowledge using this representation. This stage is very difficult because they must create a valid structure tree to analyze each of problems proposed. In the tree generated each node corresponds to a question and each leaf will be an answer. The questions must be approached to identify the types of problem or cases existing in the bank of problems (data base). The knowledge base should be built through interactions with other experts, the information retrieved from different sources such as books and the graduate student's own experience. Remember, they as instructors are "experts" in the subject selected. The idea is the student's answers guide him/her to a solution that could be correct or not. In each case, the system must be an adviser.

Figure 3 shows an example of a binary tree developed with thirteen nodes where the student should identify a type of a physic movement such as a uniform linear movement. Each node is a question, and each leaf is a conclusion or answer. Figure 2 shows the questions and answers that belong this binary tree.

### Develop of the solver module

Once the problem is identified and the solution method selected, the next step is to create the solver module. In this stage, each word problem needs its own solver. The complexity of this module is a function of the type of problem to solve. The module could be simple with questions such as "select the correct answer among different options", "true or false", or more complex such as "solution of mathematical equations" and "solution of a specific method". Some students select an initial problem, which is modified with new restrictions creating a set of cases that could be solved with the same solver module but considering different situations.

In this stage, some students show more computational abilities than others, but in all cases the result is positive because they must design and program a solution method per each word problem using any programming tools such as an authoring system or a programming language. This module likes others can run alone or joined to the hypermedia intelligent tutoring system developed because it works as an executable program (.exe).

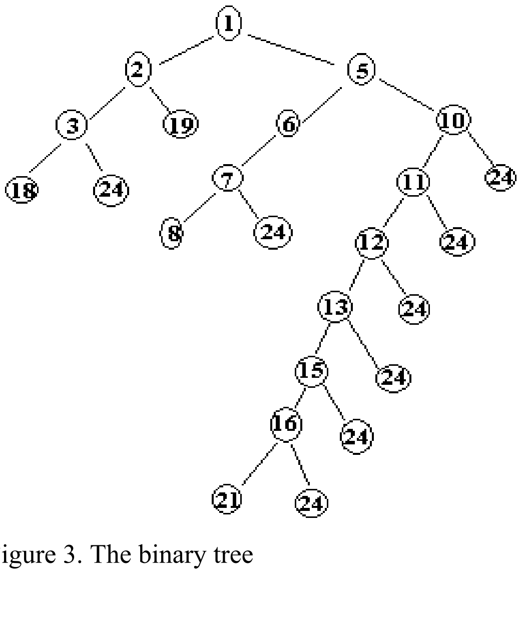


Figure 3. The binary tree

**Integration and compilation of all modules as a whole application**

The last step is to integrate all modules in one application to obtain this result; there is a tool that belongs to Hypermedia Intelligent Tutor Generator (HITG), which does all links among the different modules programmed. Although the application developed was only a sample, it can be considered a good learning practice to understand what could be educational software. The students developed important skills related with the design of a system and its parts, design and representation of a knowledge base, design and develop of multimedia resources, and programming of software. This last element is very important because the students who have passed this course are instructors of different classes not related to computer sciences. At the end, each student can be considered as a beginner developer of educational software who was able to create his/her first application.

**Written and defense of the term paper**

Once the project is ended, each student must write his/her own term paper that is presented together with the software. Both are defended in special session at the end of the course. The style of this assignment is presented in the general conception of the Informatics master program.

Because this class is part of the master program, some students motivated for this experience have used HITG as tool in their project of thesis such as [1], [5], [6], [7], [8], and [9]. One hundred forty one students have obtained their master degree in Informatics. From them fifty-eight have passed our educational informatics course. This amount represents the 41 percent.

The experience developed above has demonstrated how students learn well when they are engaged in active exploration, interpretation and construction of ideas. This approach facilities a succeed learning among students of postgraduate courses, and stimulate them to use informatics technologies as a resource to improve the teaching learning process.

.

## 4. CONCLUSION

The experience developed has permitted that students play an active role during the teaching learning process. The have learned a concrete way to apply the informatics technology combined with pedagogical criteria to design a kind of educational software such as a hypermedia intelligent tutoring system applied to different fields such as mathematics, physics, chemistry and biology.

This experience has demonstrated that a postgraduate course can be a “workshop” where students receive new information; learn from another’s experiences and their own experiences facing the problem solving. The class is not a process where students only listen to the professor “swallowing information” passively.

The pedagogical experience developed with Hypermedia Intelligent Tutoring Generator during these years has permitted to begin new researches related with this theme. Some of them associated to web environments with cooperative learning.

## 5. REFERENCES

[1] L. Chiang, L. and D. Garcia, “Tools for Development of Multimedia Intelligent Educational Software”, UNOPAR **Scientific**, Vol. 1, No.1- DEZ, University of North of Parana, Londrinas, Brazil, 1997, pp 87-99

[2] D. Garcia and L. Chiang, “Problems Classifier System ”, **Industrial Engineering Journal**, Vol. XVI, No. 3, Cuba, 1995, pp 134-139.

[3] M. Garay, D. Garcia, D, et al., “OPTIMA: A Hypermedia Intelligent Tutoring System for Mathematical Modeling Teaching”, **Proceedings of the World Conference on_Educational Multimedia and Hypermedia. ED_MEDIA 95**, Austria. July 1995.

[4] M. Garay, **Design of Automated Management System**, Havana, Cuba, Cuban Ministry of Higher Education. Pub. 1986.

[5] M. Gomez, and D. Garcia, “Hypermedia Intelligent Drill System for Electrochemistry”, **Proceedings of Congress Informatics’98**, Havana, Cuba, 1998.

[6] R. Ibanez and D. Garcia, “Tutoring _Drill System: Ionic Equilibrium in Chemistry”, **Proceedings of Fifth Congress of Computer Sciences**, San Juan, Argentina, 1998.

[7] L. Iriarte and D. Garcia, “Hematology Intelligent Tutoring“, Master Thesis in Educational Technology, Havana Polytechnic Institute, Havana, Cuba, 1998.

[8] Y. O’Farrill and D. Garcia, D. “Drill - Evaluator System Geometry Problems: SEPGE”, **Proceedings of XI Meeting about Educational Mathematics**,_Editorial Group IBEROAMERICA, S.A. de C. V. Mexico, 1997

[9] V. Rodriguez and D. Garcia, “Hypermedia Generation System”, **Proceedings of III Hispanic Educational Information Congress**, Barranquilla, Colombia, 1996.