Transactions on Information and Communications Technologies vol 19, © 1997 WIT Press, www.witpress.com, ISSN 1743-3517

An intelligent tutoring system for training biochemical engineers in mathematical modelling and simulation of food production using bacteria

P. Melin[a] & O. Castillo[b]
[a] *School of Engineering, CETYS University,*
EMail: emelin@mail.tij.cetys.mx
[b] *Dept. of Computer Science, Instituto Tecnologico de Tijuana,*
EMail: ocastillo@mail.tij.cetys.mx,
P.O. Box 4207, Chula Vista CA 91909, USA

Abstract

We describe in this paper a computer program that combines Artificial Intelligence (AI) techniques with Multimedia technology for the education of mathematical modelling and simulation of food production using bacteria growth. The computer program can be considered an Intelligent Tutoring System (ITS) for the domain of Mathematical modelling and Simulation (MMS) of Food production (FP). The ITS has the goal of training new engineers in the complex task of modelling food production and then performing the right simulations to obtain the dynamical behaviors of the biochemical system. Food production is the result of bacteria growth in a "bioreactor" that produces chemical compounds critical to the type of food that is desired. The computer program can then be used in the design of new bioreactors or in the control of the ones that are functioning, because the modelling of this systems enables the understanding of the underlying dynamics of food production. The computer program contains the knowledge of the human training experts in this area in the process of teaching MMS of FP. The knowledge base consists of a set of fuzzy rules for performing the mathematical modelling of a given biochemical system and also a set of rules to perform the simulation of the model. The ITS uses efficiently fuzzy logic techniques to evaluate the learning process of the students (engineers) of the domain. The ITS also uses multimedia to improve the efficiency of the learning process.

1 Introduction

We describe a computer program that can be considered an Intelligent Tutoring System, see McCalla [14], with a multimedia environment for the domain of Mathematical Modelling and Simulation of Food Production using Bacteria. The process of teaching this domain is difficult since it requires a background on Mathematics (Differential Equations and numerical Analysis) and

Biochemistry. In this ITS we are assuming that the people have the required prior knowledge mentioned above. When a new person uses the ITS, he or she enters an individualized learning process of the domain. The ITS maintains, for each student, all the relevant information about the learning status of the students. Also, the ITS monitors the actual learning process of the students by applying tests (periodically) and evaluating the responses by the students. the learning process, considered in this ITS, consists mainly of four steps:

1.- Acquiring the basic concepts of MMS of FPB.
2.- Learning the process of mathematical modelling for the domain.
3.- Learning the method for simulation of the models.
4.- Learning to apply MMS to complex food production problems and learning to design bioreactors with the information from the MMS.

The ITS has the responsibility of giving to the users the best possible information in each of the above steps, so that they can get the best possible understanding of MMS for FPB. Our prototype ITS uses Fuzzy Logic techniques, see Badiru [1], to improve the efficiency of the learning process described above and also to evaluate the level of understanding of the students of this domain. The use of Expert Systems technology and Fuzzy Logic enables the right use of the knowledge of the human experts for training in this domain.

The reasons why we consider that an ITS is needed for this domain are the following:

(1) There are not enough training experts in Mathematical Modelling and Simulation of Food Production using bacteria in the food manufacturing plants of Mexico.
(2) The domain of MMS for FPB is a complex one, because there are many different types of mathematical models that can be used for a given problem and also there are many possible dynamical behaviors for a particular biochemical system, ranging from simple periodic solutions to complicated "chaotic" solutions, see Devaney [10].
(3) There is a need of making more efficient the control of the food production process and this can be done by obtaining accurate simulations of the models and then by using this information to obtain the "best" control possible.

We believe that an ITS like ours can help solve all of this problems. First of all, the ITS has the necessary knowledge to simulate training experts for this domain. Second of all, the ITS has the knowledge to teach to the user/student how to obtain the "best" mathematical model for a given problem and also has the knowledge to teach how to perform simulations of the model, and consequently to obtain the dynamics of the biochemical system. Also, the ITS can help in the task of obtaining the "best" control possible for a particular bioreactor, because understanding the dynamics of the system is very important in determining the right path to "control".

The authors have developed before several prototype intelligent systems for mathematical modelling and simulation, see Castillo and Melin [4, 6, 8, 9], and the Knowledge obtained in the development of these systems has been used now to build the "Expert Module" of the ITS described in this paper. On the other hand, the authors have also developed several prototype ITS's for other domains of application, see Castillo and Melin [3, 7], and this experience has been used now for building the ITS described in this paper. However, we have to mention that different problems have been encountered in developing the ITS for MMS that have not been addressed before to our Knowledge. Also, in the ITS for MMS of food production, we are including as a topic in the knowledge base a new method developed by the authors for the identification of bacteria using the fractal dimension, see Castillo and Melin [5], which is based on the experimental fact that different types of bacteria have colonies of different geometrical complexity. This method of bacteria identification is also based on the mathematical concept of the fractal dimension, see Mandelbrot [13], and we consider that is a very important part of the knowledge that the engineers have to receive for understanding the domain of food production using bacteria. Finally, we have to say that the need for building an ITS for Mathematical Modelling and Simulation of FPB arised after the authors did the research on modelling and simulation for bacteria growth control, see Melin and Castillo [15], because all of the knowledge that was discovered had to be learn by the engineers of the food production plants.

2 The problem of food production using bacteria

Many products of considerable economic value for us humans are the result of the metabolic functions of microorganisms. From the industrial point of view, the substrate can be considered as the raw material and the microorganisms as the chemical "micro-industry" that transforms this material into new products. Some of the requirements for this industrial microbiological process are:

1.- The microorganism to be used has to be capable of producing large quantities of the product. The microorganism has a high growth rate and is not harmful to the consumers.
2.- The substrate has to be available in large quantities and has to be relatively cheap. In some cases the substrate is a subproduct of another industrial process.

The product resulting from industrial microbiology can be classified in the following groups: 1) Alcoholic Beverages, 2) Chemical Pharmaceutical Products, 3) Food Supplements and 4) Other Chemical Products. The microorganisms used for developing this products can be Fungi, Viruses or Bacteria.

We will consider briefly the problem of food production using bacteria for industrial purposes. Many different food products are the result of bacteria population and the best example of this is the set of dairy (milk) products.

We will consider the problem of food production for the case of yogurt. In this case the use of Streptococcus thermophilus and Lactobacillus bulgaricus is to produce Lactic Acid which is the critical chemical compound necessary for obtaining this product with the exact properties. The right use of this two types of bacteria in this case, i.e., right temperature and time, results in more quantity of yogurt and of better quality.

Another case is the one of cheese production. Different kinds of cheese require different types of Bacteria and different production processes. However, we can say that in general the goal is to produce the maximum quantity of cheese possible and this requires controlling the growth of Bacteria present and also controlling in the best way possible the conditions of production (temperature and time, for example). We can say almost the same thing for the butter and other food products.

We can say then, that from the industrial point of view the goal is to obtain the maximum quantity of the food product with the desired chemical and biological properties. However, this goal from the engineering point of view can be translated to obtaining the "best" control possible for the food generation process, and this involves modelling Bacteria population in the substrate. This problem will be discussed in greater detail in the next section, where a family of mathematical models (of the Lotka-Volterra type) is presented for food production mathematical modelling.

3 Mathematical modelling of the problem

The mathematical problem is then of modelling the population of bacteria in the substrate as they produce the chemical compound needed for the corresponding food product. For a specific case this may require testing several models before obtaining the "best" mathematical model for the food process. In this paper we consider the best model the one that fits more closely the data for a given problem. After deriving this "best" model, then the problem is of obtaining computer simulations to decide on the "best" control possible for the food generation process. In this paper we consider the best control the one that maximizes food production for a specific case.

For the case of yogurt we need to have a mathematical model for the population of S. thermophilus and L. bulgaricus and the quantity of Lactic Acid produced by both bacteria. In this case we require a Lotka-Volterra type model consisting of a system of three simultaneous differential equations, modelling the situation of two bacteria populations and one chemical compound concentration.

For other food products the number of bacteria needed for production may be different than for our example. In this section we show mathematical models for one, two or three bacteria to explain how we can try in model this problem.

We will consider first the case of using only one bacteria for food production. The mathematical model in this case can be of the following form:

$$dN/dt = r(1 - N/K)N - \beta N$$

$$dP/dt = \beta N$$

where: N = population of bacteria, P = quantity of chemical product, r = rate of growth of bacteria, K= limiting capacity of the environment (substrate quantity) and β = biochemical conversion factor.

We will consider now the case of two bacteria used for food production:

$$dN_1/dt = [r_1 - (r_1/K_1)N_1 - (r_1/K_1)\alpha_{12}N_2]N_1 - \beta N_1$$

$$dN_2/dt = [r_2 - (r_2/K_2)N_2 - (r_2/K_2)\alpha_{21}N_1]N_2 - \gamma N_2$$

$$dP/dt = \beta N_1 + \gamma N_2$$

where: N_1 = population of bacteria1, N_2 = population of bacteria2, P = quantity of chemical product, r_1 = rate of growth of bacteria1, r_2 = rate of growth of bacteria2, K_1 = capacity of the environment for bacteria1, K_2 = capacity of the environment for bacteria2, β = biochemical conversion factor from bacteria1 to product, γ = biochemical conversion factor from bacteria2 to product, α_{12} and α_{21} are coefficients.

Another interesting case will be if one considers two "good" bacteria for food production and one "bad" bacteria that is "attacking" the other bacteria:

$$dN_1/dt = [r_1 - (r_1/K_1)N_1 - (r_1/K_1)\alpha_{12}N_2]N_1 - \beta N_1 - \sigma_1 N_1 N_3$$

$$dN_2/dt = [r_2 - (r_2/K_2)N_2 - (r_2/K_2)\alpha_{21}N_1]N_2 - \gamma N_2 - \sigma_2 N_2 N_3$$

$$dN_3/dt = [r_3 - (r_3/K_3)N_3] + \sigma_1 N_1 N_3 + \sigma_2 N_2 N_3$$

$$dP/dt = \beta N_1 + \gamma N_2$$

where: N_3 = population of bacteria3 (bad bacteria), r_3 = rate of growth of bacteria3, K_3 = capacity of the environment for bacteria3 and σ_1, σ_2 = rate of attack of bacteria3 to the other bacteria.

This last two cases are more interesting from the mathematical point of view, because it is well known fact from Dynamical System theory that a model of three or more coupled non-linear differential equations can exhibit the behavior Known as "chaos". This chaotic behavior has to be avoided for this kind of problems, because we need to have a stable food production process. Then, part of the problem in this case will be to control the food production process avoiding the chaotic regime of this type of mathematical models.

4 Description of the intelligent tutoring system

We describe below the general architecture of the ITS, and then we give a detailed description for each of the modules, except for the Inference Engine. The description of the inference engine is based mainly on the Prolog Programming Language, see Bratko [2], and is not given in this paper.

4.1 Architecture of the intelligent tutoring system

The architecture of an ITS is very similar to that of an expert system (ES) for a specific domain of application. However, there are also some main differences between an ITS and an ES. One of the main differences is in the Knowledge Base (KB), because for an ITS the KB needs to contain Expert Knowledge about teaching (training) methodologies and not only Expert Knowledge about the domain of application. The Knowledge Base of the ITS consists of three parts: Expert Module, Teaching/Learning Module and Student Module. For an ITS the last two modules are the ones that enable the training of new students for the domain and also enable monitoring of their learning process.

4.2 Description of the knowledge base

The knowledge base of the ITS consists of three modules: the Expert Module, the Teaching/Learning Module and the Student Module. The Expert Module contains the knowledge about the domain of MMS for food production, i.e., the knowledge about the methods to obtain mathematical models for food production and also the knowledge to perform simulations on these models. The Expert Module also has the knowledge to generate problems for the user/student to solve. The Teaching/Learning Module contains the knowledge about teaching/training methodologies and the knowledge for diagnosis of student's attributes. The Student Module is a data base of the student's performance history and a catalog of student's attributes.

4.3 Description of the expert module

The expert module contains the knowledge about modelling and simulation for food production problems. First, we describe the knowledge about the method for mathematical modelling that is contained in this module. Then, we describe the knowledge about the method for simulation of the models. Finally, we describe a method for bacteria identification using the fractal dimension.

4.3.1 Method of automated mathematical modelling

The problem of automated mathematical modelling can be defined as follows:

Given: A data set (time series) with n data points, $D = \{d_1, d_2, ..., dm\}$ where $di \in R^n$, $i = 1,...,m$, $n = 1, 2...$.

Goal: From the data set D, discover automatically the "best" mathematical model for the time series.

This problem is not a simple one, because in theory there can be an infinite number of mathematical models that can be build for a given data set, see Rao and Lu [17]. So the problem lies in knowing which models to try for a data set and then to select the "best" one. We can state the problem more formally in the following way:

Let M be the space of mathematical models defined for a given data set D. Let MA = {M_1, ..., Mq} be the set of admissible models that are considered to be appropriate for the geometry of the data set D. The problem is to find automatically the "best" model Mb for time series prediction.

We consider mathematical models as "dynamical systems" of the following form: $dY/dt = F(Y)$
where Y is a vector of variables of the form: Y = (Y_1, Y_2, ..., Yp) and F(Y) is a non-linear function of Y. Other kind of mathematical models are the discrete "dynamical systems" of the following form: $Y_t = F(X)$
where X = (Y_{t-1}, Y_{t-2}, ..., Y_{t-p}) and F(X) is a non-linear function of X. Note that in this case we have deterministic models expressed as differential or difference equations.

The mathematical models for continuous dynamical systems can be one-dimensional, two-dimensional or three-dimensional. We show below some sample models [16] that the intelligent system explores:

a) Logistic differential equation:

$$dY_1/dt = a\ Y_1(1-Y_1)$$

b) Lotka Volterra two dimensional:

$$dY_1/dt = aY_1 - bY_1Y_2$$
$$dY_2/dt = bY_1Y_2 - cY_2$$

c)Lorenz three dimensional:

$$dY_1/dt = aY_2-aY_1$$
$$dY_2/dt = -Y_1Y_3+bY_1-Y_2$$
$$dY_3/dt = Y_1Y_2-cY_3$$

The mathematical models for discrete dynamical systems can also be one, two or three dimensional. We show below some sample models [16] that the intelligent system explores:

a) Logistic difference equation:

$$Y_{t+1} = aY_t(1-Y_t)$$

b) Lotka Volterra two dimensional:

$$Y_{t+1} = aY_t - bY_tX_t$$
$$X_{t+1} = bY_tX_t - cX_t$$

c) Henon map two dimensional:

$$Y_{t+1} = X_t$$
$$X_{t+1} = a - X_t^2 + bY_t$$

In all of the above mathematical models a, b and c are parameters that need to be estimated using the corresponding numerical methods.

The algorithm for automated mathematical modelling for prediction is shown in Figure 1, see Castillo and Melin [6].

STEP 1:	Read the data set D = {d_1, d_2, ..., dm}.
STEP 2:	Time Series Analysis of the set D to find the components.
STEP 3:	Find the set of Admissible models MA = {M_1, M_2, ..., Mq}, using the qualitative values of the time series components.
STEP 4:	Find the "Best" mathematical model Mb from the set MA using the measures of "goodness" of each of the models from the set MA.

Figure 1 Algorithm for Automated Mathematical Modelling.

The expert module contains three submodules for the steps of the algorithm in Figure 1. The Submodule for Time Series Analysis contains the knowledge to extract from the data the components of the time series. We use a classification scheme, see Castillo and Melin [9], based on the use of the fractal dimension as a measure of the geometrical complexity of a time series. This scheme is a set of rules relating the value of the fractal dimension of a Time Series to a set of qualitative values for the components.

The Submodule for Expert Modelling contains the knowledge to obtain the set of Admissible Models considered to be more appropriate for a given problem. This submodule consists of a set of rules relating the qualitative values for the Time Series Components to specific Mathematical Models considered the most appropriate.

The Submodule for "Best Model Selection" contains the knowledge to select the "best" mathematical model for a given problem, i.e., given the set of selected models generated by the step 3 of the algorithm, decide which model is the "best" one to fit the data. In this paper, we are considering as the "best" mathematical model the one that minimizes the sum of Squares of Errors (SSE) for all the Admissible models of the problem. This criteria has the advantage of been valid for all the types of models that we consider in the intelligent system (statistical models and non-linear dynamical systems models).

4.3.2 Method for automated simulation

The task of computer simulation of the non-linear mathematical model of a particular dynamical system, is dependent on the appropriate selection of the parameter values to perform the best exploration possible of the dynamical behaviors of the system. The problem of selecting the "best" set of parameter values for the exploration is not a simple one, since it requires using heuristics to develop a set of admissible parameters AP, considered to be appropriate for the given mathematical model, and then using heuristics and statistical calculations to make the final decision of which parameter values are considered to be the "best" set BP.

The algorithm for selecting the "best" set of parameter values is shown in Figure 2, see Castillo and Melin [8].

STEP 1:	Read the mathematical model M.
STEP 2:	Analyze the model M to "understand" its complexity.
STEP 3:	Generate a set of admissible parameters AP using the initial "understanding" of the model. This set is generated using heuristics (expressed as rules in the knowledge base) and solving some mathematical relations.
STEP 4:	Perform a selection of the "best" set of parameter values BP. This set is generated using heuristics (expressed as rules in the knowledge base).
STEP 5:	Perform the simulations by solving numerically the equations of the mathematical model. At this time the different types of dynamical behaviors are identified.

Figure 2 Algorithm for Automated Simulation of Dynamical Systems.

We are considering in this algorithm the "best" set of parameters values for simulation, the ones that enable the correct identification of all the dynamical behaviors for a particular mathematical model. This identification of all the dynamical behaviors is very important because it will help us in the selection process of the "best" control possible for a particular food production problem.

The knowledge base (KB) of the Intelligent Tutoring System consist of a set of rules containing heuristics and mathematical knowledge about the problem of computer simulation of non-linear dynamical models. This rules were generated by machine learning techniques, using a set of training examples (models). A machine learning program was developed to generate the KB for simulation of dynamical systems. To give an idea of how this knowledge is contained in the KB we will show below some sample rules for several types of dynamical systems.

1) Bidimensional Models: This type of mathematical models can be written in the following manner:

$$X' = \alpha\, f(X,Y)$$

$$Y' = \beta\, g(X,Y)$$

In this model, the Equilibria (X*,Y*) is stable if: $\alpha\, f_x + (g_y - \beta) < 0$, where f_x and g_y are partial derivatives. In AI language we have the rule:

RULE 1:

IF $[\alpha\, f_x + (g_y - \beta) < 0]$ **THEN** Equilibria = stable

Also we have the following rule for a Hopf Bifurcation:

RULE 2:

IF $\alpha_0 = (\beta - g_y)/f_x$ **THEN** Hopf_Bifurcation

We have to note here that the computer program can obtain the symbolic derivatives for the functions in the conditions of the rules. This is critical for the problem of simulation, since we require this derivatives to obtain the values of the parameters α and β.

2) Three-Dimensional Models: This type of mathematical models can be written in the following form:

$$X' = \alpha\, f(X,Y,Z)$$

$$Y' = \beta\, g(X,Y,Z)$$

$$Z' = \gamma\, h(X,Y,Z)$$

In this model, the Equilibria (X*, Y*, Z*) is stable if a, b, c > 0 and (ab - c) > 0, where a, b and c are defined by the characteristic equation: $\lambda^3 + a\lambda^2 + b\lambda + c = 0$. In AI language we have the following rule:

RULE 3:

IF a,b,c>0 **AND** (ab - c)>o **THEN** Equilibria = stable

Other rules follow in the same manner for the different types of dynamical behaviors possible for the mathematical models.

4.3.3 Method of bacteria identification using the fractal dimension

This method uses the fractal dimension to make a unique classification of the different types of microorganisms, because it is a known experimental fact that the colonies of different types of bacteria have different geometrical forms. The problem is then to finding a one to one map between the different types of bacteria and their corresponding fractal dimension, in this way obtaining a unique method of identification of microorganisms for the food industry. The first step in obtaining this map is to find experimentally (in the lab) the different geometrical forms for the bacteria. The second step is to calculate the

corresponding fractal dimensions for the different types of bacteria. This fractal dimension can be calculated for a selected type of bacteria with several samples, to obtain as a result a statistical estimation of the dimension and the corresponding error of the estimation. In order to make an efficient use of this map between the different types of bacteria and their corresponding estimated dimensions, we need to implement it as a module in the computer program.

The method for the identification of microorganisms using the fractal dimension can be stated mathematically in the following form: let M_m be a one to one map between the sets I_m and D_m, where the set I_m can be called set of identifications of microorganisms and the set D_m can called set of fractal dimensions. The set of identifications can be as follows:

$$I_m=\{staphylococcus_{aureus}, streptococcus_{fecalis}, pseudomona_{aureuginosa}, salmonella_{typhi}, ...\}$$

and the set of fractal dimensions can be as follows:

$$D_m = \{D_{sa}, D_{sf}, D_{pa}, D_{st}, ...\}$$

where D_{sa} is the fractal dimension of the Staphylococcus aureus bacteria, D_{sf} is the fractal dimension of the Streptococcus fecalis bacteria, D_{pa} is the fractal dimension of the Peseudomona aureuginosa, D_{st} is the fractal dimension of the Salmonella typhi and so on.

In all of the above the fractal dimension is given by the following definition: $D = [\log(N)/\log(1/r)]$, for an object of N parts each scaled down by a ratio, r. For an estimation of this dimension we can use the following equation:
$N(r) = \beta\ 1/r^D$, where N(r) = number of boxes contained in a piece of object and r = size of the box. Counting the number of boxes for different sizes and performing a regression we can obtain the box dimension of an object.

The authors have implemented this method as a computer program to obtain an automated method of identification of bacteria, see Castillo and Melin [5]. In this case this method is used to verify the types of bacteria that we have at any moment in our industrial microbiology process, in this way controlling the types of bacteria in the process. This is important for quality control because in some cases harmful bacteria can appear during the production.

4.4 Description of the teaching/learning module

The Teaching/Learning Module consists of two parts: The Instructional Knowledge Base and the Controller Module. The "Instructional Knowledge Base" consists of teaching/training methodologies and a rule_base for diagnosis of student's attributes. The "Controller Module" updates the current student's attribute values using the student's history and the current performance. Of course, this Controller uses the Inference Engine as a way to infer the new student's attributes.

To give an idea of the way this Module is structured, we will show some sample rules of the part of the "Instructional Knowledge Base" that performs the evaluation of the learning process. We assume that at the end of a training

session the student is given a "test" by the ITS. For example, for stage one of the Learning process the evaluation is done using three types of questions: direct questions, thinking questions and explanation questions.

In figure 3 we show the rules to evaluate the level of learning of stage one.

In figure 4 we show the rules needed to make a decision on the acceptance of the student to stage two of the learning process.

Rule 1:
IF Percent_of_direct_questions ≥ 90
AND Percent_of_thinking_questions ≥ 80
AND Percent_of_explanation_questions ≥ 75
THEN Evaluation_of_stage_one = excellent

Rule 2:
IF 80 ≤ Percent_of_direct_questions < 90
AND 70 ≤ Percent_of_thinking_questions < 80
AND 60 ≤ Percent_of_explanation_questions < 75
THEN Evaluation_of_stage_one = good

Rule 3:
IF 70 ≤ Percent_of_direct_questions < 80
AND 60 ≤ Percent_of_thinking_questions < 70
AND 50 ≤ Percent_of_explanation_questions < 60
THEN Evaluation_of_stage_one = regular

Rule 4:
IF Percentage_of_direct_questions < 70
AND Percent_of_thinking_questions < 60
AND Percent_of_explanation_questions < 50
THEN Evaluation_of_stage_one = bad

Figure 3 Rules to evaluate the level of learning of knowledge in stage one.

Rule 5:
IF Evaluation_stage_one = excellent
THEN Decision_learning_process = stage_two

Rule 6:
IF Evaluation_stage_one = good
THEN Decision_learning_process = stage_one_revision

Rule 7:
IF Evaluation_stage_one = regular
THEN Decision_learning_process = stage_one_strong_revision

Rule 8:
IF Evaluation_stage_one = bad
THEN Decision_learning_process = stage_one_almost_all

Figure 4 Rules of evaluate possible acceptance of student to stage two of the learning process.

Of course, the above rules are only for stage one of the learning process, but there many more rules in the actual "Instructional Knowledge base" for the other stages of the learning process. Also, this module contains other rules for updating the student's attribute values using the student's history and the current performance.

4.5 Description of the student module

The Student Module consists of two parts: The Student's Attribute Catalog and the Student's Performance History. The Student's Catalog contains information like Cognitive Levels, Thinking Skills, Learning Strategies, Problem Solving Abilities and Domain Knowledge States. The "Student's Performance History" contains information about the past evaluations for each of the students.

In our ITS we classified student's attributes in the following manner. "Problem solving abilities" are classified as: low, intermediate and high-critical thinking. Acquisition of "Domain Knowledge" is considered as: null, shallow, intermediate, deep and expert. The "Learning needs" are classified as: individual, hands-on, experimental, learning cycle. The "thinking skills" are evaluated as one of the following: know, understand, apply, transfer and analyze. The "Cognitive levels" are considered as: concrete, transitional and formal. This classification is very similar to the one described in the papers by Dori [11, 12], however there are some differences reflecting some specific problems in our domain of application.

The student module contains rules to make an evaluation of the student's attributes mentioned above, using as information the results of the tests given to the students in each stage of the learning process. Also, the student module has rules to build the "student's performance history" based on the student's attributes obtained above. Of course, this is also the main information that is used in the "teaching learning module" for the generation of problems and questions for the students during the learning process.

5 Validation of the intelligent tutoring system

The validation of an ITS consists of two parts: 1) testing the knowledge of the domain ("Expert Module") and 2) testing the knowledge of teaching methodologies ("Teaching/Learning Module"). For the first part, we performed an extensive validation of the methods for modelling and simulation for food production using bacteria. A number of real test cases were used to compare the solutions of the experts against the solutions by the "Expert Module". The results of the Computer program (Expert Module) were correct for 95% of the cases considered in the validation process. This in fact only validated that the ITS had the necessary knowledge about the domain (as to be able to teach it). On the other hand, to validate the knowledge of teaching of the ITS, we experimented with two groups of students to see if the learning process was enhanced by the computer program. The results are encouraging, over 90% of

the students were able to learn the domain (at certain level) in less time than without the ITS. This in fact, gave us some evidence that the ITS helps in the understanding of the domain. We recognize however, that more tests are needed to make a complete validation of the ITS. We can conclude from this results that the ITS can be considered an efficient educational (training) tool for the domain of Mathematical Modelling and Simulation of Food Production using Bacteria.

Also, after using the ITS as a training tool in the food processing plants, the users suggested several possible improvements to make it more easy to use and to extend the applicability of the computer program. Several of this possible improvements suggested by them and also some that we think can be useful, can be find in the next section.

6 Conclusions

We have developed an Intelligent Tutoring System for Mathematical Modelling and Simulation of Food Production using Bacteria using AI techniques and Fractal Theory. This computer program teaches the methods of automated mathematical modelling and simulation described before. The idea is to capture the dynamics of bacteria population and at the same time of chemical production, using non-linear mathematical models. The models represent the process of biochemical transformation between the microbial life and their generation of the chemical production. The ultimate goal in food engineering is to maximize the production of food and this can be achieved by performing simulations of the models. The ITS monitors the actual learning process for each of the students until they achieve the required level of understanding of modelling and simulation of food production.

The importance of heaving an ITS for training new engineers in this domain can be seen from several points of view. First of all, from the economical point of view the impact of this ITS is in reducing costs of production for the company, because of the more efficient training method resulting from the application of AI and educational techniques to this domain. Second of all, from the social point of view the impact of this ITS is in making the engineers better at their work (increasing the moral level). Third of all, from the educational point of view the impact of the ITS is in the contribution to the advance of the research in applications and methodologies of AI and Multimedia in Education.

The ITS can be improved in the following ways:

1.- Build a better user interface so that it can be used more easily.
2.- Provide a large domain of mathematical models and more methods of simulation for the knowledge base.
3.- Refine the teaching methodologies that are used to make the system more efficient in the task of training new students.

We are planing to work along the above lines of research in the near future.

Keywords: Modelling, Simulation, Knowledge-Based Systems, Food Industry

References

1. Badiru, A.B. (1992). "Expert Systems Applications in Engineering and Manufacturing", Prentice-Hall.

2. Bratko, I. (1990). "Prolog Programming for Artificial Intelligence", Addison Wesley.

3. Castillo, O. and P. Melin, (1994a). "An Intelligent System for Research and Education of Non-Linear Dynamical Systems", Proceedings of IMACS'94, Vol. 1, pp. 82-85, Georgia Institute of Technology.

4. Castillo, O. and P. Melin (1994b). "An Intelligent System for Discovering Mathematical Models for Financial Time Series Prediction", Proceedings of the IEEE Region 10's Ninth Annual International Conference, pp. 217-221, IEEE Computer Society of Singapore.

5. Castillo, O. and P. Melin (1994c). "Developing a New Method for the Identification of Microorganisms for the Food Industry using the Fractal Dimension, Journal of Fractals, Vol. 2, No. 3, pp. 457-460, World Scientific.

6. Castillo, O. and P. Melin (1995a). "An Intelligent System for Financial Time Series Prediction Combining Dynamical Systems Theory, Fractal Theory and Statistical Methods", Proceedings of the IEEE/IAFE 1995 Conference on Computational Intelligence for Financial Engineering (CIFER'95), pp. 151-155, IEEE Neural Networks Council.

7. Castillo, O. and P. Melin (1995b). "QUACONTRA: Quality Control Training in the Food Industry using an Intelligent Tutor", Proceedings of the Eighth International Conference IEA/AIE '95, p.p. 835-844, Melbourne, Australia.

8. Castillo, O. and P. Melin, (1995c). "An Intelligent System for the Simulation of Non-Linear Dynamical Economical Systems", Journal of Systems, Analysis and Simulation, Vol. 19, pp. 767-770, Gordon and Breach Publishers.

9. Castillo, O. and P. Melin, (1996). "Automated Mathematical Modelling for Financial Time Series Prediction using Fuzzy Logic, Dynamical Systems and Fractal Theory", Proceedings of CIFER'96, pp. 120-126, IEEE Neural Networks Council.

10. Devaney, R. (1989). "An Introduction to Chaotic Dynamical Systems", Addison Wesley Publishing Company.

11. Dori, D., M. Alon and Y.J. Dori (1994). "Team Training Shell: a Groupware, Multimedia-Supported Application Generator", Proceedings of ED-Media 94, p.p. 166-171, Vancouver, Canada.

12. Dori, Y. J. and D. Dori (1994). "Object-Process Analysis of Intelligent Computer Assisted Instruction Shell: The Polymer Courseware-a case in point", Proceedings of ED-Media 94, p.p. 172-177, Vancouver, Canada.

13. Mandelbrot, B. (1988). "The Fractal Geometry of Nature", W. H. Freeman and Company.

14. McCalla, G. (1994). "Artificial Intelligence and Educational Technology: A natural synergy", Proceedings of ED-MEDIA 94, p.p. 47-49, Vancouver, Canada.

15. Melin, P. and O. Castillo (1996). "Automated mathematical modelling and simulation for bacteria growth control in food engineering", Proceedings of AIENG'96, pp. 7-8, Computational Mechanics Publications.

16. Rasband, S. (1990). "Chaotic Dynamics of Non-Linear Systems", John Wiley & Sons.

17. Rao, R. B. and S. Lu (1993). "An Knowledge-Based Equation Discovery System for Engineering Domains, IEEE Expert, pp. 37-42.