

A tool for the continuous assessment and improvement of the student's skills in a mathematics course

J. Soler; J. Poch; E. Barrabés; D. Juher; J. Ripoll.

Departament d'Informàtica i Matemàtica Aplicada. Universitat de Girona.
Campus de Montilivi. 17071 Girona (Spain)
E-mail: jordi.poch@udg.es

Abstract

A system for the continuous assessment of the student's skills in a mathematics course is presented. Its main features are communication through the internet and use of a computer algebra system (Mathematica) to achieve a high degree of interactivity. The assessment is made through a personal exercise book generated by the system for each student at the beginning of the course. The present use of the system in academic activities at the Universitat de Girona and some further possibilities of development and use are commented upon.

On présente un système pour le contrôle continu des étudiants dont les caractéristiques principales sont la communication par internet et l'utilisation d'un logiciel de calcul symbolique a fin d'atteindre un haut degré d'interactivité. On fait le contrôle sur une collection d'exercices individuelle pour chaque étudiant, générée par le système au début du cours. On commente l'utilisation de ce système à l'Universitat de Girona et quelques possibilités de développement.

Keywords: continuous assessment, computer algebra system, interactivity, distance learning.

Mots-clé: contrôle continu, logiciel de manipulation symbolique, interactivité, enseignement à distance.

Introduction

Technology has opened a wide range of possibilities in education and its potential applications pose a challenge both pedagogical and technical to the teacher and researcher. Besides the dramatic increase in communication facilities, a new theater has been opened in recent years, namely that of computer algebra systems (CAS). The new technology-rich environment has prompted a number of projects which focus on both interactive mathematical learning and the renewal of classical mathematical curricula (see Smith and Moore 2001, Smith 1998). Valuable experiences on student-centered learning, such as the Connected Curriculum Project, are currently under way (see Hannah 2001), together with some deep thinking on the nature of learning in interactive environments and its possible dangers (see Krantz 2000, Arnold 1998, Brousseau 1996, Santaló 1994).

The ACME project is intended for the teaching of mathematics in the first years of technical/engineering

studies and it is aimed at combining the possibility of fast and reliable communication with the use of symbolic manipulation software in order to improve and assess the students' skills. The project has a bearing on two general aspects of the educational activity: personalized attention to each student, and assessment of his/her skills.

The Department of Mathematics and Computer Science of the Universitat de Girona has been developing the ACME project, a tool for the continuous assessment of the students' skills, since 1998. Its leading ideas can be summarized in two main lines:

- Communication through the internet, which allows the student to communicate with the system from any computer with a standard internet connection.
- Use of symbolic manipulation software (Mathematica) to create and correct online the exercises of an intermediate-level mathematics course in technical/engineering studies.

The use of the internet as a very convenient means for distance learning is already a standard issue, and needs no further comment. As for the second point, the use of symbolic manipulation software makes possible a high level of interactivity and flexibility and is the main novelty introduced by the ACME system. The acronym ACME stands for the catalan for "Continuous Assessment and Improvement of Skills".

Objectives of the ACME Project

An inter-mediatelevel mathematics course in technical/engineering studies has a great deal of "mechanical" work for the tutor as far as the assessment of the students' skills and progress are concerned. The main objective of the project is to render this mechanical work as fully mechanical as possible, and have it done by a computer. Some of the goals of the system are

- Implement a system of continuous assessment of the student's progress.
- Make personalized attention to the student easier.

- Free the tutor from the task of marking a great number of exercises. The time saved in this way can be dedicated to personal attention to the student.
- Assess the degree of participation of the student.
- Encourage personal work in the student.

Modular Structure of the System

The whole system has been designed with a modular structure which makes it useful in a wide range of situations. The system should be serviceable at any level of mathematics and should be adaptable to subjects other than mathematics.

In particular, it can be of great help to students with an under-average degree of mathematical skill if used as a “zero-level course” preliminary or parallel to the main course, along the lines described in the last section.

Some comments on the actual implementation of the system in Industrial Engineering studies in the Universitat de Girona will be given in the last section as well.

The ACME System

Brief Description

The ACME system provides each student with a personal exercise book, under the general principle of “all exercises are different but all are equal”. They are different as far as their formulation, values of parameters, functions involved, are concerned, but they are equal in the sense that the method to be used in the solution of each particular exercise is the same. The progress of the student through his/her personal exercise book provides the basis for the continuous assessment of his/her skills and gives valuable information on the difficulties he/she meets. This information can be used to guide the student through those subjects which present a greater difficulty to him/her.

The whole course is divided into chapters. Each chapter has a traditional collection of exercises and each exercise must be solvable with a standard method that can be implemented in a Mathematica code. Of course this rules out exercises in which a mathematical proof is asked for, but these are not the main objective of an intermediate-level mathematics course in engineering studies.

Different versions of each exercise are written, which include different formulations as well as a wide range of possible values of the parameters involved. These parameters can be numbers, functions or even names or words (the particular wording of the formulation can actually be thought of as a parameter).

The different formulations of an exercise, together with a number of lists for the possible choices of the parameters and the Mathematica code for solving it, is

what we call a *source exercise*. Each source exercise is then a definite mathematical exercise, with a definite solving method, which is formulated differently, with different values of the parameters, to different students. The set of all source exercises is the collection of exercises of the course and the system generates a personal copy for each student, at the beginning of the course, with a random choice of formulation and values of the parameters involved.

Each student can enter his personal exercise book through the internet and can have any number of exercises open for answer at a given time, though there may be a deadline for answering some or all of the chapters. When the student thinks he/she has got the right answer, he/she enters it and the system executes online a Mathematica code and answers back whether the solution is or is not correct. The Mathematica code is built from the values in the student’s book and the code in the source exercise. All this information is registered by the system.

Tutors can enter the system in a matrix-like way: either they can choose a student and look at the state of his/her work or they can select a chapter and look at its state with respect to the whole group of students.

Finally, there exists an *Edition Modulus* which makes easy the creation of source exercises.

An example of a rather simple source exercise is included next in order to help the reader grasp the idea of the system.

Exercise 2.05 --Level 0

Formulations:

* Find the volume of a solid whose sections along planes perpendicular to the x -axis are P_1 if the base of each section lies between the curves P_2 in the (x,y) -plane.

* The base of a solid is the region enclosed by the curves P_2 in the (x,y) -plane and the vertical sections parallel to the line $x=0$ are P_1 . Find the volume of the solid.

* A solid is considered whose base is the region limited by the curves P_2 in the (x,y) -plane and whose sections along vertical planes parallel to the line $x=0$ are P_1 . Find the volume of the solid.

* The sections of a solid along vertical planes parallel to the x -axis are P_1 which have their base between the curves P_2 in the (x,y) -plane. Find the volume of the solid.

Parameters.

P_1
equilateral triangles # Sqrt[3]/4
squares#1

half-circles (diameter lying on the plane) $\# \pi/8$

regular hexagons $\# 3 \sqrt{3}/2$

rectangles (with height twice the length of the base) $\# 1/2$

isosceles triangles (with height twice the length of the base) $\# 1$

P2
 $y = x^2$ and $y = x^2 + 1$ $\# x^2, (x^2 + 1)/2$.

$y = 3 + 3x - x^2$ and $y = 3x^2 - 6x - 9$ $\# 3 + 3x - x^2, 3x^3 - 6x - 9$.

$y = x^2 - 1$ and $y = 5 + 4x - x^2$ $\# x^2 - 1, 5 + 4x - x^2$.

Remarks: in Mathematica format functions must be written explicitly $y = f(x)$ and only $f(x)$ must be entered. Furthermore:

- a) functions must intersect at two points only.
- b) the region enclosed must be of the form $\{(x, y) : x_1 \leq x_2, f(x) \leq y \leq g(x)\}$

Mathematica code

```
Clear[sol, p1, p2, f, g];
sol=S0;
p1=P1;
p2={P2};
f[x_]=p2[[1]];
g[x_]=p2[[2]];

s=Solve[{y==f[x], y==g[x]}, {x, y}];
s=x/.s;
x1=Min[s];
x2=Max[s];
Integrate[p1*(f[x]-g[x])^2, {x, x1, x2}];
v=N[%];
```

```
If[Abs[(N[sol]-v)/v]<=5*10^(-4),
Print["Correct"],
Print["Not correct"], Print["There is probably a syntax error"]]
```

The system makes a random choice of a formulation among the four possible ones and fills in the parameters P1 and P2 with a random choice from the corresponding parameter values. Each parameter is written in Latex format (left of the #) for the exercise book and in mathematica format (right of the #) for its use in the code (in this case the number given on the right is the area of the section when the length of the base is unity).

The paragraph “remarks” gives a brief explanation that could be useful in case the values of the

parameters should be modified or more possible values added.

The example above is deliberately simple because it is intended only to give an idea of what a source exercise looks like. In this example, a simple numerical check of the answer given by the student is made. It is clear, however, that Mathematica allows much more. For instance, one (or even both) of the functions defining the region could depend on a parameter. Then the volume of the solid would be a function of this parameter and the comparison with the expression entered by the student should be made through the Simplify function provided by Mathematica.

The Student's Interface

The student has access to his/her personal exercise book through a login code and password. The home page of the system contains the following items:

- A menu from which a particular Chapter page can be accessed.
- A link to the tutor's e-mail address for any question the student could have.
- A help menu.

Each Chapter page informs the student of the number of exercises in the chapter, how many of them have been correctly solved, the number of unsuccessful trials (and whether the error was mathematical or syntactical) and the deadline (if any) to answer the exercises of the chapter. There is the option of printing the exercises and the solutions already given.

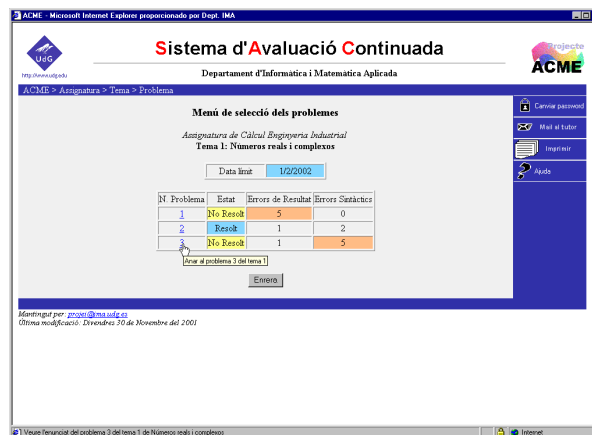


Fig 1. Chapter page

Clicking on any of the exercises (shown by number) in this page gives access to the Exercise page, where this exercise is formulated, together with some remarks on how the answer should be written, since it must be a syntactically correct Mathematica expression. Any comments the tutor may have added to the exercise are shown as well (cf Exercise page in the the tutor's interface).

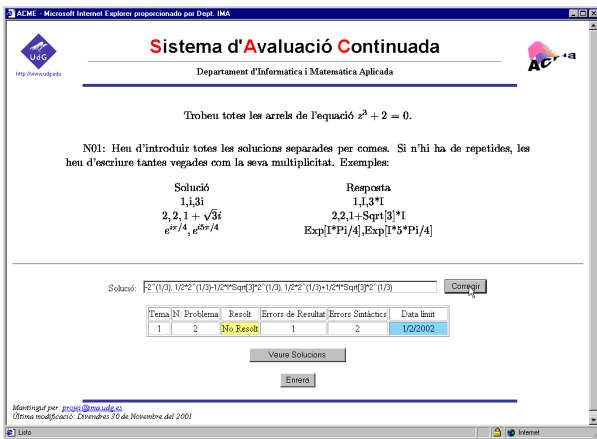


Fig 2. Exercise page.

The Tutor's Interface

Tutors have access to the system through a login code and password. Once in the home page, the tutor must choose one of his/her courses and activate an Options Menu with the following items

1. [Chapter page](#).
2. [Student page](#).
3. [Exercise page](#).
4. [Utilities page](#).
 - [Statistics](#).
 - [Create marks file](#).
 - [Deadlines](#).
 - [Browse log files](#).
5. [Check source exercises page](#).

Each option displays a submenu and gives access to the corresponding page. The contents of each page are described below.

Chapter Page. The [Chapter page](#) shows a list of the students under the supervision of the tutor and the results of each student in this chapter of his/her personal exercise book. The results of an exercise are always displayed in the following format:

1. Status of the exercise, which can have the values Solved or Not Solved,
2. Number of Errors, which is subdivided into two fields: Mathematical Errors and Syntactical Errors. A color code is used to highlight, in either field, the number of unsuccessful trials exceeding a value previously established by the tutor.

This page also shows a statistical summary of the whole chapter with respect to the group of students: percentage of solved exercises, mathematical errors and syntactical errors.

Each student's name is linked to the [Student page](#) and the numbers of the exercises are linked to the [Exercise page](#).

Student Page. The results of the whole exercise book of a particular student are shown in this page. It can be accessed through the [Student page](#) submenu (entering the login code of this student) or directly clicking on the student's name in the [Chapter page](#).

The page shows a list of the exercises, grouped by chapter and given by number. Next to each number the results of the exercise are displayed, in the format stated above.

A summary of the results is also shown at the bottom of the page: percentage of solved exercises, mathematical errors and syntactical errors. There is the possibility of clicking on a button "See all the answers" which gives access to the log file of all the communications between the student and the system.

Each exercise number is linked to the [Exercise page](#).



Fig 3. Student page.

Exercise page. This page shows a particular exercise of a particular student, and can be accessed through the Options Menu or directly from the [Chapter page](#) or the [Student page](#), clicking on an exercise number

The top of the page shows the formulation of the exercise, together with the remarks on how to write the answer, exactly as the student sees it in his/her personal exercise book. A history of the answers tried by the student is displayed, with date and time of the trial and the answer given by the system (correct or not correct).

A window in this page allows the tutor to try a solution and see the answer of the system. In this case the system performs exactly the same operations as if the answer had been introduced by the student. This is a safety device for checking if the system is working correctly (to rule out, for instance, an error in the Mathematica code associated to this exercise). The solutions tried in this way, of course, are not registered by the system.

A link allows the tutor to add comments or remarks to the formulation of this particular exercise at any time. These comments will be displayed whenever the student opens this exercise. The comments to the formulation of an exercise can be made to affect only one student, a group of students with a particular choice of parameters or the whole group.

Utilities Page. This option provides the tutor with a number of utilities. A summary of the results of the whole group of students can be found in the [Statistics page](#), and information on the deadlines for each chapter in the [Deadlines page](#). The [Create marks file page](#) is linked to an Excel spreadsheet in order to create a marks file.

The [Browse log files page](#) gives access to all the log files of the system and is intended mainly for the system manager.

Check source exercises page. This utility examines the structure of a new source exercise in order to cross-check all the inner references (number of parameters, number of formulations, Mathematica code).

Help Module

The object of this module is to make available to the student a model exercise similar to the exercises in his/her personal book, with several levels of help which will show up automatically after a number of wrong answers or if the student asks for help.

The student enters the system in the usual way (login + password) and asks for an exercise of a particular type. Then he/she can either try to solve it or ask for a hint. In the second case an explanation of the procedure to follow is given. When the student thinks he/she has got the right answer, he/she enters it, the Mathematica code is executed online and a “correct” or “not correct” answer is given by the system. After a number of unsuccessful trials a second level of help is activated.

The second level consists of a number of partial questions which lead the student to the solution of the exercise and must be answered in a sequential way. In each of these questions, after a number of unsuccessful trials, the correct answer is shown together with an explanation. If this is not enough, the whole solution of the exercise is shown.

Help exercises are generated from a collection of source help exercises much in the same way as the exercises in each personal book. Their structure, however, is far more complex because they must contain, besides the usual list of parameters and the solving Mathematica code, all the explanation and hints on how to solve it, the partial questions and their own Mathematica codes and the explanation for each partial question. The main trouble is that the way to check an answer to a partial question can be dependent on the answers given to some previous partial questions.

The Help Module is intended as a workbench which does not affect the marks of the exercise book, so that the student has any number of attempts before he feels he can answer an exercise in his book, where the number of trials can be limited.

Use of ACME in a Mathematics Course

Continuous Assessment

The ACME system can be used to assess the progress of the student through the course and the results in his/her exercise book can be made to enter the mark given at the end. Sometimes it may seem advisable to validate his/her mark in the personal exercise book through an exam of exactly the same type of questions the student has been working on. A simple way of doing so is to generate one more exercise book and set an exam with a number of exercises from it. In an ideal case, the results in this exam should be, at least, as good as those obtained in the progress through the exercise book.

A certain degree of flexibility may be advisable in the sense of giving the best mark between that of the exercise book and that of the exam if the latter is better, or if it is worse but the difference is reasonable. On the other hand, if a student has got a good mark in the exercise book and utterly fails the exam, then this is a case for individual study.

If it is felt that the assessment of the student's knowledge should comprise some other questions, such as theoretical questions or questions which are not just “skills”, but require the analysis and subsequent explanation of a more complex problem, a separate exam can be set on this basis. How the different marks obtained in the personal book, in the validating exam and from any other source enter the final mark given to the student is a question that must be established beforehand by the professor in charge of the course. Actually, the system is modular enough to allow many combinations and make the final assessment more or less dependent on the number of errors on each exercise or the number of deadlines not met.

At the Universitat de Girona the ACME system has been working in a first-year Calculus course in Industrial Engineering along the above lines. So far results are very encouraging because most students follow the ACME system (there is the possibility of not following it and then the assessment is made through a classical exam at the end of the term) and the number of consultations to the tutors is remarkably higher than that in other courses. The number of cases in which a student has a good mark in the exercise book and fails completely the validation exam is negligible.

A Module of Basic Mathematics

A substantial number of the students entering technical/engineering studies fall short of the required mathematical standards and it is usually very difficult for them to overcome their handicap because the skills they are in lack of are not the object of any particular mathematics course. These gaps in the mathematical knowledge must be discovered and filled by the student himself/herself outside the main line of the course and the whole task is often beyond his/her reach.

Experience shows that the shortcomings in basic mathematical skills range over a great many topics and the time it takes to overcome them varies widely from one individual case to another, making a traditional course on basic mathematics rather difficult to implement. The flexibility of the ACME system makes it an ideal tool in this situation if a Module of Basic Mathematics is run parallel to the standard course. This module should enable the student to find out, through a series of tests, what his/her shortcomings are, and should provide suitable exercises to help him/her overcome the gaps in his/her mathematical abilities. The student's progress through the Module of Basic Mathematics must, of course, be supervised by his tutor in the same way as in the main course.

Acknowledgments. Thanks are given to Mr. Ferran Prados for his help in the development of the software. This work has been supported by the Universitat de Girona and the Escola Politècnica Superior.

References

- Arnold, V.I., 1998; *Modelos matemáticos duros y modelos matemáticos blandos*. Butlletí Societat Catalana de Matemàtiques, 13.
- Brousseau, G.; 1996, *La didáctica de las matemáticas en la formación del profesorado*. Butlletí Societat Catalana de Matemàtiques, 11.
- Hannah, J., 2001 Using Connected Curriculum Project Modules in a Differential Equations Course. *Journal of Online Mathematics and its Applications* 1 (3).
- Santaló, L.; 1994. *La matemática, una filosofía y una técnica*. Ariel. Barcelona.
- Krantz, S., 2000, Imminent Danger: From a Distance. *Notices of the AMS* 45 (5), 533.
- Smith, D., and Moore, L., 2001, The Connected Curriculum Project. *Journal of Online Mathematics and its Applications* 1 (3).
- Smith, D., 1998, Renewal in Collegiate Mathematics Education. *Documenta Mathematica*, Extra Volume ICM (777-786).