COLLABORATION AND ASSESSMENT
IN A TECHNOLOGICAL FRAMEWORK

Fabrizio IOZZI

Università Bocconi

Istituto di Metodi Quantitativi

V.le Isonzo 25, 20136 Milano, Italy

fabrizio.iozzi@uni-bocconi.it

ABSTRACT

Following one year's experience of lecturing Calculus to undergraduate students at Università Bocconi, Milan, Italy, we have investigated the way students collaborate among themselves and with the lecturers when using an e-learning software (for further information see also the papers by M. Impedovo and G. Osimo); then we studied several approaches to the problem of the assessment of students' knowledge.

In the first part we have focused on the subjects (which are the preferred topics among students and why), the way the discussions are brought on (which kind of discussions are more popular and how the students discuss the subjects) and the impact of the discussions on the performances of the students (are they related to the way the students are involved in the collaborative environment?).

In the second part of this research, three methods to assess students' performances have been compared: a particular mathematical software, the evaluation sections of an e-learning software and a software, developed by the author, specifically designed for lecturers. In the latter case, the technological framework is explained in detail. In particular we discuss: a) the choices made for the interface; b) the intranet set up to guarantee maximum security before, during and after the examination and c) the modularity of the software developed. We consider these aspects interesting by themselves and because the problems they pose are too often neglected.

Keywords: assessment, collaboration, software.

The analysis reported in this paper is related to the first semester of one year's Calculus course to undergraduate students at Università Bocconi, Milan, Italy. The way students collaborate among themselves and with the lecturers has been studied. Students have used at the same time an e-learning software and a mathematical software. In addition, several approaches to the problem of the assessment have been compared and a sustainable solution is outlined. The pedagogical issues involved in this framework are described in depth in another article presented at ICTM 2002 by Michele Impedovo [1]. A complete description of the introduction of collaborative software at Università Bocconi has been published [8].

The goal of this project is a survey of the student online activity in the computer assisted Calculus course.

1. Online behaviour of students

A classroom of 140 students has been monitored during a Calculus course. With respect to analogous courses in other Italian universities, two novelties have been introduced:

a) students used an online collaborative software (OCS) and a particular mathematical software (MS) at the same time;

b) three interim examinations and the final exam were completely electronic, i.e. students had to answer questions and prepare solutions to problems using only the above-cited software.

The students were completely new to OCS and MS. While OCS is very user-friendly, MS requires some skills to be used efficiently. For this reason, a number of lessons on MS were given to students.

Students and teachers in the discussion area wrote 390 messages. Among the 390 messages, 260 messages (66%) were about mathematics, 76 (20%) about software issues, 43 (11%) about the organisation of the course and the last 11 (3%) were either private messages between teachers and students or messages about netiquette (see Fig. 1).

There were 331 discussion threads. Teachers initiated 96 discussions and students initiated 235 discussions. Only 72 out of 140 students created discussion threads. In the average each "computer active" student created three threads. In addition, 28 threads (10%) originated and were answered only by students.

Interestingly, among the 260 messages on mathematics, 69 (26%) originated neither in teacher’s questions nor in elementary mathematical questions or comments, but by students themselves. All of these messages contained an attached file created with MS. There were also 12 more messages with attached files as electronic solutions to homework. These messages showed a great confidence in the use of mathematics. Several applications to economic subjects, e.g. consequences of tax reform, were autonomously found and deeply analysed by students.

Among the 76 messages about MS, 10 (13%) showed an advanced use of this software and investigations into numerical issues. It happened that students even helped university IT staff to solve installation problems of MS. Students generally like to work with computers and therefore they have an in-depth knowledge of the software they work with. This often makes students ask questions about MS not directly related to the course. Obviously, students expect teachers to answer all these questions with the same competence they show in the relevant mathematical issues.

As expected, the distribution of attached MS files was not uniform during the semester. At the beginning of the course students learnt to write mathematically in ASCII (e.g. the meaning of "3^4" is 3 raised to the 4th power) as this is the only way to convey concepts in OCS. The more they used MS the more they choose it to express mathematical ideas and doubts. At the end of the semester all the mathematical threads contained an attached file written with MS.

In Figure 2, exam grades are plotted against the total number of messages written in OCS. There seems to be no correlation between participation in online activities and exams grades. However, as expected, it is seen that those students who electronically answered some of the questions of their colleagues generally obtained good results).

2. The problem of assessment

The issue of technology becomes critical when one considers student assessment. In the move towards new technological courses, attention should be given to whatever is related to examination. In fact, there is no innovation if, after a bunch of technological lessons, examination takes place in a traditional way, i.e. using pencil and paper. However, if changes in the classroom practice are slow, the way teachers prepare and deliver tests is even slower.

In this framework, the choice of a Computer Assisted Assessment (CAA) model is crucial. This section focuses on the technological issues that arose when CAA was elected to be the only method of assessment in the Calculus course given at Università Bocconi.

2.1 Assessment with OCS and MS

Usually, OCS has a section devoted to assessment. As general-purpose software, OCS is not fully case-sensitive (e.g. limited type of questions, limited way of grading answers). Consequently, OCS fits only the needs of an average teacher. In addition, there is no way to embed an MS worksheet with mathematical formulas, even if without active content. In the discussion area of any OCS the issue of writing mathematical formulas is usually solved using a specific ASCII code. However, this practice diminishes the readability of the text that is very difficult to correct. This process is also unsatisfactory from a more general point of view, as students are *writing* and not *doing* mathematics.

On the other hand, no MS has assessment capabilities. In fact, using MS, there is no way to design mathematical questions allowing students to interact and be graded by the system. Despite this limit, it is common practice to assign a problem to be solved creating a worksheet with MS. Then, the file is submitted to OCS in an appropriate section. Other solutions were proposed (e.g. embedding Java applications in assessment software [2]) but they require programming skills and therefore are not practical.

A mixed solution was chosen in the course held at Università Bocconi. Students were given a set of 8 multiple choice questions (MCQs) plus a problem to be solved using MS for each examination. Students wrote mathematics and performed calculations, both symbolic and numerical, only using a computer. Questions were graded automatically by the system while the problem answers were evaluated manually by teachers. This solution has proved to be efficient and reliable. However, teachers are only partially satisfied for the limited flexibility of the system. Hence, in order to evaluate other solutions, existing commercial assessment software (AS) has been reviewed.

AS controls the assessment process in all phases. Commercial AS is in general very powerful in designing questions in many different types such as multiple choice, true or false, etc. An underlying database holds all the information about the exam. Some products even allow the inclusion of Java applets in the question. The design process usually ends with the creation of a comprehensive exam-file (generally of proprietary format) which is submitted to the server and, at the right date and time, to students as html (single or multiple) page. After answering questions, students submit the page via a normal html-form mechanism. The exam is then partially graded by the system and reviewed by teachers. It is also possible to statistically analyse answers. Apparently everything is fine but it is worth taking into account the following issues:

- Question types offered by commercial AS do not always satisfy teachers. Teachers do not make the transition to a CAA if forced to change the way they usually assess their students;
- Mathematical questions often require specific pre- and post-processing e.g. parameterisation;
- PCs sometimes crash. If this happens, there is no way to recover data;
- A unique identification for each student must be assigned. This is often already provided by the university information systems, but assessment software (which is proprietary) can not be integrated with foreign databases. The identification thus relies only on the assessment software security model, which is generally not well designed. Using paper and pencil there is automatic authentication of the writer. However, in a technological environment, all the cares must be taken to ensure one is who he/she claims to be. This is especially true in schools and universities where graduation has a legal value;
- To transfer files from the teacher's PC to the exam server raises security concerns. AS security models are generally very basic and, thus, not adequate. The only solution would be to set up a secure channel between the teacher and the exam server but this relies on IT staff and could be difficult to maintain;
- AS uses proprietary protocols and databases. However, in a few cases and under particular circumstances, some work to integrate AS with other information systems can be done (a detailed description of an actual experience similar to that described below is reported [3]);
- AS is usually paid on a per user basis and therefore is affordable only by few institutions.

In 1998, Università Bocconi started the project named EVEREST whose goal was to adopt a standard system of CAA. All the above issues were considered and it was decided to develop an open source solution that began working in June 2000 and now is adopted in several courses and master classes at the university.

2.2 A sustainable solution

The conclusion of the EVEREST project was that only a completely custom CAA system could address all the issues raised above. XML (Extensible Markup Language [4]) has been elected the common language of the whole application. This choice also allows including particular languages such as mathematics in the description of questions. Finally, it was decided to write the server side of the application using only open source software whose benefits are widely known. This solution was also sustainable: it can be easily extended, maintained and debugged; besides these great advantages, its modularity allows contribution from other people. There is a short description of the application flow below.

The exam is designed using a custom Windows application written in Microsoft Access. An XML file (with documented structure) is the final output. This file contains every detail of the exam: the text of all questions, their single grading and the way questions are to be put together and when. At present, the application allows multiple choice questions with 3 or 5 answers and open questions in which students are required to write a brief essay. The XML file is uploaded to an exam server using a protected https connection, which is similar to those used for e-commerce and grants security. In all phases, the exam server remains hidden from the rest of the academic network. In fact, in order to access the examination, a certificate of authentication is needed.

At the examination day, the system manager locks the PC classroom with a two-click operation. From this moment on, the PCs in the classroom can only connect to the exam server. This also prevents students from using the Internet to communicate to one another. Then, the teacher opens an “exam manager” page in a browser from which he/she can start the exam procedure, close it and see how students are dealing with their questions (see Figure 3). The teacher has complete control over their access to the exam (e.g. he can have them re-enter the exam if they had mistakenly submitted the assignment).

A single exam is subdivided in as many pages as questions (see Figure 4). Each answer is recorded on the server as soon as it is filled. A single PC crash is not a problem: the student can use a different PC without loosing data (obviously, there are more PCs than students).

After closing the exam, the teacher can automatically correct the multiple-choice questions. If more than one teacher is involved in the correction, the application composes as many html files as the teachers and send them via email. As far as the author knows this feature is not available in any of the commercial AS.

The server application has been written in PERL and a new major release, written in PHP and MySql, entered its alpha testing in early 2002. This new release is more modular and allows more types of questions to be accepted and processed by the application. In particular, it permits the embedding of images into questions.

The software interface to the students has continuously changed to reflect students needs and habits in browsing the web. There have been several changes in the layout, buttons and login procedures.

Finally, it should be noted that other researchers are studying the problem of the organisation of CAA sessions. For example, in late 2001, the British Standards Institution issued a guide (BSI 7988) to introduce minimum requirements for any organisation that uses computers to make assessments in the UK [5].

2.3 Future work

CAA is a fast pace moving subject in online learning, often underestimated in its importance in the university organisation. For mathematics its importance is even greater, because two different programs must coexist and work together. CAA requires great care as traditional assessment practices are completely changed. Therefore, a careful investigation of the needs of the institution is necessary.

The use of commercial AS produces several benefits but it also obscures some points in the whole assessment management process. This process is critical in an educational institution and people involved in (e.g. teachers and students) would like to have control on it. A custom solution can be one of the possible answers to this problem. This solution has some advantages: it can be customised to the need of each teacher, fully controlled and monitored, is as secure as the institution computer network and is always open to modifications.

Obviously, this solution can be deployed only with a distributed effort and therefore Università Bocconi plans to release the core of the application under some open source licence.

Among the possible extensions, which are now under investigation, two are interesting for mathematics teachers:

a) the incorporation of some TeX to HTML translator in the application. This incorporation permits teachers to write questions in TeX and students to view them in HTML within a browser window. As TeX is also partially structured, it is possible to pre-process TeX written questions in such a way that, e.g., “a+b” can become “3+4” for one student and “2+5” for another. This approach has previously been used [6].

b) the new MML (Mathematical Markup Language [7]) 2.0 standard from the W3C Consortium provides a further way of translating math into text. Mathematical questions can be written using the most recent versions of MS that can save their output in MML format. For displaying math, the Amaya browser, developed by the W3C Consortium, is capable of displaying MML. Amaya is currently being tested for use as the browser of choice for mathematical display.

3. Conclusions

The analysis of online behaviour of students and computer assisted assessment leads to the following conclusions:

a) a mathematical course can only benefit from the use of MS together with OCS. OCS becomes the area of discussion and exchange of MS files that, after a short period of training, are the way chosen by students to write mathematics. According to the author, the teacher has to take great care in the choice of the MS; an inappropriate (incorrect???) decision might mean that students keep on doing math in a non technological way;

b) the area of discussion is indeed effective: students pose questions and discuss subjects, sometimes autonomously. It is well known that clever students often pose difficult questions. For example, students were taught exponential growth is greater than the polynomial one. A student investigating the behaviour of two sequences, n^{100} and e^n, discovered that even if the latter grows faster than the former, it can be difficult to find for which n e^n is greater than n^{100}. This observation gave birth to two interesting threads, one concerning numerical mathematics and the other computer programs (two students used their programming skills to approximately solve the equation $n^{100} = e^n$);

c) due to the relatively large amount of messages about MS and their advanced level, teachers should be good at using MS and, in general, computers. In addition, a solid programming background helps teachers, as MS usually has programming capabilities. According to the author’s experience, students are generally more confident with programming than with mathematics, thus they often try to solve problems via computer. It is anachronistic to tell them that this solution is not correct.

d) online participation seems to have no effect on students’ performances. Online activities require confidence with computers but not necessarily with mathematics itself. In addition, high participation rates may originate in a high number of questions or in a high number of answers or both. In the author’s opinion, the lack of correlation between students' participation and performances simply means that mathematics should not be confused with the use of any software even if a mathematical one. The collaborative environment is a mean not the goal of mathematical education.

e) The use of MS forces students to learn it . In order to consolidate this practice, schools/universities must have at least adequate laboratories. As students become aware of the computer evaluation, they ask more and more laboratory lessons. This is indeed the main goal of the author (and of the others involved in the project): to turn mathematical lectures into laboratory sessions, having students experience mathematics before they learn it;

f) finally, the technological issue is to be taken seriously. There is a common feeling among mathematicians that computers and software are easily run. In the author's experience this is not true. The use of computers and software cause a wide range of problems. Teachers should be able to face them in the most efficient way. This requires, for example, an in-depth testing of the chosen software and some support from the information technology department and even other staff.

References

[1] Impedovo, Michele, *NT (NEW TECHNOLOGY) HYPOTHESIS*, ICTM 2002 Proceedings

[2] Daugherty, Brian. *Using Java Applets to Deliver Mathematics Assessment*, Proceedings of the 5th CAA conference, http://www.lboro.ac.uk/service/ltd/flicaa/conferences.html

[3] Woodbury, John, Ratcliffe, Mark and Thomas, Lynda. *Building and Deploying an Extensible CAA System: from theory to practice*, Proceedings of the 5th CAA conference, http://www.lboro.ac.uk/service/ltd/flicaa/conferences.html

[4] http://www.w3.org/XML/

[5] http://www.bsi-global.com/index.html

[6] Rewitzky, Ingrid. *Towards Automated Testing*, Proceedings of the 5th CAA conference, http://www.lboro.ac.uk/service/ltd/flicaa/conferences.html

[7] http://www.w3.org/Math/

[8] Renzi, S. Klobas, J. E. .*First steps toward computer-supported collaborative learning in large classrooms*, Educational Technology & Society, 3(3), pp 317-328.

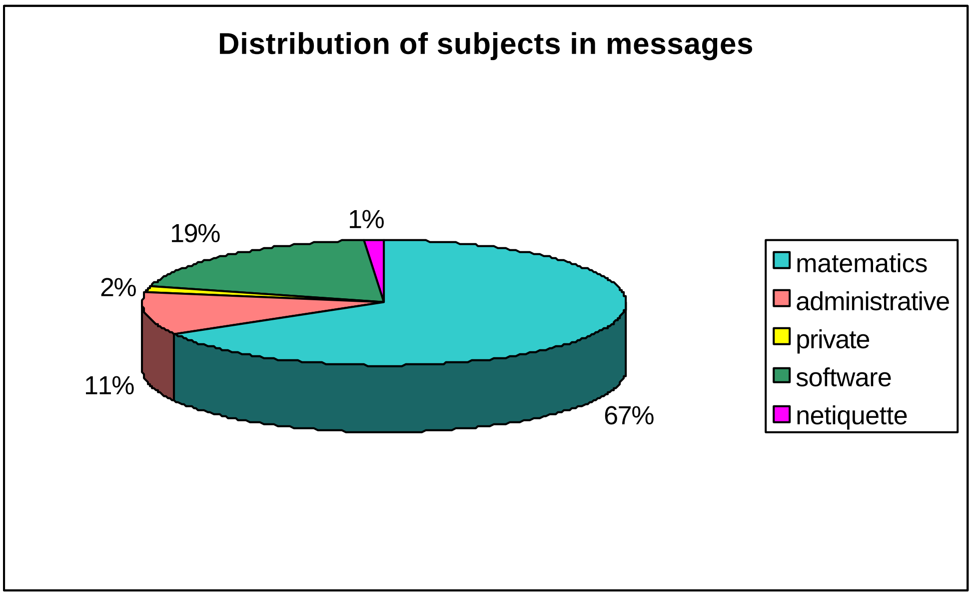

Figure 1: Distribution of subjects in messages in the collaborative area.

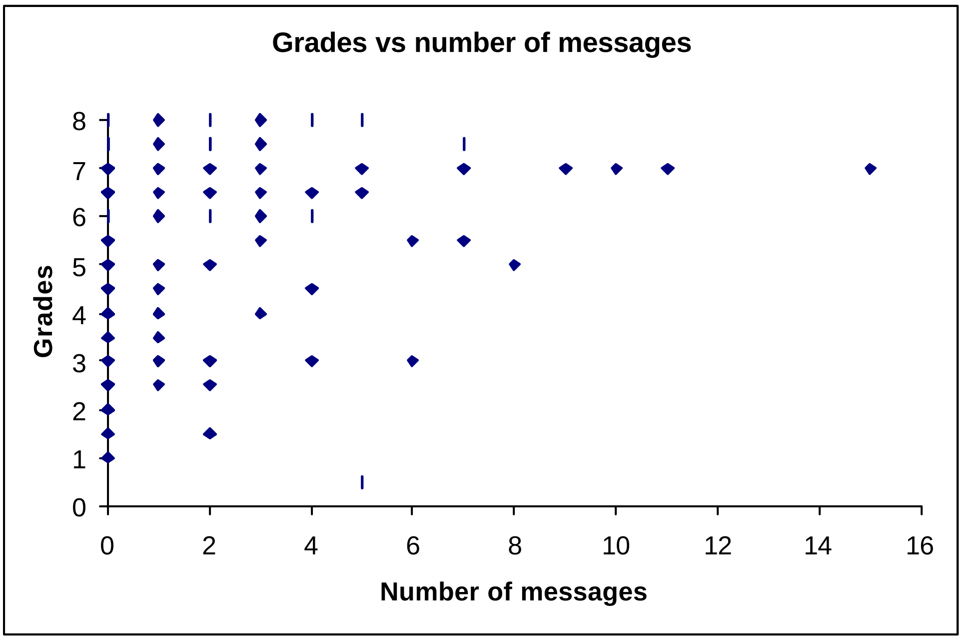

Figure 2: Each point represents one (or more) student. Its abscissa is the number of messages posted by the student to the collaborative area in OCS; its ordinate is the grade obtained by the same student in the mid-year examination. Grades range from 0 (minimum) to 8 (maximum) at a 0.5 step.