Abstract

We have adapted the Loughborough University system of automated MATLAB assessment in engineering to provide assessment and feedback for a new computational maths module in Liverpool. The aim was to make feasible individual feedback to large cohorts of students in a much shorter time than was previously possible at Liverpool. An important aspect of the project was to develop further the system and enhance its portability so facilitating its wider use in other modules and disciplines where MATLAB programming is a key element. Good progress was made in exporting the system and making it more easily adapted. We have developed some associated software to sift out groups of submitted work which shows evidence of collusion. Student reaction at Liverpool was mixed, highlighting the need to get students above a minimal level of programming competence before being able to benefit properly from the automated feedback system.

Background

The need for the project arose when the Department of Mathematical Sciences was given the task of training up to 300 second year engineering students in numerical analysis and programming in MATLAB. Limited manpower resources for assessment and feedback were available. In particular we wished to do this within a time scale where students could learn from feedback before their next assessment. Previous programming training had been provided in a piecemeal fashion across a variety of engineering modules, none of which were able to provide a systematic approach to learning MATLAB, assessment and feedback. Collusion had also been a problem. The learning and teaching aim was to have a more concentrated and integrated attempt to teaching numerical analysis and programming and to use the automated assessment system to provide regular individual feedback and marks in a time-limited class test environment.

Implementation

The code developed at Loughborough was re-organised making it more generic and easier for the user to identify which parts of the code (itself written in MATLAB) require altering for each application. Documentation, in the form of a ‘toolkit’ has been developed to guide the user through the practical aspects of this process and to implementation using a VLE (based on ‘Blackboard’ at Liverpool and ‘Moodle’ at Loughborough).

The particular context at Liverpool was a time limited (2 hour) class test but the code can of course be applied to any electronically submitted assignment or programming
project whose final product can be put in the form (crucially) of a Function MATLAB File (FMF). Basically, the assessment code checks whether the submitted FMF runs and, if it does, tests the requested output against a correct reference code using a variety of inputs. It is possible to anticipate and feedback upon common errors. For each application some effort is therefore required to design the relevant checks and an associated marking scheme. The latter can reward appropriate protection in the submitted code for improper input parameters.

The results, anonymised total marks and individual feedback files, are made available via the VLE and web. Useful marking details and summaries are also provided for the tutor to assist with assembling general feedback, based on the full set of submissions, for the whole class.

An additional facility allows the tutor to convert the marking script (by simply changing one flag) into a packaged version which the students can use to get pre-submission feedback on how well their code is doing against the final marking criteria. This is an important aid when developing and testing prior to final submission. In particular, this can prevent the student performing blind submission of code which is doomed to failure due to some simple programming error.

We have also developed code which compares the active portions of all submitted code (FMFs) to check for collusion. It assigns a numerical correlation coefficient to similar pairs of FMFs and then, where necessary, collects these into groups which exhibit common coding and so warranting further investigation.

Barriers

In this first application at Liverpool, we did not properly anticipate the ‘threshold’ effect alluded to in the abstract. We had not realized how long it would take students to move from the simple command line use of MATLAB to grasping the ideas of programming and, in particular, the concept of a subroutine or function.

We also came to realise that our deliberate strategy of teaching numerical mathematics and programming in an engineering context all at once was overly ambitious. The syllabus and strategy had been planned in consultation with the client department but this approach has proved to be too complex and confusing for the weaker students. Previous implementations at Loughborough had, in part, overcome this barrier by integrating more training in MATLAB prior to using the assessment system.

Enablers

The HE STEM grant has allowed a considerable amount of work to be done on the VLE implementation side and for time to be spent on testing code as it was being developed. Comparison of experience at Liverpool and at Loughborough has been particularly valuable. Both

the Mathematical Science Department and the School of Engineering at Liverpool have provided teaching assistants (TAs) who have helped staff during the training/practical sessions. The largest single enabler has been the staff time of Prof Irving provided by the University of Liverpool and of course the pre-project development time provided by the University of Loughborough.

None of the TAs has had to be involved in the assessment coding or running of the assessment itself. So, in fact, the aim of reduced manpower in assessment has been met since that has only involved one member of staff dealing with all 250 student submissions. The expertise and interpersonal skills of Dr Hughes were invaluable in managing the ongoing support and new developments from the Liverpool Blackboard team, which ensured that the VLE interface worked well for both staff and students.

Evidence of success (impact)

We have successfully addressed key project aims and objectives including:

1. Transferred effective practice in assessing large cohorts to the University of Liverpool, as evidenced by the short marking time required to assess 250 MATLAB scripts per assignment.

2. Feeding back lessons learned and updated code from Liverpool to Loughborough so that processes can be improved with further efficiencies in the marking of scripts and improved plagiarism detection.

3. We have developed a model of implementation which can be used across STEM disciplines beyond the two institutions.

Feedback from Liverpool students was somewhat coloured by the inadequate early MATLAB training shortcomings described earlier. Through these issues encountered, we are now in a position to improve our own processes and to identify potential pitfalls for other institutions.

We have received interest from academics at several other institutions and have taken part in a number of dissemination activities. In particular, we have

- Initiated a Special Interest Group in ‘MATLAB Learning, Teaching and Assessment’ under the auspices of the HEA Engineering Subject Centre;
- submitted a paper to the proceedings of the 10th International Conference on Technology on Mathematics Teaching;
- given a presentation at the CETL-MSOR Conference 2011.

Quality Assurance

Each assignment (class test) is checked by at least two other people to pick up any obvious errors, such as in programming or in the student guidance notes. We operate
a Wiki-style site for contributors to the module so as to facilitate checking and updating of software.

Formal student questionnaires were administered at different points in the module by both client and providing departments. We also ran a module blog on the VLE at Liverpool to keep in touch with student concerns.

**Recommendations for others**

- There are potential time saving benefits through this approach, but this method needs careful planning as it should not be at the expense of teaching quality or the student experience.

- Develop an overall plan and methodology for teaching and assessment of Matlab programming and discuss any issues with other staff and one or two student representatives prior to adopting and implementing this approach.

- Structure MATLAB teaching and learning to ensure basic programming is grasped before engaging students with the automated assessment system. In particular, check that all have proved they can write a simple Function MATLAB File and invoke it from a MATLAB script file.

- Manage student expectations before they are introduced to the system.

- Develop a good working relationship with institutional VLE developers/support staff

**Other observations**

We have given some thought as to whether this automatic assessment system could work in the context of another high level mathematics programming language, for example Maple. This might be done in one of at least two ways:

1. translate everything directly into Maple equivalents;

2. Reconstruct most of the student and tutor interfaces into a web form (CGI script) which then calls a Mathematics engine such as Maple to run the submitted and reference codes. The assessment (comparison and reporting) script can then be written in any language, including MATLAB as now.

There appear to be technical problems with the first approach in that there appears to be no Maple analogue of the packaged (.p) version of a MATLAB file (.m). This is required for the packaged version of the reference code and the student self-testing code. The second approach would seem, in any case to be more generic. There exist well-documented examples of invoking MATLAB and Maple applications from the web and further evidence of the increasing use of mathematical web services in general.

---

**HEA UK Travel Fund**

The HEA’s new UK travel fund will enable staff and students in UK higher education to exchange and disseminate good practice in learning, teaching and assessment and engage with their peers. The funding may be used to help the applicant(s) attend conferences, network meetings and special interest groups that are being run in the UK. Please note that funding may not be used for events that are being held outside the UK. The application should demonstrate that:

- the event is related to at least one of the HEA’s current seven priority themes: assessment, education for sustainable development, employability, internationalisation, flexible learning, retention and success, reward and recognition of teaching;

- the activity would have a positive impact in the sector e.g. including the nature of the engagement proposed by the applicant, scope and purpose of the event;

- the event would give opportunities for professional development or capacity building for the applicant(s);

- the application should offer value for money.

This is an open call running throughout the academic year 2011-12. For further information and the application document please go to [www.heacademy.ac.uk/travel-fund](http://www.heacademy.ac.uk/travel-fund)