Intelligent Environments 2019
A. Muñoz et al. (Eds.)

doi:10.3233/AISE190048

Developing "Smart" Tutorial Tools to Assist Students Learn Calculus, Taking Account of Their Changing Preferred Approaches to Learning

Mastaneh DAVIS[a,1], Gordon HUNTER[a,1], Lynn THALAAL[a], Vivien TRAN BA[b] and Alice WOODING-OLAJORIN[a]
[a] *Faculty of Science, Engineering and Computing, Kingston University, London, KT1 2EE, U.K*
[b] *École Nationale Supérieure d'Ingénieurs de Caen (ENSI-Caen), Caen, France*

Abstract. We present the result of a small study where we investigate what types of resources current students of Mathematics and Mathematics for Engineering prefer for assisting them with their studies of those topics. We found that modern students seem to have a clear preference for on-line resources over traditional textbooks. However, there is currently a lack of good quality resources of that type which allow students to carry-out conventional mathematics exercises on-line and still get appropriate, meaningful and informative feedback on their answers. We then describe our efforts towards addressing this problem through the development of an "intelligent" tutorial system for Calculus which provides feedback tailored to the student's responses, noting where and how they have made common errors.

Keywords. Learning Technology, Intelligent Tutorial System, Student Preferences, Calculus, Mathematics, Technology Enhanced Learning

1. Introduction

Acquiring at least an intermediate level of proficiency in Mathematics is a necessity for success in most Science and Technology disciplines, and many of these require knowledge of and skills in differential and integral calculus. However, many students – particularly those from non-traditional backgrounds, and students with disabilities or other special needs - find these topics difficult [9, 10], but time and resources to provide face to face tutorial support for them are often very limited. In the past, some text-based resources such as "programmed learning" books have provided step-by-step guidance for students wanting to work through example problems in private study. However, by surveying a substantial cohort of current Engineering and Mathematics degree students, we have found that modern students appear to have a marked preference for web-based resources over traditional textbooks.

Although many tutorial websites for Calculus topics do exist, many of these are rather limited in the ways in which they enable students to test themselves on their progress in mastering the relevant skills and techniques. Most such websites only offer multiple choice or short (usually numerical) answer questions. There is a need for self-test resources which allow students to enter their answers in an algebraic format of input, to have these answers checked both against the correct answer and for "common mistakes", and for feedback to be given which is both relevant to the student's answer

[1] Corresponding Authors; E-mail: mast@kingston.ac.uk ; G.Hunter@kingston.ac.uk

and helpful to promote the student learning. If no feedback were given, the students might not be able to identify how or where they had gone wrong in the solution, severely limiting the benefit they gain from using the system.

In this paper, we firstly describe our findings on the preferred approaches of current undergraduate Engineering and Mathematics students to studying mathematical topics outside of class time, including what resources they like to use, which highlights the need for good quality on-line resources. We then describe our attempt to remedy the situation via our *CalculEng* system, which provides a framework for structured questions, allowing algebraic format input of answers by students. Through the use of a Computer Algebra System, our tool allows the student's attempt at solution against both the correct answer, and against expected "common mistake" answers. Appropriate feedback can then be given according to whether the student's answer was correct, included one or more anticipated "common errors", or did not follow an expected pattern.

A limitation of the initial version of *CalculEng* was the requirement for questions, correct answers and anticipated errors, with corresponding feedback responses, all to be hand-coded in XML. However, more recently, we have developed an editing tool (as a Java application) which enables the creation and editing of structured questions in a straightforward and user-friendly way, such that the required XML is generated by the application. This editing tool greatly simplifies the task of creating and editing questions, and should make it easier for teachers to create and share their own resources via *CalculEng*, hence increasing the range of topics covered and number of practice problems available to students.

Pilot studies using our system for a small number of Calculus topics on a relative small group of students have already been carried out, and we hope to extend both the range of topic material covered and the variety of students to whom these are offered in the near future.

The remainder of this paper is structured as follows. In the next section, we review some of the related previous work on the use of computer-based educational technology to enhance students’ learning of mathematics in Higher Education. In section 3, we describe our study to investigate students’ views on how they prefer to learn mathematics, and what type(s) of resources they like using to help them do so. This study showed that modern students have a preference for web-based resources, but (as noted above), there is a lack of the right type of on-line, freely-available practice exercises which can be automatically marked whilst also giving the student user constructive, meaningful feedback on their attempted solutions. In section 4, we describe our attempt to address this via our *CalculEng* tutorial tool. We first outline the development of the early version of *CalculEng*, noting its strengths and weaknesses, before detailing how we have attempted to remove those weaknesses through a new editing tool for structured questions and noting the current status of our system. Finally, we present our conclusions and suggest future directions for this work.

2. Previous Related Work on Educational Technology for Mathematics

There have been many attempts to address the difficulties which many students face when trying to learning mathematical topics. Since the 1960s and 70s, academics teaching Mathematics to Engineering students investigated the potential of using

"Programmed Learning" approaches in course materials and textbooks. In this methodology, theoretical concepts were introduced in conjunction with a large number of worked examples, broken down into small individual "steps" or "tasks" and arranged in a way such that students could "cover up" the latter parts of the model answer, try each task themselves, then reveal the next part of the model solution and compare that with their attempt. This approach led to several highly successful textbooks, including the very popular volumes by Ken Stroud [18] from Coventry University in the UK – Stroud's "*Engineering Mathematics*" is now in its 7th edition [19] and popular as ever with students.

Appropriate on-line resources and exercises should provide additional "virtual tutorial" support at any time and location, which is consistent with the expectations of modern students. Hence, the use of suitable e-materials is expected to improve the quality of student's learning as these materials would provide a platform which will allow students instant access at anytime and anywhere. Furthermore, use of structured exercises will allow the students to develop and test their own knowledge and understanding of mathematical topics, concepts & methods. Linking the system to a Computer Algebra System (CAS) will also enable checking of the student's answers for mathematical/algebraic consistency with, rather than requiring an exact match to, the model answer, allowing answers to be expressed in different, but equivalent and possibly equally correct, forms and still being marked as "correct". Furthermore, use of this CAS, in conjunction with hand-crafted rules encoded in XML, allow the detection of "common errors" in solutions, and the possibility of offering constructive feedback specifically tailored to the error(s) the student has made.

Although there have been previous attempts to produce on-line resources and "self-test" questions for mathematics, most of these have only either provided multiple choice or numerical questions without detailed feedback to students - e.g. *Mathletics* [7] – or are subscription services, in some cases tied to the purchase of particular textbooks – e.g. *MapleTA* [11] and *MyMathLab* [13]. There are a few systems which are free to use on line, allow algebraic input and provide reasonably detailed feedback to students – such as *MathDox* [2] or *CALMAT* [1, 6], although most materials for the former are only available in Dutch, and for the latter are at a rather elementary level more appropriate to pre-University studies. *STACK* [15, 16] is an exception in that it covers material at a higher level, makes use of a CAS and does offer useful feedback to students on their solutions, but several tutors (via personal communications at conferences) have reported that it is not straightforward to create one's own resources for *STACK*.

The *CalculEng* system [3, 4, 5] has been produced to support students with their mathematical studies, in-course assessments and improve their progression. This system offers students a set of exercises on elementary differential and integral calculus and covers material relevant to a good range of engineering topics, including problems on engineering applications. These on-line materials are designed using the *Question and Test Interoperability* (QTI) specification, which is widely being used to represent on-line questions and assessments. Each question is encoded using QTI XML code [12], in which the question, and the dynamic behaviour of the question, are described. The QTI framework provides a programming facility, which allows the tutor to author the mathematical exercises, encoded using XML, and writes the mathematical equations and formulae by employing MathML. These mathematical exercises developed using the existing *QTIWorks* system [14], hosted at the University of

Edinburgh, U.K., which allows the questions to be linked to the freely available *Maxima* Computer Algebra System [17], to check the questions and student's responses for mathematical consistency with the correct solutions.

The *CalculEng* system provides a set of structured exercises, which allow the students to enter their answers in a window (provided especially for their responses) in ASCII-based mathematical format, rather than just making a selection from a list of choices or entering a numerical value. Some of the aspects of these questions, such as specific parameters and coefficients in the equations and formulae, are written to be generated randomly and, by so doing, enable students to develop the ability to recognise the same problems when expressed in different forms. Moreover, the basis of the system is that each question can identify a student's error via a set of rules, which are encoded in XML [12]. The system allows the student's answer to be checked against a list of perceived "common errors" for that type of problem and then provide feedback, tailored to the particular type of mistake made. Therefore, the system provides readily available support, informs students of their mistakes and includes the facility whereby they can request a hint and/or the full solution.

Students are able to use multiple-section structured questions on the application of calculus to engineering problems, with detailed feedback on each step being provided. In these multi-section questions, feedback is revealed to the student in a step-by-step process. Further technical details of how the questions are encoded can be found in [3, 4, 5]. However, as noted in section 1 above, the first version of *CalculEng* required the questions, correct answers and "common mistake" answers all to be encoded into QTI XML by hand – a time-consuming, tedious and error-prone job which deterred many tutors from creating resources for it. Furthermore, this first version of *CalculEng* only permitted relatively simple single part questions, in contrast to typical multi-section questions, developing a theme, with inter-dependencies between the answers to successive sections, common in many mathematical problems. Recently, we have developed a more sophisticated editing tool to address many of these issues, which will be described in section 4 below.

3. How do Current Students Prefer to Study Mathematics ?

Our team for this project consists of an interesting balance – two of us (MD and GH) are highly experienced teachers of Mathematics and its applications, but completed our own mathematical studies many years ago. Whilst we have a lot of experience of teaching mathematical topics, and are very familiar with misapprehensions and common mistakes students make when solving mathematical exercises and problems, our studies pre-date computers being on every desk or the World Wide Web being available everywhere. Our own studies followed a format of formal lectures (in large groups), and tutorials/problems classes (in smaller groups), plus self-study using our lecture notes and textbooks, plus working through exercises set by our teachers. However, the other three members of our team are current students – one of undergraduate Mathematics (AW-O) and two Masters level students of Engineering (VTB and LT), both of whom had had to study a substantial range of mathematical topics during their Engineering degrees.

3.1. Methodology for our Study

As a group, we devised a set of questions which would be put to student volunteer participants – all of whom study a substantial amount of Mathematics, including differential and integral calculus, within their degree programmes. Many of these questions would have responses on a Likert-type scale (often of the general type "Strongly agree", "Agree", "Neither agree nor disagree", "Disagree" or "Strongly disagree"), whilst others related to their preferred modes of study or the types or resources they liked using to help them study and learn mathematical topics. Several such questions allowed the responder to choose "Other", in which case they were asked to specify to what their choice of "Other" referred. A few questions allowed completely free responses, in their own words, from the student participants. The set of questions to be used, the "Information for Participants" sheet, "Consent to Participate" form and details of how the resulting data would be stored (and how long for) and what they would be used for were all sent to the Ethics Committee of Kingston University's *Learning and Teaching Enhancement Centre* (LTEC), from which the project received approval to proceed.

Four groups of students were approached with requests to participate in our study. Two groups were Bachelors degree specialist students of Mathematics (19 first year, 19 second year, all full-time), another group were first year Bachelors degree students of Civil Engineering (18 Students : 8 full-time, 10 part-time), whilst the third group were first year students of Aeronautical Engineering (16 students, all full-time). All these groups had to study Calculus as part of their degree programmes. In total, we interviewed 72 students in the present study.

3.2. Results and Discussion

We found some similarities, but several notable differences between the four groups studied. For the purposes of this paper, we will focus on their preferred approaches and resources to help them study and revise mathematical topics. The data for the two groups of Engineering are summarized in Figure 1, whilst the corresponding data for the two groups of specialist Mathematics students are presented in Figure 2.

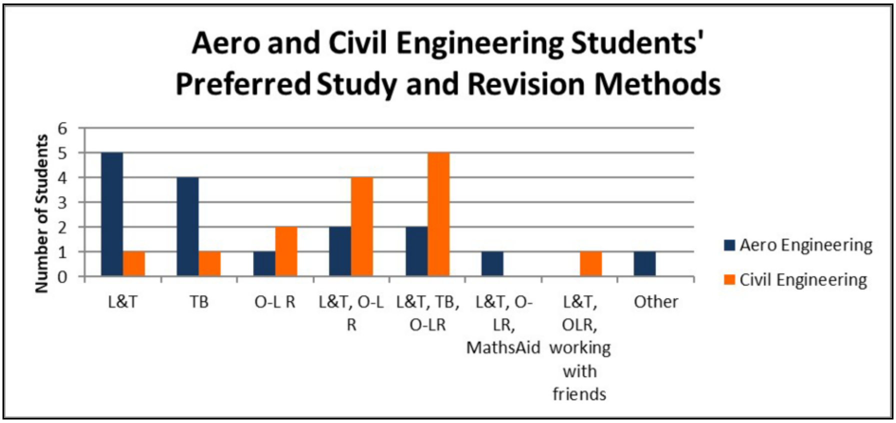

Figure 1. Engineering Students' Preferred Methods and Resources for Studying Mathematical Topics

In Figures 1 and 2, L&T means Lecture notes and Tutorial Problem sheets provided by lecturers, TB are traditional Text Books, O-L R are On-Line Resources, including text-based websites and video-based resources such as *Khan Academy* and *YouTube*, plus on-line quizzes. MathsAid is the University's "drop-in clinic" student support scheme for mathematics topics, and the others are specific combinations of the above.

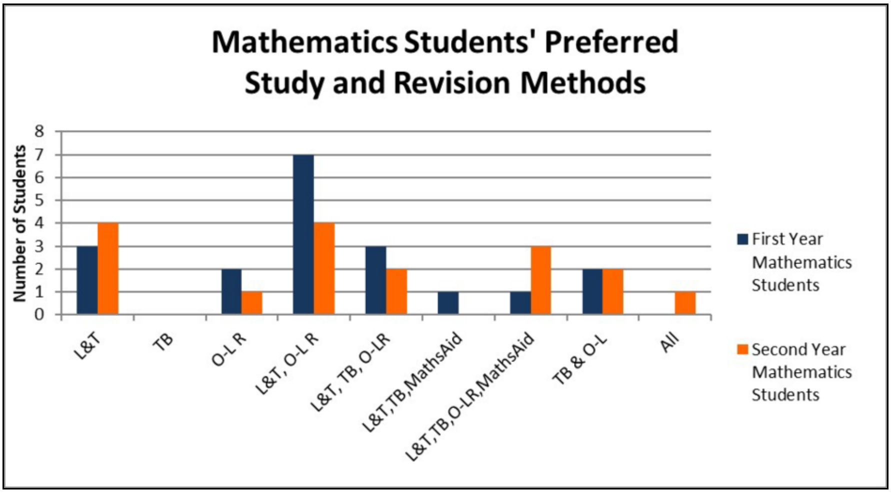

Figure 2. Mathematics Students' Preferred Methods and Resources for Study. Abbreviations are as for Figure 1.

Figures 1 and 2 above show some interesting contrasts between the groups. Aeronautical engineering students seem to have a preference for more traditional learning resources, namely lecture notes, problem sheets and textbooks, whereas the Civil engineering students mainly preferred on-line resources. The mathematics students seemed to favour a mixture of on-line materials and lecture notes and problem sheets, with none claiming to make extensive use of traditional text books. *MathsAid* seemed to be more popular with second year (Level 5) students, possibly because they were more familiar with it being available. Overall, On-Line Resources (of various types) were popular with the majority of the students surveyed, with only 18 out of the 72 claiming not to make much use of them, compared with only 5 out of the 72 who claimed to rely primarily on traditional text books. (This contrasts with the findings of [3] where only 23% of the respondents in 2015 claimed to make regular use of web-based resources.) These findings strengthen the argument for investing in the development of high-quality, easy to use and extend on-line resources for the teaching and learning of mathematics in degree level studies, covering many topic areas and providing practice examples which offer useful, meaningful feedback to their student users. Further statistical analysis of the participants' responses – including those to the other questions is currently underway.

4. *CalculEng* – our on-line tutorial tool for helping students learn Calculus

As discussed above, we have identified a need for high quality tutorial resources to help students learn and practice mathematical topics such as differential and integral calculus. Ideally, such materials should offer students the opportunity to enter their answers in standard mathematical notation (not just allowing multiple choice or short numerical answers), check the answers for consistency both with the "correct" model solution and with anticipated "common error" answers, and then provide the student meaningful feedback which will assist his or her learning of the topic or technique. As noted in section 1, on-line tutorial exercises which do not provide feedback on a student's answers will make it hard for the student to identify where, how or why they made mistakes. This will impair "feed forward" progress towards them fully understanding and mastering the topics and techniques being learned. These are all things which we attempt to address in our *CalculEng* system.

4.1. The early versions of CalculEng

The initial versions of *CalculEng* [3, 4, 5] required a lot of preparation on the part of the tutor. The tutor had to set an appropriate single part question, work out the "correct" answer, and also try to anticipate several mistakes which a less able or slightly careless student might make – for example, differentiating the given function when the question asked the student to integrate it, or forgetting to multiply by a necessary factor. The correct answer and each of the anticipated "common mistake" incorrect answers would then have to be encoded into QTI XML by hand, along with appropriate feedback comments for each response. This was a very tedious, time-consuming and potentially error prone process for tutors, and deterred many of our colleagues from using the system.

An example *CalculEng* question, with its response to the student's incorrect answer, is shown in Figure 3. Further examples, together with the corresponding QTI-XML code, are given in [4].

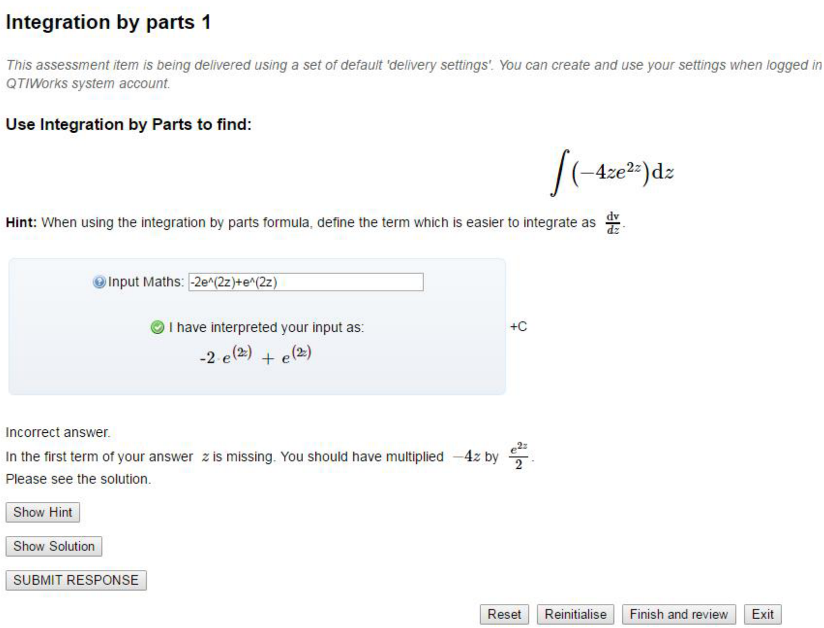

Figure 3. An example *CalculEng* question, including a student's "common mistake" incorrect answer, and the system's feedback appropriate to that particular "common mistake".

The early versions of *CalculEng* only permitted simple single part questions, whereas more realistic mathematical problems often consist of several related parts. For example, consider the following problem from elementary Dynamics :

"An object is oscillating, undergoing simple harmonic motion, such that its displacement at time t is given by $x = 3\cos(2t) + 5\sin(2t)$. (a) Find the velocity, v, of the object; (b) Hence find the values of t which give maximum or minimum values of the displacement; (c) Find these maximum and minimum values of the displacement."

This problem first requires the student to differentiate the function specifying x with respect to time t, to give the velocity v. He/she then has to solve the equation $\mathrm{v} = 0$ for those values of t which make v zero. Finally, the student then has to substitute each of those values of t into the expression defining x to find the actual maximum and minimum values of that displacement x. This type of structured question, with inter-dependencies between its sections, was not possible within the framework employed by the first versions of *CalculEng*, so was something we addressed in the latest version.

4.2. The latest version of CalculEng and our Question Creation and Editing Tool

As noted above, the initial version of *CalculEng* had serious limitations, which restricted the extent to which it could be easily used and extended by colleagues. To address these points, we created a *Java* tool to make the creation and editing of questions much more straightforward, allowing teachers essentially to type-in questions as a combination of plain text and *ASCII-Math* notation for mathematical symbols, plus permitting dependencies between the answers to successive sections of questions, as in the elementary Dynamics example above, including awarding partial credit for "follow through" errors. For example, if the student has incorrectly differentiated the expression for the displacement, but then had found the appropriate values of the time t which made his/her incorrect expression for the velocity v equal to zero, our new version of CalculEng could award partial credit for those incorrect, but consistent with the earlier error, values.

The essential features of the editing tool were identified as :

(i) A computer software package which will enable a teacher to specify a question in relatively natural mathematical language, and the corresponding "correct" and "common error" answers, with appropriate feedback, then

(ii) Convert this question (and both "correct" and "common mistake" answers) into correct QTI XML code.

and the additional desirable features as :

(iii) Allow multi-part questions, where subsequent parts depend on results from earlier parts, **BUT** such that

(iv) the system can also check if (incorrect) answers to later parts were consistent with earlier errors (i.e. were "follow-through" errors).

4.2.1. Implementing the Question Creation and Editing Tool

We implemented this tool in *Java* using *NetBeans* on a Windows PC. Our system renders equations and formulae which have been converted to MathML by the free Equation to MathML encoder provided by *Wiris "Maths for More"* [20].

Our editing tool allows three broad types of questions : (1) simple question(s) , primarily used by students for practicing their mathematical knowledge and skills, which gives the student appropriate feedback on each of his/her answers, and can allow the student to be given hints and/or the model solution on request; (2) test question(s), allowing a single attempt by each student and primarily used for assessment. However, this can still give a student feedback appropriate to the student's answer. Finally, (3) a complex or compound question, which can have between 2 and 4 sections, and can allow inter-dependency between answers to successive parts of the question, and award marks according to whether a student's solution is completely correct, partially correct, or completely incorrect, including awarding partial credit for answers consistent with a "follow through" error. The complex question type can also give students feedback on their responses to each part of the question. Examples of use of the editing interface for each type of question are given in Figures 4, 5 and 7.

The allowed dependencies between sections of a complex or compound question are specified by a (topological) tree structure, as shown in Figure 6. The correct path through parts 1 and 2 of such a question is to obtain the correct answer (listed here as "b") to part 1, then follow that up by obtaining the correct answer to part 2. This correct path through the question is marked in red in Figure 6. However, another possibility is that the student makes an anticipated "common mistake" in part 1, but obtains an answer to part 2 which is entirely consistent to his/her answer to part 1. This is indicated as path 2a in Figure 6. The student should get feedback (and partial credit) indicating what he/she had done correctly, and what incorrectly, here. Another, but more surprising, possibility is that the student gets the correct solution to part 2, despite getting part 1 incorrect. This could be after the student spotting that he/she had made an error in part 1 after submitting his/her answer to it, or could be due to luck. This is marked as "2b" on Figure 6, and again requires specific feedback. The final possibility is that the student gets both parts of the question completely incorrect. Again, this requires appropriate feedback to the student. Not surprisingly, the QTI-XML code generated by our system for a compound question can be very complicated.

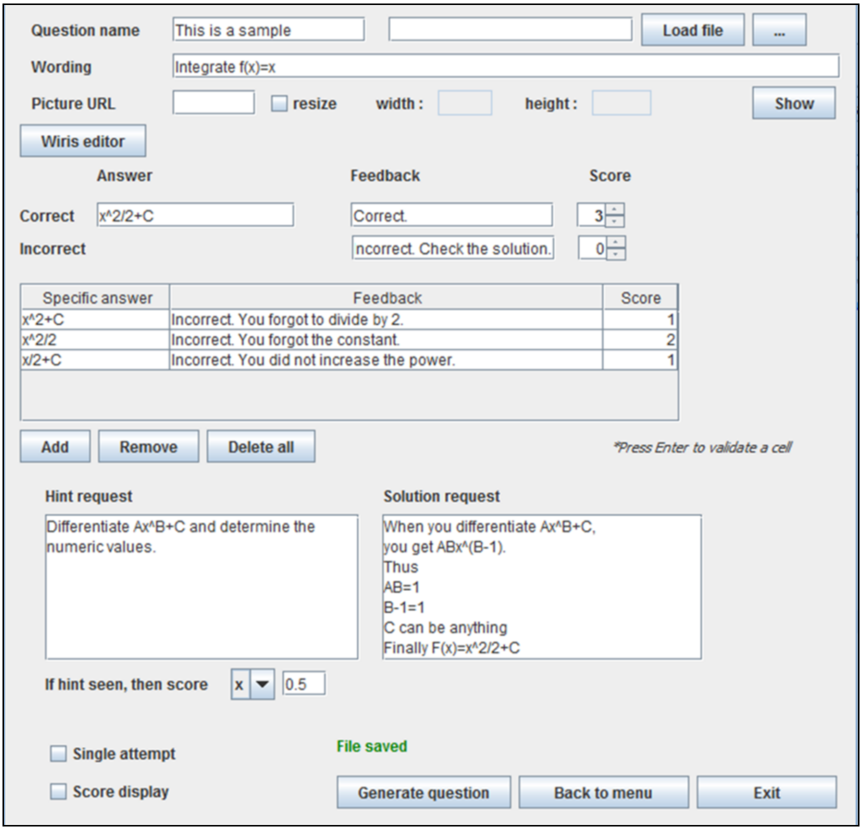

Figure 4. Example of use of the editor for a simple question.

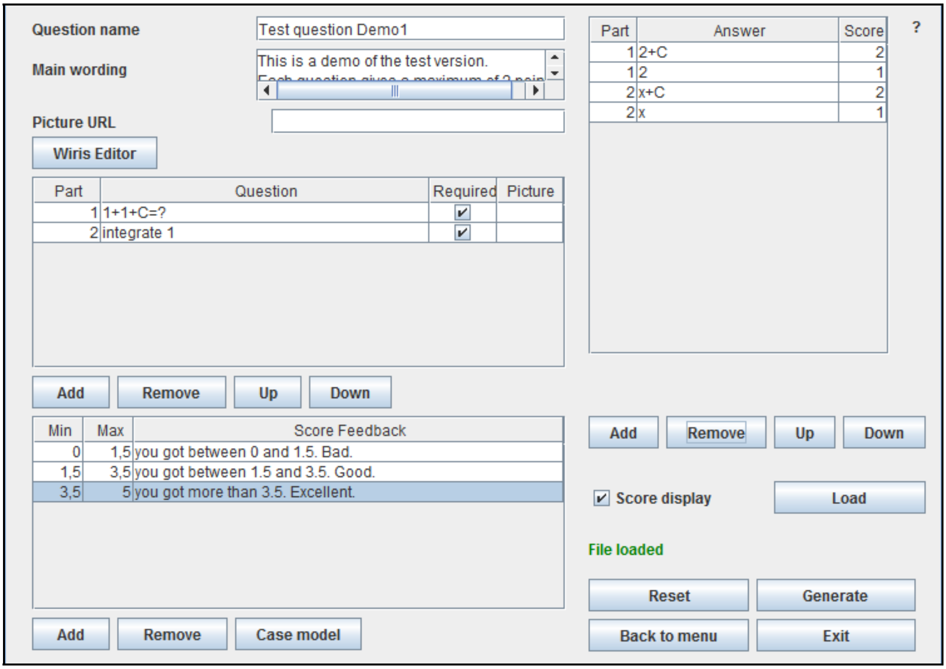

Figure 5. Example of use of the editor for a test question.

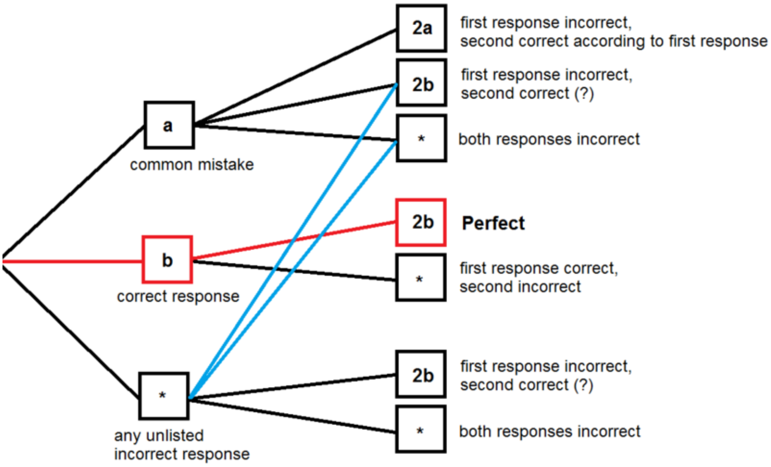

Figure 6. Tree structure indicating the possible dependencies between the answers to two parts of a compound question.

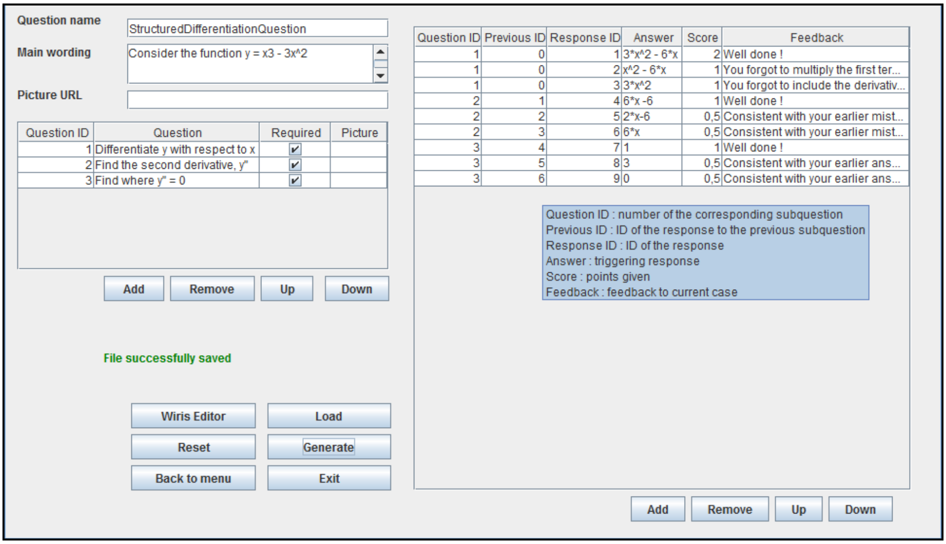

Figure 7. Example of use of the editor for a compound question.

5. Conclusions and Future Work

Our survey of current students' preferred approaches and resources for studying mathematical topics showed that the proportion of students preferring web-based learning resources has increased since our previous study in 2015. This emphasises the need for extensive high-quality on-line teaching and learning materials for more advanced mathematical topics, including self-test tutorial exercises which provide students with appropriate feedback on their answers. Our *CalculEng* system makes some progress to achieving this requirement, our new editing tool allowing us to create a more comprehensive range of exercises for students, including more useful multi-part structured questions with inter-dependencies between the answers to successive sections. This tool will make the task of creating new questions, or editing existing ones, easier, less tedious and time consuming, which should encourage more teachers to make use of *CalculEng* and increase the range of topics covered and the number of exercises available to students, greatly enhancing the utility of the system. Our system has attracted interest from other Higher Education institutions, and we hope to extend both the applicability and evaluation of *CalculEng* to a wider range of students, subject disciplines (e.g. Chemistry, Physics, Economics or Business subjects) and institutions.

However, at present, *CalculEng* is essentially an "Expert System", with the mathematical knowledge it uses encoded by expert teachers, rather than a genuinely intelligent system. In the future, we hope to integrate Machine Learning and/or Deep Learning approaches with *CalculEng*, enabling it to learn from students' responses to the questions, and identifying what errors students actually make, rather than relying on those errors which teachers anticipate students will make when doing the teachers' exercises. We aim to achieve this by logging all the student users' various types of interactions with the system, analyse these statistically, noting which lead to positive outcomes and which to negative ones, and try to "learn" patterns and generate appropriate feedback from these. A methodology for such a logging of interactions, and a preliminary analysis thereof, has been carried out in the context of a system, *NoobLab*, to assist in the teaching and learning of computer programming [8], and we intend to follow that paradigm in order to make CalculEng more "intelligent".

Acknowledgements

Lynn Thalaal's and Alice Wooding-Olajorin's contributions to this project received financial support from Kingston University's *Student Academic Development Research Associate Scheme* (*SADRAS*). Vivien Tran Ba's contribution was partly supported by the ERASMUS student exchange scheme of the EU. We would like to thank all the Kingston University students who volunteered to participate in our study of their learning preferences, and also Ramon Eixarch and Pablo Guerrero of *Wiris "Maths for More"* for some useful advice.

References

[1] D. Booth (1998). "Managing Mathematics with CALMAT". CTI Mathematics quarterly: Maths & Stats Volume 9, Number 3, August.

[2] A.M. Cohen, H. Cuypers & E. Reinaldo Barreiro (2006) "*MathDox* : mathematical documents on the web" In M. Kohlhase (Editor), "OMDoc : An Open Markup Format for Mathematical Documents [version 1.2] (pp. 262-265)" Berlin: Springer-Verlag. Also http://mathdox.org/new-web/index.html

[3] M. Davis, J. Dhanbhoora, G. Hunter and W. Wiesyk (2015a). "*CalculEng* for Virtual *MathsAid*: The Development and Evaluation of a Set of On-Line Tutorial resources for Elementary Calculus", Proceedings of IMA International Conference on Barriers and Enablers to Learning Maths: Enhancing Learning and Teaching for All Learners, University of Glasgow, Glasgow, Scotland.

[4] M. Davis, J. Dhanbhoora, G. Hunter and W. Wiesyk (2015b) "*CALCULENG* : Towards an Intelligent Environment for the Teaching and Learning of Calculus", Proc. Symposium on Future Intelligent Educational Environments (SOFIEE 2015), Prague, Czech Republic

[5] M. Davis, M. & G. Hunter (2016) "*CalculEng* – An On-Line Tutorial Tool to Assist the Teaching and Learning of Calculus", Proceedings of the 18th SEFI Mathematics Working Group Seminar, Gothenburg, Sweden, June 2016, pp 82 – 87.

[6] N. Gorman, D. Healy, M. Marjoram, C. O'Sullivan and P. Robinson (2009) "The Use of Technology in Mathematics Support - approaches used and lessons learnt at ITT Dublin", 4th Irish Workshop on Maths Learning and Support Centres, 4th December 2009, Dublin City University, Republic of Ireland.

[7] M. Greenhow (2008) "*Mathletics* – a suite of computer-assisted assessments", MSOR Connections, Vol. 8 (3), August 2008, pp 7 – 10

[8] G. Hunter, D. Livingstone, P. Neve & G. Alsop (2013) "Learn Programming++: the Design, Implementation and Deployment of an Intelligent Environment for the Teaching and Learning of Computer Programming", Proceedings of 9th IEEE International Conference on Intelligent Environments (IE'19), Athens, Greece, pp 129 - 136

[9] W. James & D. Gardner (1995). "Learning styles: Implications for distance learning". New Directions for Adult and Continuing Education, 67.

[10] B. Jaworski, J. Matthews, C. Robinson & T. Croft (2012) "Engineering Students Understanding Mathematics (ESUM): research rigour and dissemination", https://www.raeng.org.uk/publications/other/engineering-students-understanding-mathematics1

[11] Maplesoft (2015) *MapleT.A.10* http://www.maplesoft.com/products/mapleta/

[12] P. Neve, D. Livingstone, G. Alsop & S. Milne (2012). "User oriented QTI authoring" International Computer Assisted Assessment (CAA) Conference; 10-11 July, Southampton, U.K.

[13] Pearson Education (2015) *MyMathLab* http://www.pearsonmylabandmastering.com/northamerica/mymathlab/

[14] QTI Works https://webapps.ph.ed.ac.uk/qtiworks/

[15] C. Sangwin (2013) "Computer Aided Assessment of Mathematics", Oxford University Press

[16] C. J. Sangwin and M. Grove (2006) "*STACK*: addressing the needs of the neglected learners", Proceedings of the WebAlt Conference, January 5-6, Technical University of Eindhoven, Netherlands, pp 81-95

[17] W. Schelter (1998, updated 2014) "*MAXIMA*, a Computer Algebra System" http://maxima.sourceforge.net/

[18] K. A. Stroud (1970). "Engineering Mathematics : Programmes and Problems". Macmillan

[19] K. A. Stroud & D. J. Booth (2013). "Engineering Mathematics (7th Edition)". Palgrave Macmillan

[20] Wiris "Maths for More" http://www.wiris.com/en/