Workshop Proceedings of the 11th International Conference on Intelligent Environments
D. Preuveneers (Ed.)

doi:10.3233/978-1-61499-530-2-138

CALCULENG : Towards an Intelligent Environment for the Teaching and Learning of Calculus

Mastaneh DAVIS[a,1], Jeraze DHANBHOORA[a], Gordon HUNTER[a,2] and Wioleta WIESYK[a]

[a] *Kingston University, London, KT1 2EE, U.K.*

Abstract. Development of proficiency in mathematics is an essential aspect of many higher education programmes of study. This applies both to specialist mathematics students, and students of many other disciplines, including engineering, most natural sciences and some social/human sciences, business and commercial subjects. Students' knowledge of and expertise in mathematics (or lack thereof), at least at an elementary level, can have a major impact on many other areas of their studies and their subsequent career prospects. However, mathematics is an area which many students find difficult, particularly those from "non-traditional" academic backgrounds, including disabled and mature students, and they often do not realise its relevance and importance to their other courses, nor do they (or can they) devote as much time or effort as they perhaps ideally should, and face to face tutorial support is often limited. In this paper, we describe the design, development and initial evaluation of CalculEng, a system to offer such tutorial support for learning differential and integral calculus - primarily aimed at Engineering students. This system provides structured questions, which are automatically marked, with the aid of a Computer Algebra System, and intelligent, relevant feedback - based on the mistakes made by the student - provided. At present, this feedback is hard-coded using expert-entered rules. However, ways in which the system could be made to intelligently learn patterns of common student errors, and offer feedback accordingly, are being investigated. Our resources should be of particular benefit to students who, due to disabilities or family commitments, may have difficultly attending classes in person.

Keywords. Computer-Aided Teaching & Learning, Calculus, E-Learning, Automated Tutorial Support, Automated Assessment, Intelligent Feedback

1. Introduction

Mathematical skills are core aspects of most scientific, technical, business and finance degrees. Proficiency in mathematics – at least at an intermediate level – is essential to any professional engineer or scientist. However, these mathematical topics are often found difficult by students, particularly those from "non-traditional" academic backgrounds, including disabled and mature students, who may have problems attending classes due to mobility, working or family commitments, and in accessing traditional support resources [9]. Furthermore, the class time available for face-to-face

[1] Corresponding Author. E-mail : Mast@kingston.ac.uk
[2] Corresponding Author. E-mail : G.Hunter@kingston.ac.uk

tutorial support for mathematics in other disciplines is often limited, and class sizes – even for tutorials – are frequently rather large compared with the number of tutorial support staff available. These factors pose major challenges for both students and academic staff.

There have been many attempts to address such issues. 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] – Stroud's "Engineering Mathematics" is now in its 7th edition [19] and as popular as ever with students.

Some higher educational establishments have gone for an approach which restores the more traditional face-to-face approach to tutorial support, but in a more flexible way. An example of this is the system of "MathsAid" drop-in clinics held at Kingston University and some of its affiliated colleges [1]. At each campus, drop-in sessions are held most weekdays (usually for 2 or 3 hours around lunchtimes) run by either a member of academic staff, postgraduate student or senior, trained undergraduate student. These sessions are designed to provide support to students of all disciplines who are having difficulties with mathematical and/or statistical concepts and exercises which they encounter during their studies, no matter what their main area of study is. Hence, in addition to people taking mathematics or statistics as their principal subjects, many students from chemistry, biology, economics, business, or engineering subjects attend these sessions in order to resolve issues and problems. However, it is always emphasized that these sessions are not there to give students answers to assessed exercises [1]. Moreover, these sessions are not substitutes for their regular tutorial sessions on their courses.

More recently, some computer-based and on-line resources and exercises have been developed to address these issues for teaching and learning mathematics. A few example of these are the CATAM [7], CALMAT [2, 5, 11], Mathletics [6], MathDox [3], GeoGebra [8] and STACK [15, 16], and there are also commercial systems such as MapleT.A. [14] and MyMathLab [13]. Each of these systems has some good and some poor aspects, and only a few of them, such as STACK, provide any feedback on the students' answers. CALMAT is primarily targeted at senior high school students and the core material it covers is more at a rather elementary level. Additionally, it only permits multiple choice and short answer questions and, although these questions may include an option to reveal a "hint" to students in the middle of an attempt at solving them, the restricted type of responses which may be entered do not permit checking of a student's answer for algebraic consistency with the model solution, nor do they offer meaningful feedback to students. Similarly, the "Mathletics" system developed at Brunel University in London [6] only offers a limited range of question types and only very basic feedback to students on their answers, which is not tailored to the specific nature of the student's mistakes. Furthermore, MapleTA [14] and MyMathLab [13] are proprietary systems, with subscription payable. Although the MathDox system developed at the Technical University of Eindhoven [3] covers a more comprehensive and advanced range of mathematical topics, and is interfaced with a Computer Algebra

System (CAS) which enables the checking of students answers and working for consistency, most of the resources developed for it are currently only available in Dutch !

In many institutions, a Virtual Learning Environment (VLE), such as BlackBoard or Moodle, is used as a tool for formative and/or summative assessment. The method of assessment normally uses very short answer or multiple–choice questions (MCQs) that are not well suited for the assessment of mathematical topics. Additionally, this form of assessment tool fails to provide a useful and detailed feedback.

However, 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.

Identifying all the above issues has helped us to create an on-line system, called CalculEng, covering a good range of intermediate level calculus topics, allowing students to carry out a series of exercises, and offering hints and meaningful feedback, dependent on the student's answers. These resources not only have the benefits of both computer-based and face-to-face tutorial support but also, by breaking exercises into small stages, in the spirit of "programmed learning" textbooks, they should assist students to learn and master the essentials of differential and integral calculus.

The aim of this project is firstly to evaluate these on-line materials by using both qualitative (e.g. via questionnaires) and quantitative (via students' performance in formative assessments) perspectives, and then to investigate ways in which the system could become more intelligent, by observing and analysing students' answers – including their errors and mistakes – and learn from these, adjusting the feedback offered to students in a way tailored to the individual student's responses, ideally without having to rely on hard-coded rules devised and entered by an expert.

2. Development of CalculEng and Resources for its Use as a Virtual Tutorial Tool

The initial development of CalculEng was inspired by the existing QTIworks system [21] hosted at the University of Edinburgh, U.K. This system allows tutors to author tutorial-style mathematical exercises, encoded using a variant of XML, and have these rendered on-screen, employing MathML to represent mathematical equations and formulae, with the aid of a MathML-LaTeX conversion tool called SnuggleTeX [4]. QTIworks exercises can be linked to the freely-available Maxima [17] Computer Algebra System, allowing student's answers to be checked for mathematical consistency with the correct solution, and hence permitting far more flexibility than just multiple choice or short answer questions which require an exact match to the model solution for the student's response to be considered correct. The QTIworks project encourages tutors to submit their own resources and questions to be hosted on their site. However, at the time of writing, only materials on relatively elementary mathematical topics were available to the public via QTIworks. Notably, those QTIworks resources in existence for the teaching and learning of differential and integral calculus, which are essential aspects of many university courses in science, engineering, economics and other disciplines, were at a rather trivial level, more appropriate to senior high school, rather than University, mathematical studies. This prompted the need to develop new resources, suitable for the teaching and learning of calculus at a sufficiently advanced level to meet the requirements of at least the first year university curriculum.

We decided that the QTI framework was suitable for our needs and, although requiring a large amount of effort per question, a user-orientated tool [12] was available for facilitating the mark-up of questions and their solutions in the QTI variant of XML. We also identified that offering students feedback on their solutions – whether these were correct or incorrect – would be invaluable, and that, provided the nature of the errors they had made in an incorrect attempt at a question could be determined, this feedback could be tailored to the mistake(s) the student had made. Furthermore, if a set of "common, standard errors" were determined for each question and marked-up in QTI XML, the student's response could be checked against this using Maxima and feedback appropriate to that particular error could be given to the student. This set of features should make our QTI-based system a very useful "virtual tutorial" support tool for students learning calculus – and potentially also to their tutors. We decided to call our system "CalculEng", which could either be considered to represent "Calculus Engine" or, since the system was initially intended to support Engineering students learning calculus, "Calculus for Engineers".

In an attempt to follow the "programmed learning" approach adopted by successful textbooks such as Stroud's "Engineering Mathematics" [18, 19], CalculEng offers students structured exercises on elementary differential and integral calculus (including applications of both of these), such that the students can enter their answer to each section as a mathematical expression, typed-in using an ASCII-based mathematical format (an example of this is shown in Figure 1 below), rather than just making a selection, as is the case for multiple-choice questions, or just entering a numerical value. Although the general nature and structure of each question is pre-defined, specific parameters and coefficients in formulae and equations can be selected randomly (within limits or ranges pre-specified by the tutor, e.g. in order to ensure that a quadratic equation involved in the solution has real roots) automatically by the system. The system then checks the student's response for algebraic consistency with the "model answer", using the Maxima computer algebra system [17] and whether the format of the response is correct (e.g. has the student simplified his/her answer sufficiently ?). Furthermore, a set of rules, encoded in XML, for each question, allow the student's answer to be checked against a list of perceived "common errors" for that type of problem (e.g. has the student differentiated a function which the question required him/her to integrate ?), and then provide feedback tailored to the particular type of mistake made. This is expected to assist students with understanding and mastering mathematical concepts, and ultimately should help them in problem-solving situations in their main subject of study. In multi-section questions, detailed feedback will be revealed to the student in a step-by-step process. This feedback facility, customised to the precise question and student answer, is a particularly powerful feature of CalculEng, as it can be used for both formative and summative assessment.

Expression as rendered on-screen : $7e^{(3z-2)}\cos(5z-6)$

ASCII Format : 7*e^(3*z-2)*cos(5*z-6) or 7e^(3z-2)*cos(5z-6)

Figure 1. ASCII-based mathematical format for a student entering a given mathematical expression.

The calculus exercises used in CalculEng were originally selected from conventional paper-based tutorial exercises which two of us (MD & GH) had developed over many years of teaching calculus at foundation and first year university level. However, some further, primarily engineering application-based, exercises were put forward by another of us (JD), who is currently completing his MEng degree in Engineering at Kingston University. Model solutions were prepared by the person who had originally set each question, and some anticipated "common student mistakes" (e.g. the student forgot to include the correct multiplying factor in a differentiation exercise using the "chain rule") identified for each one. Appropriate constructive feedback comments were devised for each anticipated answer – correct, consistent but not simplified, any particular "common mistake" or "otherwise incorrect" – and the question and each of the expected attempts at solutions encoded using QTI XML code [12].

An example integration exercise is illustrated in Figures 2 – 6. Firstly, the question, and a student's answer containing an anticipated "common mistake" are shown, as rendered on the screen, together with the system's feedback for that particular "common mistake" (Figure 2). This is followed by excerpts of the XML encoding of the original exercise, including the Maxima code for the generation of random parameters and coefficients in the question, and the MathML and SnuggleTeX code for rendering the question on the screen, given in Figure 3. In Figure 4, the XML code for determining whether the student's response is correct, or which (if any) of the anticipated common incorrect solutions, or some "other incorrect" solution (i.e. containing an unanticipated error) the student has entered, is shown. Figure 5 displays an excerpt from the XML code corresponding to the feedback appropriate to each of the correct and incorrect answers (including those containing the expected "common errors" and also unanticipated mistakes). Finally, what gets displayed when the student requests a hint for solving the problem is shown in Figure 6.

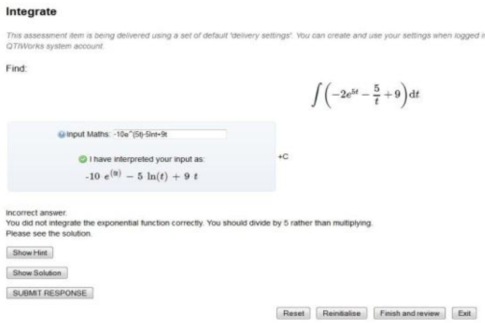

Figure 2. Example CalculEng exercise. The student has made a reasonable attempt at solving the question, but has made one of the "common errors" anticipated by the tutor who had set the question. The student then receives feedback directly relevant to the mistake he/she has made.

```
    <!-- now do the randomisation and mathematical calculations, using maxima -->
  - <setTemplateValue identifier="tDummy">
    - <customOperator ma:syntax="text/x-maxima" class="org.qtitools.mathassess.ScriptRule">
      - <baseValue baseType="string">
          <![CDATA[ simp:true; s1 : make_random_state(true)$ set_random_state(s1); iA: ev((random(7)+2)*(random(2)*2-1),simp); iB: ev((random(9)+2)*
          (random(2)*2-1),simp); iC: ev((random(8)+2)*(random(2)*2-1),simp); iD: ev(random(7)+2,simp); choose(list) := list[1+random(length(list))]; varNames:
          [t,x,z]; mX:choose(varNames); i1overD: ev(1/iD,simp); absB: abs(iB); mArg1: ev(iA*exp(iD*mX),simp); mArg1a: ev(exp(iD*mX),simp); mArg2: ev
          (iB/mX,simp); iAoverD: ev(iA/iD,simp); i1overX: ev(1/mX,simp); mQ18: ev(iAoverD*exp(iD)*mX,simp); mQ6: ev(iA*exp((iD+1)*mX),simp); mQ8: ev(iB*ln
          (abs(iB))/abs(iB),simp); mLnX: ev(ln(mX),simp); mAbsLnX: ev(iB*ln(mX)/abs(iB),simp); mX2: ev(mX^(-2),simp); mQues: ev(mArg1+mArg2+iC,simp);
          mIArg1: integrate(mArg1,mX); mIArg2: integrate(mArg2,mX); mIArg3: integrate(iC,mX); mDeriv: diff(mQues,mX); mInt: integrate(mQues,mX); ]]>
        </baseValue>
      </customOperator>
    </setTemplateValue>
  </templateProcessing>
- <itemBody>
  - <p>
      <big>Find:</big>
    </p>
  - <div class="">
    - <m:math id="mathML0" display="block">
      - <m:mstyle mathsize="12pt">
        - <m:semantics>
          - <m:mrow>
              <m:mo>∫</m:mo>
            - <m:mfenced open="(" close=")">
                <m:mi>mQues</m:mi>
              </m:mfenced>
              <m:mtext> d</m:mtext>
              <m:mi>mX</m:mi>
            </m:mrow>
            <m:annotation encoding="SnuggleTeX">\[ \int(\qv{mQues})\mbox{ d}\qv{mX}\]</m:annotation>
          </m:semantics>
        </m:mstyle>
      </m:math>
    </div>
```

Figure 3. Example QTI-XML code, with interfacing to the MAXIMA Computer Algebra System, to generate a question with randomized coefficients for the particular type of CalculEng exercise, shown in Figure 1, render in using MathML, then obtain the correct solution for it.

```
- <responseCondition>
  - <responseIf>
    - <isNull>
        <variable identifier="RESPONSE"/>
      </isNull>
    - <setOutcomeValue identifier="SCORE">
        <baseValue baseType="float">0</baseValue>
      </setOutcomeValue>
    </responseIf>
  - <responseElse>
    - <setOutcomeValue identifier="oInput">
      - <customOperator ma:syntax="text/x-maxima" class="org.qtitools.mathassess.CasProcess" ma:returnType="mathsContent">
        - <baseValue baseType="string">
            <![CDATA[ ev(RESPONSE,simp);]]>
          </baseValue>
        </customOperator>
      </setOutcomeValue>
    - <setOutcomeValue identifier="ob1">
      - <customOperator ma:syntax="text/x-maxima" class="org.qtitools.mathassess.CasProcess" ma:returnType="boolean">
        - <baseValue baseType="string">
            <![CDATA[ is(is(equal(RESPONSE,mInt))=true);]]>
          </baseValue>
        </customOperator>
      </setOutcomeValue>
    - <setOutcomeValue identifier="ob2">
      - <customOperator ma:syntax="text/x-maxima" class="org.qtitools.mathassess.CasProcess" ma:returnType="boolean">
        - <baseValue baseType="string">
            <![CDATA[ is(is(equal(ev(integrate(RESPONSE,mX),simp),mQues))=true);]]>
          </baseValue>
        </customOperator>
      </setOutcomeValue>
    - <setOutcomeValue identifier="ob3">
      - <customOperator ma:syntax="text/x-maxima" class="org.qtitools.mathassess.CasProcess" ma:returnType="boolean">
        - <baseValue baseType="string">
            <![CDATA[ is(is(equal(RESPONSE,mQues))=true); ]]>
          </baseValue>
        </customOperator>
      </setOutcomeValue>
    - <setOutcomeValue identifier="ob4">
      - <customOperator ma:syntax="text/x-maxima" class="org.qtitools.mathassess.CasProcess" ma:returnType="boolean">
        - <baseValue baseType="string">
            <![CDATA[ is(is(equal(RESPONSE,ev(mArg1+mIArg2/absB+mIArg3,simp)))=true);]]>
          </baseValue>
        </customOperator>
      </setOutcomeValue>
    - <setOutcomeValue identifier="ob5">
      - <customOperator ma:syntax="text/x-maxima" class="org.qtitools.mathassess.CasProcess" ma:returnType="boolean">
        - <baseValue baseType="string">
            <![CDATA[ is(is(equal(RESPONSE,ev(mIArg1+mIArg2,simp)))=true);]]>
          </baseValue>
        </customOperator>
      </setOutcomeValue>
```

Figure 4. Example QTI-XML and MAXIMA code, to determine whether the student's answer to the CalculEng question shown in Figure 1 is correct, is just equivalent to the mathematical expression given in the question, or the student has made one of the "common errors" anticipated by the tutor (and, if so, which error). This is then used to determine which feedback message the student should be given (see Figure 5).

```
- <div class="">
     <feedbackInline identifier="Differentiated" id="feedbackInline0" showHide="show" outcomeIdentifier="FEEDBACK"> You differentiated! </feedbackInline>
  </div>
- <div class="">
     <feedbackInline identifier="Original" id="feedbackInline1" showHide="show" outcomeIdentifier="FEEDBACK"> Your input is equivalent to the expression in
        the question. </feedbackInline>
  </div>
- <div class="">
     <feedbackInline identifier="Incorrect" id="feedbackInline2" showHide="show" outcomeIdentifier="FEEDBACK"> Incorrect - please see the solution.
        </feedbackInline>
  </div>
- <div class="">
     <feedbackInline identifier="Correct" id="feedbackInline3" showHide="show" outcomeIdentifier="FEEDBACK"> Good, you have successfully integrated the
        expression. </feedbackInline>
  </div>
- <div class="">
   - <feedbackInline identifier="Incorrect4" id="feedbackInline4" showHide="show" outcomeIdentifier="FEEDBACK">
        <br/>
        Incorrect answer.
        <br/>
        You forgot to integrate exponential function -
        <printedVariable identifier="i1overD"/>
        is missing.
        <br/>
        You also did not integrate
        <printedVariable identifier="mArg2"/>
        correctly. You forgot to multiply by a constant
        <printedVariable identifier="iB"/>
        .
        <br/>
        Please see the solution.
     </feedbackInline>
  </div>
- <div class="">
   - <feedbackInline identifier="Incorrect5" id="feedbackInline5" showHide="show" outcomeIdentifier="FEEDBACK">
        <br/>
        Incorrect answer.
        <br/>
        You forgot to integrate a constant
        <printedVariable identifier="iC"/>
        .
        <br/>
        Please see the solution.
     </feedbackInline>
  </div>
- <div class="">
   - <feedbackInline identifier="Incorrect6" id="feedbackInline6" showHide="show" outcomeIdentifier="FEEDBACK">
        <br/>
        Incorrect answer.
        <br/>
        You did not integrated exponential function correctly.
        <br/>
        Please see the solution.
     </feedbackInline>
```

Figure 5. Example QTI-XML code. Again referring to the CalculEng question from Figure 1, having identified whether the student has obtained the correct answer, or made a particular "common mistake", the system then generates appropriate feedback for the student.

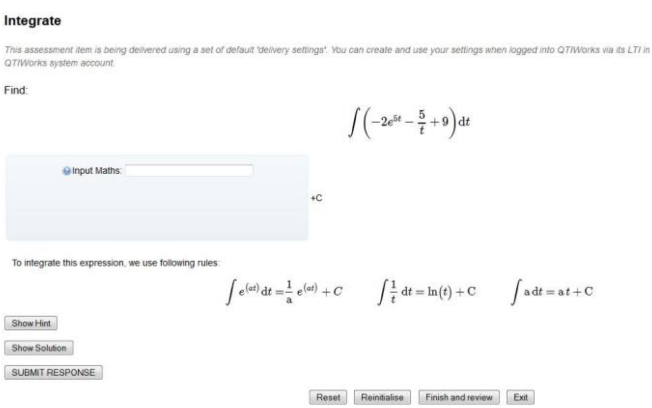

Figure 6. For the same CalculEng question as shown in Figure 2, the student has requested a hint, so the system provides useful information which should help the student solve the problem correctly.

The CalculEng system, as it stands at the time of writing, contains a good range of questions, covering the range of "standard rules" with which a first year student of mathematically-based disciplines would be expected to be familiar regarding single variable differential and integral calculus, Also included are questions on some Engineering applications of these topics – to dynamics, thermodynamics, etc. CalculEng can be accessed through a VLE system, such as BlackBoard, which enables re-use of the materials: copying questions between modules, setting formative and summative assessments and automatically grading the assessments. It is intended to extend the range of topics and examples included – for example, to cover ordinary differential equations and calculus of more than one variable, to other areas of mathematics (such as linear algebra, including simultaneous equations, vectors and matrices) and include material relevant to other application disciplines, such as economics, finance, computer graphics or computer games technology.

3. Preliminary Evaluation of CalculEng as a Virtual Tutorial Tool

It had been planned to carry out a comprehensive evaluation of CalculEng being used with students during the 2014-15 academic year. However, delays in getting the resources ready, the scheduling of parts of some modules across only parts of the academic year, plus a local restriction on surveys of students due to National and intra-institutional Student Surveys, resulted in the initial evaluation only being performed on one group of approximately 40 first year BEng Civil Engineering students, towards the end of the Spring term 2015. This period coincided with a large number of other deadlines for their in-course assessments, with the consequence that only 13 students completed the evaluation survey. The survey took place after the students had been given the opportunity of using the CalculEng system during three 2 hour supervised practical sessions. They were then given a questionnaire regarding their views on their mathematical studies, their confidence with mathematical topics and their opinions on their experience of using CalculEng. The questionnaire they were asked to complete can be found in [23].

During informal discussion, several students indicated that they believed that solving exercises on CalculEng could prove useful to them during their revision and preparation for their exams in May. They also liked the way in which CalculEng presented model solutions in a logical, step-by-step manner, and found the feedback on their answers very helpful. Quantitative results obtained from the students' questionnaire responses are shown in Figures 7 to 12 below.

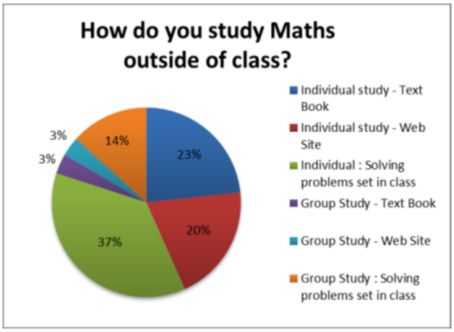

Figure 7. Student respondents' approaches to studying mathematics outside of class

As can be seen from the results in the figures, only 23% of the respondents claimed they regularly used web resources for their out-of-class mathematical studies. 51% mostly practiced exercises set in class, either individually or in groups of peers, with 26% preferring to use textbooks. The majority of the students (71%) carried out between 1 and 3 hours of individual study for the module per week, with only 15%

spending 4 or more hours per week studying for this module outside of class. Since this 30 credit module has approximately 100 hours of scheduled class time, this figure contrasts with the University's expectation that students put in 200 hours of individual study for the module, equivalent to approximately 8 hours per week !

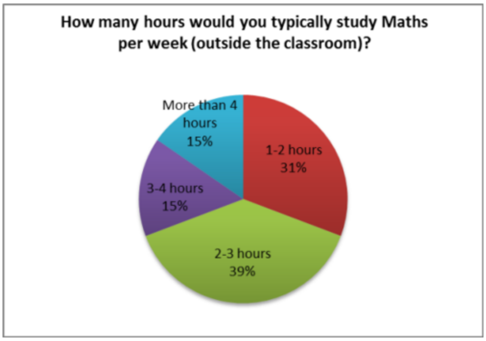

Figure 8. The amount of time per week the student respondents devoted to studying Mathematics outside of class.

Students' opinions on CalculEng were rather divided – roughly equal numbers believed that it could help their understanding of theoretical topics in calculus as did not consider that to be the case, and similarly for whether CalculEng could help them improve their time management skills when solving mathematical problems. However, slightly more students stated that they found CalculEng easy to use than disagreed with that statement.

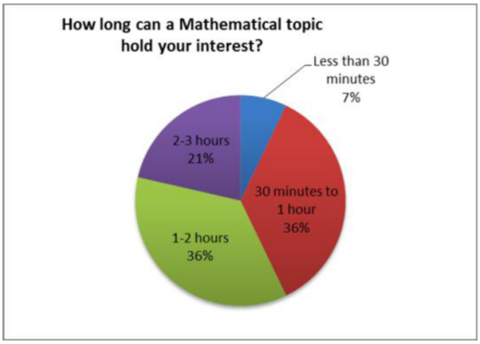

Figure 9. The amount of time the student respondents claimed that any particular mathematical topic could hold their attention.

It should be born in mind that these survey results were based on a very small sample of students, and came after those students had only had a limited opportunity to use the system. It is intended to review the on-line tutorial materials provided in CalculEng, and the questions asked to the students, before carrying-out a more comprehensive evaluation of the system during the 2015-16 academic year. This follow-up study should also allow the students more time to become familiar with CalculEng and the facilities it offers before they are expected to evaluate them.

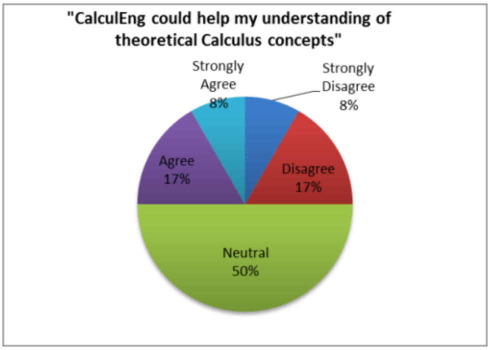

Figure 10. Student respondents' opinions on whether CalculEng could help their understanding of the theory of calculus.

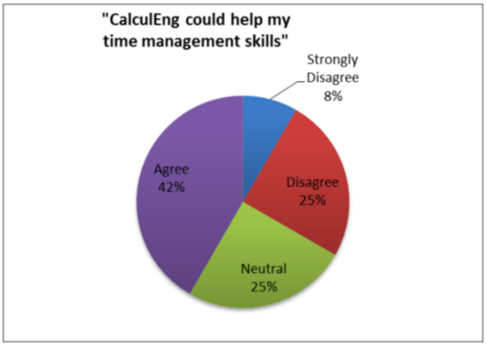

Figure 11. Student respondents' opinions on whether CalculEng could help their time management for solving mathematical exercises.

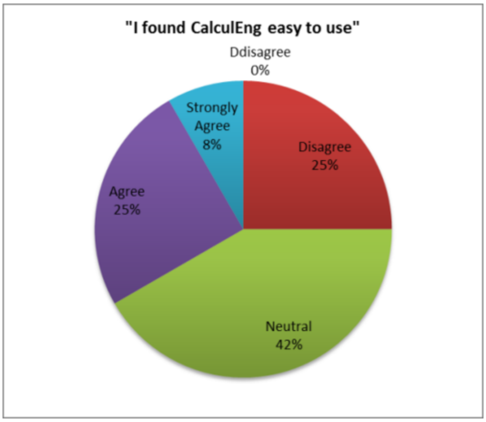

Figure 12. Student respondents' views on how easy CalculEng was to use.

4. Conclusions

In this paper, we have motivated the need for high quality, easy to use, on-line tutorial resources for mathematical topics at University level, particularly for students from “non-traditional” academic backgrounds, including disabled and mature students, who may have problems attending classes due to mobility, working or family commitments, and in accessing traditional support resources [9]. We have described our attempt at developing an on-line system, called CalculEng, for providing such resources, including exercises with appropriate feedback to the user’s answers, for students learning differential and integral calculus. We have performed an initial “pilot study” evaluation of our system on a small group of first year undergraduate Engineering degree students, and presented and discussed the results of this evaluation.

We will use the results of this evaluation to refine and improve our materials, and scrutinize the questions asked in the evaluation questionnaire, in an attempt to remove any ambiguities therein. We intend to carry out a more extensive and comprehensive evaluation of the revised CalculEng system with a larger and more diverse range of students next academic year. We will also extend the number of exercises and the range of topics which they cover and, once our resources have been thoroughly tested and evaluated, we intend to make these available to the wider community.

Finally, the “intelligence” of the current version of CalculEng is at present reliant on rules obtained from experts in the teaching of calculus, which have been “hard coded” into the XML for each question. Thus, CalculEng in its present form could be considered an “Expert System” rather than one which learns. However, opportunities exist to acquire and analyse data on what students actually do when using CalculEng – how they interact with it, what do they do correctly, and what mistakes do they make ? An example of a system which does this, but in the context of the teaching and learning of computer programming, is the NoobLab environment [20] extensively used in first year programming modules at Kingston University. Analysing and interpreting such data could, in conjunction with a machine learning system [22], enable the system to adapt the feedback rules dynamically as more evidence on students’ behaviour and errors when using the system to solve exercises is obtained. Such features could lead to CalculEng becoming a genuinely intelligent environment for providing virtual tutorial support for mathematics.

Acknowledgements

Part of this project has benefitted from financial support from Kingston University’s Summer Internship (WW) and Student Academic Development Research Associate Scheme (SADRAS – JD & WW). We would like to thank Leigh Kemp (Centre for Higher Educational Research & Practice, Kingston University), David McKain (University of Edinburgh) and Sue Milne (ELandWeb Ltd) for providing helpful advice on the QTI system and how to host it on BlackBoard, numerous colleagues at Kingston University for encouragement regarding this project, and those student participants who did complete the evaluation survey.

References

[1] N. Atkins, S. May & D. Marks-Maran (2005). “Widening participation in subjects requiring data handling skills: the MathsAid project”. *Journal of Further and Higher Education*, 29(4), pp. 353-365.
[2] D. Booth (1998). “Managing Mathematics with CALMAT”. CTI Mathematics quarterly: Maths & Stats Volume 9, Number 3, August.
[3] 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
[4] L. Fletcher, S. Milne, S. Ahmed & P. Neve (2009) The FETLAR Project – open educational resources and e-assessment tools for mathematics, 4th Irish Workshop on Maths Learning and Support Centres; 4 Dec 2009, Dublin, Ireland
[5] N. Gorman, D. Healy, M. Marjoram, C. O’Sullivan, 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.
[6] M. Greenhow (2008) Mathletics – a suite of computer-assisted assessments, MSOR Connections, Vol. 8 (3), August 2008, pp 7 – 10
[7] R. D. Harding (1974) “Computer Aided Teaching of Applied Mathematics”, International Journal of Mathematical Education in Science and Technology, Volume 5, pp 447 – 455
[8] M. Hohenwarter, J. Hohenwarter , Y. Kreis, Z. Lavicza (2008). “Teaching and Learning Calculus with Free Dynamic Mathematics Software GeoGebra”. 11th International Congress on Mathematical Education (ICME’08), Monterrey, Mexico.
[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, http://www.hestem.ac.uk/resources/esum
[11] M. Marjoram, C. O’Sullivan, P. Robinson, C. Taylor (2004) “Seven Years of First Year Algebra at IT Tallaght”, Proceedings of Engineering Education Conference 2004 Volume 2 – Recruitment and Retention Practice, pp 189-194
[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] Maplesoft (2015) MapleT.A.10 http://www.maplesoft.com/products/mapleta/
[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] G. Hunter, D. Livingstone, P. Neve, & G. Alsop (2013) “LearnProgramming++ : The Design, Implementation and Deployment of an Intelligent Environment for the Teaching and Learning of Computer Programming” In Proceedings of the 9th International Conference on Intelligent Environments, Athens, Greece: IEEE.
[21] QTIworks Project (2015) “Question and Test Interoperability” https://webapps.ph.ed.ac.uk/qtiworks/
[22] T.M. Mitchell (1997) “Machine Learning”, McGraw-Hill International, New York
[23] M. Davis, J. Dhanbhoora, G. Hunter & W. Wiesyk (2015) “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 Mathematics : Enhancing Learning and Teaching for All Learners, Glasgow, U.K., June 2015