Automation of mathematics examinations

Automation of mathematics examinations

Christopher J Sangwin[a,*], Nadine Köcher[b]

[a] C.J.Sangwin@ed.ac.uk, *School of Mathematics, University of Edinburgh, United Kingdom*
[b] koecher@dhbw-karlsruhe.de, *Baden-Wuerttemberg Cooperative State University, Karlsruhe, Germany*

Abstract

Assessment is a key component of all educational systems, and automatic online assessment is becoming increasingly common for formative work in mathematics. This paper reports an investigation of the extent to which contemporary automatic assessment software can automatically mark answers to questions from existing high-stakes mathematics examinations. The questions are taken from a corpus of publicly available core mathematics questions designed for high-achieving students aged approximately eighteen at the school-university interface. We focus on the extent to which objective properties of each final answer may be automatically established and the extent to which automatic marking reasoning by equivalence supports assessment of students' methodology. Our results show that transcribing existing paper-based mathematics examinations into an electronic format is now feasible for a significant proportion of the questions as currently assessed. The most significant barrier to using contemporary automatic assessment is the requirement from examiners that students provide evidence that they have used an appropriate method.

Keywords: Online assessment; automatic marking; mathematics; examinations.

Highlights

- We examine the extent to which mathematics examinations can be automated.
- Existing technology automatically marks the final answer and reasoning by equivalence.
- A significant proportion of existing mathematics questions can be automatically marked.
- The most significant barrier to faithful automatic marking is a lack of evidence of an appropriate method.

*Corresponding author

1. Introduction

Over the last twenty five years, but particularly over the last decade, there has been a concerted effort to develop software which automatically assesses students' answers to mathematics questions. A summary of early work in this field is given by Sleeman & Brown (1982) and a more recent survey is contained in Sangwin (2013). Some early systems relied only on multiple choice and numeric input questions, but for many years in mathematics students have been expected to type in an algebraic expression which constitutes their answer. More recently serious attempts have been made to automatically assess a student's ability to construct a chain of mathematical reasoning. Examples of such software will be provided in due course. In elementary mathematics, including high-school algebra and calculus, there are many situations where a student can provide an answer and the properties of this answer can be established objectively and automatically using a computer algebra system (CAS). The question we seek to answer in this paper is, to what extent can the assessment of existing mathematics examinations be automatically marked using contemporary automatic assessment software?

Our methodology is to take a corpus of published examination questions, together with the official mark scheme. We have examined the extent to which we can faithfully automatically mark these questions using selected representative contemporary software in a way which is faithful to the published mark scheme. The attempt to genuinely automate marking of existing questions, using existing software, is a "litmus test" which is a long way beyond a purely theoretical or speculative approach.

Constructive alignment, Biggs & Tang (2011), starts with the outcomes we intend students to learn and seeks to align teaching and assessment to those outcomes. All assessments have to balance constructive alignment with other factors such as validity, reliability and practicality. The format of an assessment constrains what is practical and influences validity and reliability. Multiple choice is an extreme example, but paper based examinations are no exception.

> One finding from the literature is that direct translation of paper-based assessments into online assessments is inappropriate; there is a need to revisit question formulation, reflecting on what it is intended to test. The process of creating CAA [Computer aided assessment] questions therefore raises fundamental issues about the nature of paper-based questions as well. (Conole & Warburton, 2005, p. 21)

Therefore, to start with an existing examination format and merely translate questions into a new format without regard for the underlying educational construct they are seeking to test might seem incongruous. If our goal was to construct an online examination, working within the constraints and taking best advantage of the format, this concern would be appropriate. The purpose of the research reported in this paper is to understand the extent to which the published intentions of examiners can be faithfully automatically marked at this moment in time, with software actually in use. That is to say, we are not the examiners and we are not (for the purposes of this research at least) engaged in the process of writing valid examinations from scratch.

Indeed, it is out of a respect for experienced examiners, professionally engaged by a large examination board, that we have started with their questions. A failure to be able to faithfully automatically mark traditional examinations may point to serious deficiencies in contemporary software. The data we seek to obtain may therefore be very useful in setting priorities for developers of such software.

We note that the ability to faithfully repeat and examine all the steps required for passing a classical paper examination is not the gold standard of a computer based test. For many users of

such systems the goal is formative practice. Other users have selected automatic marking because of the practical advantages of using computers and the internet, for example the ability to scale to large groups and to provide rapid feedback. However, the ability to automate the assessment process, while necessary, is not sufficient. Even if all aspects of a traditional examination could be sufficiently covered by a fully automated exam, it does not immediately follow that this is the most convenient way of performing such examinations. For example, the usability of the system could hinder the performance of all or some of the students so that the results are changed. Basic usability of the interface is important but usability could also relate to differences in computer skills and accessibility issues within the system. For example, Galbraith & Haines (1998) sought to disentangle attitudes related to mathematics from those associated with the technology for learning it. Lack of usability testing with students is a limitation of this study which is a question to be addressed by future research.

The previous experience of the authors strongly suggests that the task of devising automatic marking schemes sheds interesting light on assessment design and on what is currently assessed in practice. Indeed, mathematical proficiency consists of several different aspects, see Kilpatrick et al. (2001), some of which can be automatically assessed with computer based examinations more readily than others. Contemporary software is developing rapidly. Examiners experienced in writing questions for traditional paper examinations may not be familiar with what is now possible online. Having established our results, a secondary purpose of our research is to inform examiners and teachers of the extent to which we may automatically mark questions which are currently examined. Whether these questions should continue to be used in examinations is a matter for debate, and is ultimately a personal value judgement.

Indeed, an underlying motivation for undertaking this research is a concern that existing examinations may be automatically marked without due regard to the educational constructs they are seeking to test.

> The issue for e-assessment is not if it will happen, but rather, what, when and how it will happen. E-assessment is a stimulus for rethinking the whole curriculum, as well as all current assessment systems. (Ridgeway et al., 2004, p. 4)

For the purposes of this research we have therefore made no serious attempt to evaluate whether the published questions truly align with stated course goals. Whether or not the corpus of published questions we have chosen do really align with course goals does not alter the fact that teachers will, and do, naturally look to specimen examinations for practical guidance on what and how to teach. Students, naturally, also look to specimen examinations for practice. Many authors, e.g. Burkhardt & Swan (2012), have stressed how important it is to align assessment with the curriculum, going as far as saying that in order to ensure teachers follow the intended curriculum the assessments must cover the goals in a balanced manner.

Similarly, this paper does *not* seek to address the important question of whether existing mathematics examinations actually constitute valid or reliable tests of mathematical expertise. There is a long-standing discussion on this issue. Deciding whether examinations in mathematics are valid is controversial because the decision reflects a set of subjective value judgements about such things as the extent to which students should be fluent in traditional procedures including calculation and algebraic manipulation. For the purposes of this research we are not seeking to define or discuss what constitutes mathematical expertise. Indeed, to do so would potentially confound our research as we have tried to suspend our value judgements and objectively evaluate the extent to which we can automate existing assessments. Instead, we confine ourselves to evaluating the extent to which a question can be automatically marked faithfully to its published

mark scheme with contemporary software. By looking at existing mathematics examinations, the main contribution of this paper is data on the objective criteria actually being used to assess a particular answer and the extent to which these criteria can be automatically marked using currently available software.

This paper is organized as follows. Section 2 discusses the current state of the art in computer aided assessment, and provides background information on the software selected for use in this research. Section 3 defines our methodology for evaluating the extent to which questions can be automatically marked, and illustrates the methodology with an example from the question corpus. Results are given in Section 4, with a discussion following in Section 5.

2. Computer aided assessment of mathematics

Until recently, automatic assessment was commonly associated with multiple choice questions (MCQ) or similar provided response question types. Such question types are referred to as *objective* because the outcome is independent of any bias by the assessor. MCQ have been criticized for many years, e.g. Hassmén & Hunt (1994), indeed Hoffmann (1962) claims they *"favour the nimble-witted, quick-reading candidates who form fast superficial judgements"* and *"penalize the student who has depth, subtlety and critical acumen"*. For mathematics MCQ are particularly problematic as the relative difficulty of a reversible process, e.g. integration compared to differentiation, is markedly altered in different directions.

> The strategic student does not answer the question as set, but checks each answer in reverse. Indeed, it might be argued that it is not just the strategic, but the *sensible* student, with an understanding of the relative difficulties of these processes, who takes this approach. This distortion subverts the intention of the teacher in setting the question, so that we are not assessing the skill we wish to assess. (Sangwin, 2013, p. 3)

This potentially reduces the validity of the question.

It is now relatively standard practice to accept answers from students which contain mathematical content and establish the mathematical properties of those answers using computer algebra. On the basis of properties established (or not) the system generates outcomes, including feedback and a score, which fulfil the purposes of formative and summative assessment. It is also standard practice to generate random versions of questions in a structured mathematical way and to automatically generate a full worked solution which reflects this randomisation. The system stores data on all attempts at one question, or by one student, for later analysis by the teacher. The goal of developing and using such software has been predominantly formative, i.e. trying to help students improve their performance on tasks. As Dunlosky et al. (2013) concluded, formative assessment (self-testing) is one of the most effective learning strategies. In some systems questions are provided in a fixed linear structure, in others the system builds an internal model of the student's strengths and weaknesses and the system adapts the subsequent choice of questions, e.g. Appleby et al. (1997). This is a very active field with a large number of parallel developments taking place. Many of the practical developments are commercial, with software tied closely to a textbook or other learning materials. Hence, current technology for automatic assessment of mathematics is disparate and the full range of available features do not appear in one software package. This is entirely understandable given the recent development of this field, and the rapid development of technology in general. We also accept that by considering contemporary technology our results will provide a snapshot of the current state of the art.

It seems to be inevitable to us that automatic assessment software will be used for summative examinations. Examinations are used as a significant component of all school mathematics curricula of which we are aware. Traditional closed book examinations also still dominate summative assessment of university mathematics. For example, Iannone & Simpson (2012) conducted a survey of mathematics departments in England and Wales. Of the 1843 individual modules they examined over one quarter were assessed entirely by closed book examination and nearly 70% used closed book examinations for at least three quarters of the final mark. Summative assessment aims to select and grade students' performance, and the results *de facto* indicate whether a student has successfully completed their studies. Indeed, there are examples where automatic assessment software is already being used for summative examinations in mathematics. For example, Ashton et al. (2006) reported trials of automatic assessments in Scottish secondary school mathematics using the SCHOLAR system[1]. In evaluating this Scottish initiative, Fiddes et al. (2002) compared paper-based tests with online examinations of high-school mathematics and concluded that "*the medium has no effect on the marks for these tests.*"

There are many systems which could have been chosen for this study as representative of contemporary CAA. The criteria for choosing representative systems include (i) the longevity of the project, (ii) evidence of widespread international use, (iii) mathematical sophistication, (iv) availability to the authors. Over the last fifteen years, during which the first author has been working in this area, many projects have developed sophisticated features and have reported pilot studies with students, see Sangwin (2013) for a recent review. Many of these pilot systems do not make the transition to mainstream beyond the initial research project, or over a medium term. We have chosen systems which are established, and which have been used beyond the developer's institution. Availability to the authors is an important practical research concern: some online assessment systems are closed, and some do not permit authoring of questions. Note, it was not our goal for this paper to undertake a detailed comparative study between systems of the extent to which examinations could be automatically marked.

2.1. Assessment of final answers

Technology to asses an algebraic expression as a final answer is well-established: see (Sangwin, 2013, chapt 8) for a recent review. Typically the student must enter an algebraic expression into a computer. The very general notion of algebraic expression includes polynomials, sets, lists, matrices, equations or systems of equations. The entry of differential equations and logical expressions is also possible in some systems. Systems vary on precisely how students enter their answer, with the most popular options being a typed linear syntax or a drag and drop equation editor.

In this paper we used the STACK online assessment system and attempted to automatically mark existing examination questions[2]. The STACK project started in 2004, and STACK questions are used in at least 8 languages and with classes of up to 1500 students. See Sangwin (2015b) for a recent survey of usage and Sangwin (2013) for more details of the design philosophy. The primary reason for choosing STACK is the mathematical sophistication in assessing final answers: the central question in this study. STACK makes full use of a computer algebra system and is able to establish a wide range of mathematical properties of the final answer.

In common with the systems STACK represents, the software seeks to establish mathematical properties specified by the teacher. That is to say, for each question the teacher must decide

[1] See `http://scholar.hw.ac.uk/` (18 November 2015).

[2] See `https://github.com/maths` for the source code (18 November 2015).

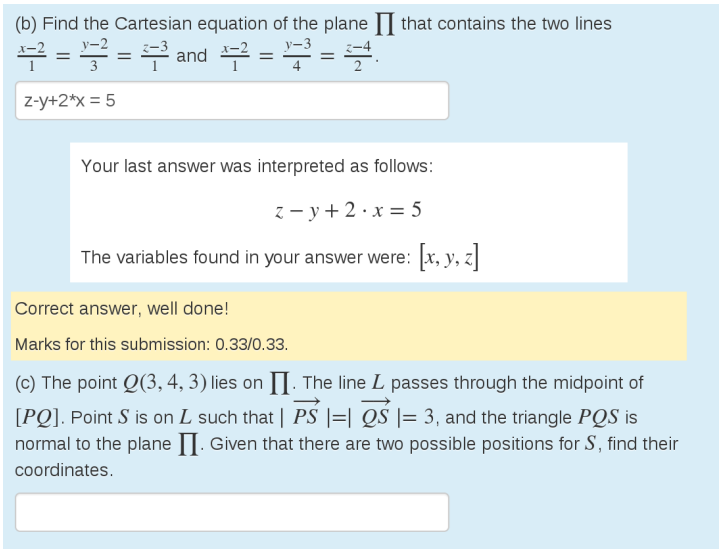


Figure 1: Example assessment of the final answer using STACK

what constitutes a correct answer, and encode these criteria as part of the question authoring process. The teacher must also decide on how much partial credit to award when only a subset of the required properties are satisfied, or if the student's answer is equivalent to that which derives from a common mistake or misconception. Assigning partial credit is a subjective value judgement, which only a human can decide. Normally the teacher seeks to establish more than one property of the final answer. The prototype mathematical properties include (i) algebraic equivalence with the correct answer and (ii) that it is written in an appropriate algebraic form, (e.g. factored). Computer algebra is able to establish a range of properties such as these. Specifying the mathematical criteria is a long way from using string matching or regular expression libraries. Computer algebra, which has an internal representation of a mathematical object, is needed to manipulate expressions and establish properties. Where the student's answer does not satisfy all the properties, the teacher is able to award partial credit and encode feedback. Potentially this is specific to the answer and directly related to possible improvement on the task. This is precisely the kind of feedback which research such as Kluger & DeNisi (1996) has suggested is most effective in a formative setting. Because the criteria are objective and specified in advance the assessment is highly reliable.

A typical assessment situation is shown in Figure 1, in which a question is shown to a student. Note, only the STACK question is illustrated here, and the other navigation elements on the web page have been excluded. The student has been asked to give an *equation* as their answer to part (b), and they have already done so. STACK separates out checking validity of their answer from establishing mathematical correctness. In this example, an *expression* $z - y + 2x - 5$ would be rejected as "invalid" as it is not an equation, rather than "wrong". Of course, it is wrong in one sense, but rejecting invalid input enables the student to have an opportunity to enter another expression. There are many reasons why an expression might be rejected as invalid, including syntax errors such a missing bracket. Once the student has a valid expression, STACK is able to establish the relevant mathematical properties of the answer. Here, the property established is that

the equation must represent the correct straight line, i.e. be algebraically equivalent to the correct answer. The precise algebraic form of the student's answer is not, in this case, relevant and the student may enter any equivalent (Cartesian) form of equation they choose. For illustrative purposes, the feedback confirming correctness is also shown in Figure 1, although this would perhaps not be available immediately to the student during an examination. There are many options for the precise form and timing of such feedback, although this is not relevant to the research reported here.

Currently STACK is able to present multiple answer boxes to the student who is expected to enter an algebraic expression into each box. In Figure 1 there are two boxes and in this example the student has chosen not to answer part (c) at this point in time. The teacher can also design a single question with multiple parts to the answer to break a lengthy process into steps, and expect the student to respond to each step. Breaking a question into pre-specified steps is quite different from genuinely assessing a student's free form working, see Ashton et al. (2006) for further comments on this issue. Typically, students do the mathematical working in a traditional manner with a pen and paper. They then enter their final answer into STACK for assessment and feedback. Students may use a calculator or CAS to aid their working, but it is the scoring of the entered result that is automatically marked. This is typical of the design of the class of software STACK has been chosen to represent, and is a serious limitation. For this reason we also consider software which automates the assessment of a process called reasoning by equivalence.

2.2. Reasoning by equivalence

Reasoning by equivalence is a particularly important algebraic activity in elementary mathematics. It is an iterative formal symbolic procedure where a term within an algebraic expression is identified and then replaced by an equivalent term. By replacing a term or sub-expression by an equivalent expression we generate a new problem having the same solutions. This is continued until a "solved" form is reached. "Solving an equation" often means transforming it into a conventional form to make the solution clear. As a specific example, to solve the equation $\log_3(x+17) - 2 = \log_3(2x)$, for $x \in \mathbb{R}$ we reason as follows.

$$
\begin{aligned}
& \log_3(x+17) - 2 && = \log_3(2x) \quad (x > 0, x > -17) \\
\Leftrightarrow\ & \log_3(x+17) - \log_3(2x) && = 2 \\
\Leftrightarrow\ & \log_3\left(\frac{x+17}{2x}\right) && = 2 \\
\Leftrightarrow\ & \frac{x+17}{2x} && = 3^2 = 9 \\
\Leftrightarrow\ & x + 17 && = 18x \\
\Leftrightarrow\ & x && = 1.
\end{aligned}
$$

As this example illustrates, at each step we either replace a term in the equation by an equivalent term, or we operate on both sides of the whole equation so that consecutive lines remain equivalent. The last line makes the solution *explicit*, i.e. it is written as $x = 1$. Note also that none of the final solutions contradict domain constraints, such as $x > 0$, which occur during the working, see Sangwin (2015a). For a large proportion of elementary mathematics reasoning by equivalence is central, or constitutes the entire task. In reasoning by equivalence we use formal symbolic replacements, so that this particular form of reasoning is itself very close to an algebraic calculation. Indeed, its similarity to calculation makes it a prime candidate for automation.

There is an important internal distinction between reasoning and argumentation.

> *Reasoning* is [...] the line of thought adopted to produce assertions and reach conclusions. *Argumentation* is the substantiation, the part of the reasoning that aims at convincing oneself or someone else that the reasoning is appropriate. (Boesen et al., 2010, p. 92)

Therefore we can have reasoning which is valid or invalid. Argumentation, in turn, may or may not correctly justify a particular step in the reasoning. Phrases such as "right method, wrong reason" are often used by teachers to describe various types of mistake.

When designing software to assess students' reasoning, there are a number of interface design decisions. For example, does a student indicate what they are intending to do in each step? Does a student choose the next step from a context-sensitive menu in the software? Does a student actually have to do what they say they will do, or does the software use CAS to calculate for the student? Does the software infer what a student has done from consecutive lines of working, and compare this with an internal model for solving this particular type of problem?

In traditional written mathematical practice it is unusual for students to explicitly spell out their intention at every step. Students mostly write the results of computations on consecutive lines, leaving the reader to infer their intentions. An example of software which facilitate reasoning by equivalence in this way is Aplusix, see Nicaud et al. (2004). Here students work line by line in a traditional way, and the system evaluates whether adjacent lines are equivalent. An alternative set of design decisions was made for MathXpert, Beeson (1992) and Beeson (1989). In this software the student indicates what they would like to do at each step, and the software undertakes the calculation for them. In some software a student only indicates they wish to "solve" an equation and the software automatically generates a full worked solution. For example, the student packages in Maple or WolframAlpha's step-by-step solving. MathExpert can also operate in this manner, and example output from MathXpert is shown in Figure 2. Software which generates a complete solution is not really suitable for assessment. These issues are discussed in more detail in, for example, Heeren & Jeuring (2014).

The recently released SOWISO system, `http://www.sowiso.nl/` (18 November 2015), appears to have a very interesting interface for reasoning by equivalence. This is a commercial system, and at the time of writing we cannot cite independent evaluations of the interface. These tutorial systems have been designed for formative assessment. They were not designed, and are not used, for examinations. That said, we predict that such technology will be modified and used for summative assessment. One purpose of this paper is to examine the extent to which answers to existing examination questions could be assessed by such software. At the time of writing STACK has prototype code for assessing reasoning by equivalence. These features are not ready for extensive use by students, but we are confident in our ability to judge whether the mathematics assessed in a particular question can be automatically assessed by an existing reasoning by equivalence engine.

3. Methodology

Mathematics, including basic statistics, is a particularly important subject both at school and university. It is a compulsory school subject. It forms a key component of all science, technology, engineering and mathematics (STEM) disciplines, and is studied at university by a wide range of other students including psychology, geography and in social sciences. We have chosen to focus

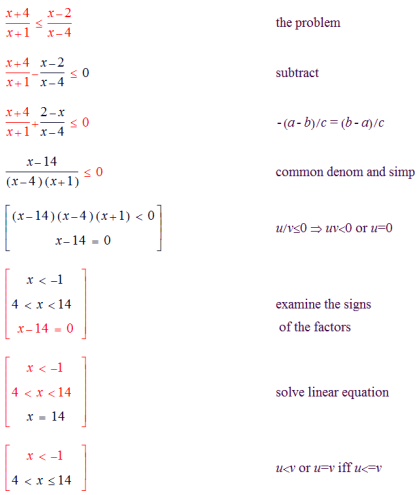


Figure 2: An autogenerated solution from the MathXpert system

on final school examinations in mathematics taken by students aged approximately 18 years old. These examinations certainly include technical mathematics, e.g. calculus, and are taken by very large numbers of students, whereas many university cohorts have fewer students.

We have selected the specimen questions on paper 1 and paper 2 for International Baccalaureate[3] (IB) Mathematics Higher level, for first examinations in 2008[4]. These questions are published online with an examination syllabus, specimen paper, and mark scheme. Details are given of how marks are allotted and in some cases alternative solutions are provided. The actual examination papers are available from `http://www.follettibstore.com/main/home` (18 November 2015) but copyright restrictions prevent us from reproducing questions or mark schemes, and so we have chosen not to use them for this research. Our corpus consists of specimen *questions* totalling 613 marks. An individual paper would comprise 120 marks, and would be allotted 2 hours. Our corpus of questions is approximately five examinations. We note that the document contains a caveat that the questions *"will not necessarily reflect balanced syllabus coverage, nor the relative importance of the syllabus topics"*. That said, the questions are published as specimens to be representative of those in real examinations and we therefore take them on face value. We have looked at the actual papers, have good reason to believe these questions are representative and hence have confidence in our results.

We believe IB Higher level mathematics is representative of core of pure mathematical topics, including algebra and calculus, which begins to be taught at school and continues as a foundation for all undergraduates in STEM degrees. All STEM students will, at some stage, learn this mathematics either at school or early in university courses. This core is common internationally, and has remained relatively stable over the last fifty years or so. This content is explicitly included in university engineering programmes, e.g. see Alpers (2013). Applications, e.g. introductory mechanics and statistics, are included in this corpus of questions. We also consider the style of the IB questions in our corpus to be representative of much current assessment practice internation-

[3]International Baccalaureate is a registered trademark of the International Baccalaureate Organization.
[4]See, for example, `http://www.math.ch/csf/mathematik/IBOtestHL08` (18 November 2015)

ally at the level of high school/university interface. The conditions under which students sit IB examinations are very traditional. Students do not have access to a Graphical Display Calculator (GDC) when answering paper 1, but such a calculator is required for paper 2 and paper 3.

3.1. Evaluating the extent to which questions can be automatically marked

The specimen document contains instructions on how to use the mark scheme, including the awarding of marks in the following categories.

M Marks awarded for attempting to use a correct Method; working must be seen.

(M) Marks awarded for Method; may be implied by correct subsequent working.

A Marks awarded for an Answer or for Accuracy: often dependent on preceding M marks.

(A) Marks awarded for an Answer or for Accuracy; may be implied by correct subsequent working.

R Marks awarded for clear Reasoning.

N Marks awarded for correct answers if no working shown.

AG Answer given in the question and so no marks are awarded.

Reading the IB specification documents it is clear to us that the IB examiners intend that students' working is equally important, if not more important, than accuracy of the final answer. This is repeated in a number of places in the instructions to markers. For example, *"As A marks are normally dependent on the preceding M mark being awarded, it is not possible to award M0 A1."*

When assessing the extent to which a question can be automatically marked we note that STACK provides a fixed structure of input boxes. Normally an input box is used for the final answer. These fixed input boxes could be used for intermediate calculations, i.e. steps in working, but currently in STACK these steps have to be specified in advance by the question author. This is a serious limitation in trying to automate the intentions of the IB examiners. STACK has been chosen as representative of systems with this design. Reasoning by equivalence software is designed to avoid this problem, allowing arbitrary lines of working. However, equivalence reasoning is only one of the types of reasoning required from students.

The mark scheme instructs examiners that *"Unless the question specifies otherwise, accept equivalent forms"*. This is precisely the task which STACK is designed to do. E.g. one answer in our corpus is $\frac{3}{\sqrt{2}} \equiv \frac{3\sqrt{2}}{2}$. Students in pure mathematics are traditionally expected to use the second form, which has no $\sqrt{}$ symbol in the denominator of a fraction. In many cases an examiner will accept both, as they are equivalent. In other situations the purpose of the question is to establish if a student is able to convert one form into another. Therefore, we need to establish that the answer is (i) equivalent to the correct answer, and (ii) in the correct form. These are the prototype properties STACK is designed to assess. In pure mathematics a floating point representation, often an approximation, is rarely acceptable.

For each question we undertook the following evaluation.

1. What form does the final answer take, and what is the syntax for entering this into STACK? We also paid attention to whether the complete answers could be captured electronically in a reasonable way. If so, can automatic assessment be envisaged for this item, subject to

Attempting to differentiate implicitly	(M1)	
$3x^2y + 2xy^2 = 2 \Rightarrow 6xy + 3x^2\frac{dy}{dx} + 2y^2 + 4xy\frac{dy}{dx} = 0$	A1	
Substituting in $x = 1$ and $y = -2$	(M1)	
$-12 + 3\frac{dy}{dx} + 8 - 8\frac{dy}{dx} = 0$	A1	
$\Rightarrow -5\frac{dy}{dx} = 4 \Rightarrow \frac{dy}{dx} = -\frac{4}{5}$	A1	
Gradient of normal is $\frac{5}{4}$.	A1	N3

Figure 3: The mark scheme for paper 1, question 31

development of a suitable interface? This is much more speculative and we address this in the final discussion section.

2. Can the question be automatically marked completely with STACK? All marks must assigned exactly as in the mark scheme.
3. Can we assess the final answer(s) automatically and completely with STACK? To quantify this we looked at the number of accuracy marks "A" awarded by the mark scheme. We did not include method marks. However, where the mark scheme permits implied method marks (i.e. "(M1)" marks) then implied method marks are included as being awarded by STACK for a correct answer. This parallels the process of marking by hand, and so we believe counting implied method marks is faithful to the intentions of the examiners. Note that we have only included (M1)A1 where implied method can be inferred from a correct answer, not (M1)M1 where the implied method marks depends on subsequent method marks. (In fact, there were only 7 (M1) marks of the form (M1)M1 so that most implied method marks depend on subsequent accuracy marks). This is a strict interpretation which does not simply count types of marks. Note that while implied method marks can be awarded automatically for a correct answer, partial credit cannot be awarded automatically for a correct method in the absence of a correct answer.
4. How many marks are available for *reasoning by equivalence*?

As a specific example we consider Q31 of paper 1. *"Find the gradient of the normal to the curve* $3x^2y + 2xy^2 = 2$ *at the point* $(1, -2)$*"*. The official mark scheme for this question is shown in Figure 3. The final answer is the rational number $\frac{5}{4}$ and the syntax for entering this into STACK is `5/4`. This question cannot be automatically assessed entirely with STACK. The mathematical properties of the final answer, however, can be established fully. Whether to reject equivalent forms such as $1 + 1/4$ and 1.25 is a decision for the examiner: they can be accepted or rejected in any combination with any chosen partial marks. Given both of the method marks in this question are implied method marks, we give a score of 3 for the final answer as assessed by STACK. We note that N3 indicates that 3 marks are available for the final answer regardless of whether any method is shown, further supporting the decision that STACK can award 3 marks for the correctness of a final answer. This is exceptional, as most method marks are not implied. In this question we judge that three marks are available for reasoning by equivalence. One implied method mark is awarded for the substitution, then then two accuracy marks for solving the equation and finding $\frac{dy}{dx} = -\frac{4}{5}$.

	M	(M)	A	(A)	R	(R)	N
# of marks	105	99	391	35	19	6	(100)
%	16	15	60	5	3	1	(15)

Table 1: Descriptive statistics of marks available on IB sample questions

3.2. Development of STACK questions

The first author, with extensive experience of designing and using STACK, worked through the examination papers and mark schemes to evaluate the extent to which STACK could be used to automatically mark the exam. To test these findings we automatically marked all the questions in paper 1 as far as possible in STACK. We therefore know what mathematical properties each of the answers should satisfy (e.g. equivalence with a correct answer) and the extent to which we are able to automatically establish them. We are also confident in our ability to correctly identify reasoning by equivalence. Where this was in doubt, we used either an existing computer algebra system, or software such as MathXpert, to automatically generate the steps in the working as proof that these steps can be automatically marked with software currently available. This demonstrates the solution process can be automatically marked in a stepwise fashion. We did not automatically mark the assessment of individual reasoning fragments.

4. Results

4.1. Marks available for specimen questions

As background information we record the distribution of marks available for specimen questions. Paper 1 contains 55 questions in two sections, and paper 2 a further 10 questions. These are broken down into 142 separate question parts for which marks are allocated separately, giving a total of 613 marks available. Some questions have alternative mark schemes, with different allocations of marks. Both schemes have been included with equal weight, so that the total number of marks considered is 655.

The distribution of marks between method, accuracy, etc. available for the specimen questions is shown in Table 1. Note that nearly 65% of marks are available for accuracy and that only 4% of marks are awarded for reasoning, both explicit and implied. 15% of the marks are awarded for correct answers if no working is shown (N). Since these marks are awarded in parallel to other A and M marks they have not been included in totals to generate percentages. It would appear from this table that accuracy is of primary importance. It should be reiterated here that accuracy "A" marks are only available where there is evidence that the appropriate method has been used, so they implicitly reflect the importance of method. 60% of marks are are awarded for accuracy only where there is evidence of an appropriate method. Method marks on their own are, for example, for knowing which method is appropriate rather than for actually being able to use the method accurately.

4.2. Extent to which questions could be automatically marked

Our analysis of these questions sought to determine the extent to which answers could be automatically assessed. Recall that this was done in three levels as follows.

1. Marks could be awarded by STACK *exactly* as in the mark scheme.

	# marks	
(i) Awarded by STACK *exactly*	110	18%
(ii) Final answers and implied method marks	226	37%
(iii) Reasoning by equivalence	217	36%
Total of max of (ii) and (iii) per question	374	61%

Table 2: The extent to which IB specimen questions can be automatically assessed

2. Marks for final answers, i.e. A, which can be awarded by STACK together with implied method marks.
3. Marks which could be awarded using a "reasoning by equivalence" engine.

The results are shown in Table 2. Note that 18% of marks can be awarded by STACK *exactly as specified in the mark scheme*. If we relax the requirement that method must be seen, and count the award of accuracy/implied method marks for the assessment of final answers then 37% can be awarded using the criteria set in the scheme. A reasoning by equivalence engine would be capable of awarding 36% of the marks which is a substantial proportion of the method marks and subsequent accuracy marks. When, for each question, we take the maximum marks available for the final answer and implied method and assuming we are able to implement reasoning by equivalence, 61% of the marking can be automated. This arithmetic initially looks odd, but some implied method marks are for reasoning by equivalence, so this is not a simple addition of marks.

There were clear differences in the extent to which we could automatically mark questions between mathematical subjects. Algebra and pure mathematics, including calculus, functions and inequalities, were least problematic. Applied topics were more problematic, but these often rely on core pure techniques such as algebra and calculus. Whether the question was difficult to automatically mark, however, depended on the precise form of the answer and the steps in the working more than on a particular subject area.

4.3. Entry of answers into STACK

The corpus of questions had 142 question parts. 16 parts used the code AG, indicating that the final answer is given, and that a student is expected to "show" or "prove". A 4 further questions used "show that" without assigning the AG code in the mark scheme. 9 parts require a graphical solution, sometimes in combination with an algebraic expression. There were a further 4 parts where we judged entry of the answer would be infeasible, including two proofs by induction. Ultimately we had 119 (85%) question parts were the final answer could be written as an algebraic expression in STACK syntax.

The answer to 16 parts were integers, and 58 some kind of number including floating point, rational, surds or numerical expressions such as $\frac{\pi}{3}$, with 12 answers containing complex numbers. A further 24 answers were sets or lists. These include sets of solutions (numbers) or lists such as $[a = 55, b = 75]$, where the answer really consists of two parts. Such examples would best be assessed as a multipart question, and are essentially numerical. The answer in 13 cases used coordinates. Inequalities, including chained inequalities such as $x < -1 \vee (4 < x \leq 14)$, were required for 6 questions and only 17 answers were an algebraic expression which contained a variable, such as $10\sin(5x)$ or $v^2/2 = \log(x^2 + 1) - \log(2) + 2$, which is surprisingly few.

Only one question required a matrix as an answer, and this is relatively straightforward to accommodate with an on-screen grid into which the student can type their individual expressions.

In STACK, currently, this is a fixed size but the interface in previous versions enabled the student to specify the size and provided a grid using the DragMath (see Sangwin (2012)) applet.

The answers to two questions were an infinite set, or infinite series. Input of these mathematical objects is problematic, see van der Hoeven (2015). For example, Q44B(di), used the ellipsis operator $\cdots$ to indicate the continuation of an infinite series

$$Q17.\ S_\infty = \left(\cos(\theta) + \frac{1}{2}\cos(2\theta) + \frac{1}{4}\cos(3\theta) + \cdots\right) + i\left(\sin(\theta) + \frac{1}{2}\sin(2\theta) + \frac{1}{4}\sin(3\theta) + \cdots\right).$$

The complexity of this expression is exceptional, but the underlying learning objective could readily be assessed by asking the student for the first 3 or so terms in this sum. For another question, Q45(g), the answer was all multiples of three, which is traditionally written as a set such as $\{3n, n \in \mathbb{N}\}$. An alternative here would be to assess the English sentence "This is real when n is a multiple of 3" using the kind of technology described by Butcher & Jordan (2010). Since this question asks students to *"Find the set of values of n for which α^n is real."*, another option would be to ask student to replace the ? in $\{?, n \in \mathbb{N}\}$ with an appropriate algebraic expression picking out the values required. This could easily be assessed by STACK, and illustrates where a slight reformulation of the question facilitates assessment without appearing to significantly change the intended learning outcomes or difficulty of the question itself. However, we fully acknowledge that subtle differences in the phrasing of a question can, and do, have profound and sometimes unexpected effects on the actual difficulty of the question. Note, for the purposes of this paper we did not reformulate the question.

Compared to the complexity of expressions routinely entered by students to questions in Higher Education, the entry of answers to questions in our corpus is straightforward, surprisingly so. Of course, students will still need to learn the required syntax if they are to make use of a system for assessment. Also, computer algebra would still be needed in many cases to establish the equivalence of surd expressions, e.g. $\frac{1}{\sqrt{3}} = \frac{\sqrt{3}}{3}$ so that a simple string match on the digits would not be adequate in the vast majority of cases.

Entering a complete mathematical argument, as required when reasoning by equivalence, is much more difficult. MathXpert avoided the problem by performing calculations for students, only requiring them to select which move to perform from a context sensitive menu of rules which could be applied to the current expression. Requiring students to type in their complete chain of reasoning, without a very efficient interface, is likely to disrupt their train of thought substantially. This remains an important unsolved human computer interface problem for mathematicians in general, not just for assessment, see van der Hoeven (2015).

4.4. Reasoning by equivalence

To further illustrate the potential of reasoning by equivalence to automatically mark examinations, we choose paper 2, question 2.

> Let $f(x) = \frac{x+4}{x+1}$, $x \neq -1$ and $g(x) = \frac{x-2}{x-4}$, $x \neq 4$. Find the set of values of x such that $f(x) \leq g(x)$.

This question has two mark schemes listed. The first relies on a graphical solution, which will be difficult to automatically mark in the foreseeable future. The second method relies on reasoning by equivalence. The first step is to set up the problem algebraically. The mark scheme is reproduced in figure 4.

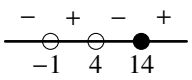
$\frac{x+4}{x+1} - \frac{x-2}{x-4} \leq 0$ M1

$\frac{x^2-16-x^2+x+2}{(x+1)(x-4)} \leq 0$

$\frac{x-14}{(x+1)(x-4)} \leq 0$ A1

Critical value of $x = 14$. A1

Other critical values at $x = -1$ and $x = 4$ A1



$x < -1$ or $4 < x \leq 14$ A1A1

Note: Each value and inequality sign must be correct. [6 marks]

Figure 4: The mark scheme for paper 2, question 2

An automatically generated solution to this problem is shown in Figure 2. This has been generated by simply typing in the question and having the MathXpert system, Beeson (1998), derive a complete and correct solution. Note the striking similarity between the two. MathXpert can be used in a number of ways, e.g. it has options to provide users with hints, to show the next step automatically or even to complete the whole problem as shown here. The basic design interface is for students to choose the next "rule" and let the software actually perform the calculation. A combination of this design with that of STACK to force the user to do the calculation stated and then subsequently actually confirm that the step has been undertaken correctly by the student would enable the assessment of a wide range of reasoning by equivalence problems. Since MathXpert enables multiple solution paths a very wide range of correct solutions can be evaluated this approach.

5. Discussion

Our results show that transcribing existing paper-based mathematics examinations into an electronic format is now feasible for a significant proportion of the questions as currently assessed. The most significant barrier to faithful automation of the current mark scheme is the requirement for evidence of an appropriate method, rather than inferring which method has been used from the student's final answer. In traditional practice students do not indicate their explicit reasons, rather they work line by line. This form of reasoning can be automatically be assessed for a wide variety of questions actually assessed.

It would be entirely appropriate when using CAA in a practical exam to write questions with the format in mind, and tailor the question to some extent to take account of the constraints of the format. This is already done for multiple choice questions. The choice of a paper based examination also influences which questions are chosen and how they are phrased, e.g. questions with non-unique correct answers which require the examiner to complete a significant computation to assess them are never set on paper. For some examples see Sangwin (2003). In particular, follow through marking is a feature of many paper based examinations and is an artefact of the format. Immediate feedback during an examination online could reduce the need for follow through

marking. The extent to which immediate feedback should be provided during an examination remains a matter for debate.

The working shown in Figure 3 illustrates how reasoning by equivalence pervades mathematics at this level. In this calculus question, part of the solution process involves setting up and solving a simple linear equation in a single unknown. Although the unknown quantity in this case is a differential quotient, this poses no serious challenge to reasoning by equivalence engines.

Our pilot methodology was confined to software which assessed a final answer, i.e. the class of software which STACK represents. It was during this pilot phase that we became aware of how important reasoning by equivalence is at this level and hence revised our approach to evaluate the extent to which assessment of reasoning by equivalence can be automatically marked. In choosing our definition of reasoning by equivalence we have not included any calculus operations, but stayed confined only to algebraic operations. Differentiating both sides of an equation retains equivalence of algebraic expressions representing the functions. I.e.

If $f(x) = g(x)$ then $f'(x) = g'(x)$.

However this is not an equivalence when reasoning. I.e. $f(x) = g(x) \leftrightarrow f'(x) = g'(x)$ is false. All we can say is that

If $f'(x) = g'(x)$ then $f(x) = g(x) + c$, for some constant c.

We can easily envisage a reasoning engine with much wider capabilities. These expanded capabilities would certainly include implications which arise from differentiating both sides of an equation. CAS supported systems can certainly establish if two consecutive lines of working arise by differentiating or integrating one line correctly with respect to a particular variable. So, technology already exists to establish correctness of student's calculus operations in free form working. For the purposes of this research our criterion is algebraic equivalence which is both clear and one which has already been implemented in software.

As another example, some of the questions which ask students to "show that" or "verify" could be rephrased as "find", without problematic input or assessment of the final answer. Much of the intermediate working is simply equivalence reasoning, and this could be assessed. Q17(a) is one example where reasoning by equivalence could be used effectively to replace a "show".

Let $\sin(x) = s$.

(a) Show that the equation $4\cos(2x) + 3\sin(x)\text{cosec}^3(x) + 6 = 0$ can be expressed as $8s^4 - 10s^2 + 3 = 0$.
(b) Hence solve the equation for x in the interval $[0, \pi]$.

There were numerous examples where such minor linguistic switches could have been made in the phrasing of the question. Minor rephrasing would significantly raise the proportion of questions where the underlying learning objective can be successfully assessed automatically using the technology described in this paper. We note that the phrasing used by the IB examiners in this example reduces the need for follow through marking. Changing the wording from "show" to "find" potentially significantly increases the difficulty. Furthermore, in this example a subsequent part follows, and so failure to get the correct answer to the first part might render the second part impossible, or seriously mislead a student and make follow through marking much more problematic. Immediate feedback during an exam may be appropriate here but such proposals probably need trials with students.

In the existing scheme, reasoning by equivalence attracts method (M) marks rather than reasoning (R) marks. There were few marks (25 = 4%) awarded for reasoning, or implied reasoning, in the mark scheme, which surprised the authors. Reasoning at this level was confined to simple individual steps. E.g. congruence of geometric angles, induction proofs, conclusions about roots (e.g. $2 - i$ is a root of a real polynomial so $2 + i$ must also be a root), and using the second derivative to reason about the nature of local maxima and minima. We reiterate that much of what is currently examined is actually reasoning by equivalence, which is a combination of logic and algebraic calculation.

In many induction proofs the central induction step relies on reasoning by equivalence only. The logical incantations necessary to create a valid proof by induction are somewhat formulaic. On this basis we suggest that assessment of student's attempts at proof by induction might be a prime candidate for automatic assessment in the near future.

We note that our analysis is confined to the level of individual questions. In the mark scheme, accuracy errors which arise from rounding or specifying the wrong level of accuracy should only be penalised once on the *paper*. STACK does not communicate information from one question to another, so that a student would potentially be penalised for every accuracy mistake and this is not the intention of the examiners. Experimental STACK code will store some "state", potentially allowing a model of the student to be created. This will enable information to be communicated from one question to another in the future.

Our research is based purely on the examination questions and the accompanying mark scheme, and not analysis of students' answers. Human examiners are able to deal fairly with non-conventional responses. STACK was designed primarily for formative feedback purposes, not for examinations. For immediate formative feedback all design decisions need to be made in advance. In an examination where feedback is delayed, a human could review all answers and how they have been assessed to look for anomalous results. In STACK it is possible to re-grade particular questions with an updated mark scheme as required. None of this removes the need for human intervention, but it certainly facilitates reliable implementation of decisions across all students, which is difficult in traditional paper based examinations.

Although we have not focused on general learning outcomes, it is appropriate to look at this in our general discussion. The syllabus specifies the general learning outcomes defined in Table 3. These outcomes are not based on particular mathematical content, but the general use of mathematical techniques and thinking. Outcomes 1, 3, 4, and 5 are clearly regularly assessed in the question corpus. Others appear to be assessed far less often. Outcomes which we judge are rarely addressed by questions in our corpus include graphical representation, pattern recognition, and modelling. These questions are precisely those which are very difficult to automatically mark currently. It is difficult to envisage how a freehand sketch can be assessed automatically against objective criteria, yet assessment of this is an explicit objective, see table 3, objective 0.2. Similarly, we judged that objective 0.7 "Recognize patterns and structures in a variety of situations, and make generalizations" was assessed explicitly by very few questions in the corpus. Structure and pattern lies at the heart of algebraic reasoning, and is a necessary part of algebraic decision making. For example equating real and imaginary parts to generate simultaneous equations is structural, and used regularly. That said, very few questions we examined were *explicit* in seeking to assess structure or pattern as the goal of the question. We note that the assessment objectives relate to the whole course which includes course work in the form of an internal assessment. Since we have not included this in our analysis we are not drawing conclusions about the specification as a whole, just the specimen examination questions.

It was striking in our analysis that the learning objectives which are hard to automatically

	Learning outcome
0.1:	Read, interpret and solve a given problem using appropriate mathematical terms
0.2:	Organize and present information and data in tabular, graphical and/or diagrammatic forms
0.3:	Know and use appropriate notation and terminology
0.4:	Formulate a mathematical argument and communicate it clearly
0.5:	Select and use appropriate mathematical strategies and techniques
0.6:	Demonstrate an understanding of both the significance and the reasonableness of results.
0.7:	Recognize patterns and structures in a variety of situations, and make generalizations
0.8:	recognize and demonstrate an understanding of the practical applications of mathematics (10 times)
0.9:	Use appropriate technological devices as mathematical tools.
0.10:	Demonstrate an understanding of and the appropriate use of mathematical modelling

Table 3: General learning outcomes

assess are not regularly tested by the questions in our corpus. Reasoning marks appear to be available only when students are asked for a specific reason for their answer or a reasoning step in their solution, e.g. accepting or rejecting a hypothesis. However, there are forms of assessment, such as comparative judgement, which may also be useful for items which do not have objective criteria. See e.g. Pollitt (2012), Jones et al. (2014). Short answer questions are also being used with accuracy rates which exceed those of humans in some science assessment, e.g. Butcher & Jordan (2010). The convergence of technology such as comparative judgement and short answer offer to complement the specific mathematical assessment technology examined by our research. Such a combination of question types may be able to offer a rounded assessment experience to students which reliance on a single question type such as MCQ cannot.

We have taken the corpus of IB questions as representative of mathematics assessments at this level. Existing online assessment systems, combined with reasoning by equivalence software, can automatically mark the assessment of a significant proportion of current work, as actually tested by existing questions. The criteria required of a correct answer for a significant proportion of final answers can be established objectively and reliably. We fully expect this technology will be available and used for high stakes mathematics examinations in a wide variety of settings in the near future.

Alpers, B. (2013). *A Framework for Mathematics Curricula in Engineering Education: A Report of the Mathematics Working Group*. Technical Report SEFI Mathematics Working Group. ISBN 978-2-87352-007-6.

Appleby, J., Samuels, P. C., & Jones, T. T. (1997). DIAGNOSYS – a knowledge-based diagnostic test of basic mathematical skills. *Computers in Education*, *28*, 113–131.

Ashton, H. S., Beevers, C. E., Korabinski, A. A., & Youngson, M. A. (2006). Incorporating partial credit in computer-aided assessment of mathematics in secondary education. *British Journal of Educational Technology*, *27*, 93–119.

Beeson, M. (1989). Computers and mathematics. chapter Logic and computation in Mathpert: An expert system for learning mathematics. (pp. 202–214). Springer-Verlag.

Beeson, M. (1992). Mathpert: Computer support for learning algebra, trig, and calculus. In *Logic Programming and*

Automated Reasoning: Proceedings of the International Conference LPAR '92 St. Petersburg, Russia, July 15–20 (pp. 454–456). Springer volume 624.

Beeson, M. (1998). Computer-human interaction in symbolic computation. chapter Design Principles of Mathpert: Software to support education in algebra and calculus. (pp. 89–115). Springer-Verlag.

Biggs, J., & Tang, C. (2011). *Teaching for Quality Learning at University*. (4th ed.). Open University Press.

Boesen, J., Lithner, J., & Palm, T. (2010). The relation between types of assessment tasks and the mathematical reasoning students use. *Educational Studies in Mathematics*, *75*, 89–105.

Burkhardt, H., & Swan, M. (2012). Designing assessment of performance in mathematics. *Educational Designer: Journal of the International Society for Design and Development in Education*, *1*.

Butcher, P. G., & Jordan, S. E. (2010). A comparison of human and computer marking of short free-text student responses. *Computers and Education*, *55*, 489–499.

Conole, G., & Warburton, B. (2005). A review of computer-assisted assessment. *Research in Learning Technology*, *13*, 17–31.

Dunlosky, J., Rawson, K. A., Marsh, E. J., Natan, M. J., & Willingham, D. T. (2013). Improving students' learning with effective learning techniques: Promising directions from cognitive and educational psychology. *Psychological Science in the Public Interest*, *14*, 4–58.

Fiddes, D. J., Korabinski, A. A., McGuire, G. R., Youngson, M. A., & McMillan, D. (2002). Does the mode of delivery affect mathematics examination results? *Alt-J*, *10*, 60–69.

Galbraith, P., & Haines, C. (1998). Disentangling the nexus: Attitudes to mathematics and technology in a computer learning environment. *Educational Studies in Mathematics*, *36*, 275–290.

Hassmén, P., & Hunt, D. P. (1994). Human self-assessment in multiple choice. *Journal of Educational Measurement*, *31*, 149–160.

Heeren, B., & Jeuring, J. (2014). Feedback services for stepwise exercises. *Science of Computer Programming*, *88*, 110–129.

van der Hoeven, J. (2015). Towards semantic mathematical editing. *Journal of Symbolic Computation*, .

Hoffmann, B. (1962). *The tyranny of testing*. Crowell-Collier.

Iannone, P., & Simpson, A. (2012). *Mapping University Mathematics Assessment Practices*. University of East Anglia.

Jones, I., Swan, M., & Pollitt, A. (2014). Assessing mathematical problem solving using comparative judgement. *International Journal of Science and Mathematics Education*, *13*, 151–177.

Kilpatrick, J., Swafford, J., & Findell, B. (2001). *Adding it up: helping children learn mathematics*. National Academy Press, Washinton D.C.

Kluger, A. N., & DeNisi, A. (1996). Effects of feedback intervention on performance: A historical review, a meta-analysis, and a preliminary feedback intervention theory. *Psychological Bulletin*, *119*, 254–284.

Nicaud, J. F., Bouhineau, D., & Chaachoua, H. (2004). Mixing microworlds and CAS features in building computer systems that help students learn algebra. *International Journal of Computers for Mathematical Learning*, *9*, 169–211.

Pollitt, A. (2012). The method of adaptive comparative judgement. *Assessment in Education: Principles, Policy & Practice*, *19*, 281–300.

Ridgeway, J., McCusker, S., & Pead, D. (2004). *Literature Review of E–assessment*. Futurelab Series 10 Futurelab. ISBN: 0-9544695-8-5.

Sangwin, C. J. (2003). New opportunities for encouraging higher level mathematical learning by creative use of emerging computer aided assessment. *International Journal of Mathematical Education in Science and Technology*, *34*, 813–829.

Sangwin, C. J. (2012). The DragMath equation editor. *MSOR Network Connections*, *12*, 5–8.

Sangwin, C. J. (2013). *Computer Aided Assessment of Mathematics*. Oxford University Press.

Sangwin, C. J. (2015a). An audited elementary algebra. *The Mathematical Gazette*, *99*, 290–297.

Sangwin, C. J. (2015b). *Who uses STACK? A report on the use of the STACK CAA system*. Technical Report Loughborough University.

Sleeman, D., & Brown, J. S. (Eds.) (1982). *Intelligent Tutoring Systems*. Academic Press.