



**Research Memorandum**  
ETS RM-11-12

**Automated Scoring of CBAL  
Mathematics Tasks With m-rater**

---

**James H. Fife**

**September 2011**

**Automated Scoring of CBAL Mathematics Tasks With m-rater**

James H. Fife  
ETS, Princeton, New Jersey

September 2011

As part of its nonprofit mission, ETS conducts and disseminates the results of research to advance quality and equity in education and assessment for the benefit of ETS's constituents and the field.

To obtain a PDF or a print copy of a report, please visit:

<http://www.ets.org/research/contact.html>

**Technical Review Editor:** Jim Carlson

**Technical Reviewers:** Liz Marquez and John Blackmore

Copyright © 2011 by Educational Testing Service. All rights reserved.

E-RATER, ETS, the ETS logo, and LISTENING. LEARNING. LEADING. are registered trademarks of Educational Testing Service (ETS). M-RATER is a trademark of ETS.



## **Abstract**

For the past several years, ETS has been engaged in a research project known as Cognitively Based Assessment of, for, and as Learning (CBAL). The goal of this project is to develop a research-based assessment system that provides accountability testing and formative testing in an environment that is a worthwhile learning experience in and of itself. An important feature of the assessments in this system is that they are computer-delivered, with as many of the tasks as possible scored automatically. For the automated scoring of mathematics items, ETS has developed m-rater scoring engine. In the present report, I discuss the m-rater-related automated scoring work done in CBAL Mathematics in 2009. Scoring models were written for 16 tasks. These models were written in Alchemist, a software tool originally developed for the writing of c-rater™ scoring models (c-rater is ETS's scoring engine for scoring short text responses for content). In 2009 the c-rater support team completed a collection of enhancements to Alchemist that enables the user to write m-rater scoring models. This collection of enhancements is known as KeyBuilder. In 2009 I reviewed the literature to see to what extent problem solutions that are expressed in the form of a structured sequence of equations can be automatically evaluated.

Key words: m-rater, mathematics tasks, human-scored responses, scoring models, automated scoring, solution steps, CBAL, KeyBuilder, conditionally scored, Alchemist

## **Acknowledgments**

This paper describes work done by a number of ETS staff members. In particular, John Blackmore and Mike Wagner solved many of the problems raised by the need for conditional scoring, Eleanor Bolge wrote the modifications to the LEP key that is described in the paper, Dennis Quardt wrote the original KeyBuilder application and the generic graph key, and Margaret Redman formatted the CBAL response data for m-rater scoring.

The author would also like to thank John Blackmore, James Carlson, and Elizabeth Marquez for their helpful comments on an earlier version of the manuscript.

For the past several years, ETS has been engaged in a research project known as Cognitively Based Assessment of, for, and as Learning (CBAL). The goal of this project is to develop a research-based assessment system that provides accountability testing (assessment *of* learning) and formative testing (assessment *for* learning) in an environment that is a worthwhile learning experience in and of itself (assessment *as* learning; Bennett & Gitomer, 2009). Assessments are being developed in mathematics, reading, and writing. One feature of the project is that accountability assessments will be administered periodically during the course of the year instead of all at once, at the end of the year; these assessments are called, reasonably enough, periodic accountability assessments (PAAs). An important feature of these assessments is that they are computer-delivered, with as many of the tasks as possible scored automatically. For the automated scoring of mathematics items, ETS has developed the m-rater scoring engine (Bennett, Morley, & Quardt, 2000). M-rater can score items for which the response is an equation or a graph.

Graf, Harris, Marquez, Fife, and Redman (2010) described some of the early work involved with the mathematics component of CBAL. In the present report, I discuss the m-rater-related automated scoring work done in CBAL Mathematics in 2009. In CBAL Mathematics, two PAAs were administered in Maine in spring 2009—one in grade 7 and one in grade 8—and two in the fall. The original plan had been to score all four PAAs in 2009. However, at the request of the participating Maine teachers, the fall PAAs were administered too late in the year to be scored before the end of the year. These PAAs were scored at the beginning of 2010. Of the tasks in the spring PAAs, 16 were suitable for automated scoring with m-rater. Scoring models for these tasks were written in Alchemist, the online software tool in which content specialists can quickly write scoring models for most m-rater-scored mathematics items. The models were then used to score the student responses, and the scores were validated.

When the response to a task is an equation or a graph, m-rater can score the response as right or wrong with close to 100% accuracy (Bennett et al., 2000). M-rater can also identify features of the response and assign a partial-credit score based on the presence or absence of certain features. Several CBAL tasks have been scored with partial-credit rubrics in this way. But when the response to a task is a worked-out solution to a problem, m-rater cannot evaluate the solution to determine which parts of it are correct or represent valid arguments. But if the student provides a solution in the form of a structured sequence of equations, it may be possible to use

m-rater to evaluate the sequence and determine the extent to which the sequence represents a correct solution to the problem. In 2009 I reviewed the literature to see what, if anything, has already been done along these lines.

Alchemist was originally developed as a tool for the writing of c-rater™ scoring models (c-rater is ETS's scoring engine for scoring short text responses for content). The collection of enhancements to Alchemist that enables the user to write m-rater scoring models is known as KeyBuilder. The construction of KeyBuilder was completed in 2009. This report begins with a brief summary of the history of KeyBuilder. The report then continues with a detailed discussion of the different types of scoring models for CBAL tasks that were written in 2009, the challenges that were present with each type, and how those challenges were successfully dealt with. The report also includes a summary of the results of a literature review of automated scoring of solution steps. A final discussion indicates areas for further research.

### **KeyBuilder**

Prior to the development of KeyBuilder, the scoring key for an m-rater item was a complex piece of code that was handwritten by a senior programmer, based on verbal or other instructions from the item author and/or the test developer. Keys normally required at least eight hours to write and could not be created or reviewed by content experts. Several years ago, ETS began developing an application called KeyBuilder, whose purpose was to enable test developers and other content specialists to write for themselves the scoring keys for most constructed response mathematics items to be scored by m-rater. The application was to have two components:

- An executable file that would take as input a text file containing the necessary parameters and would produce as output an m-rater key
- A graphic user interface (GUI) into which test developers would enter the required parameters and which would generate the text file.

The executable file was written several years ago. The GUI was designed in 2006 and was to have been built in 2007. However, toward the end of 2006, research scientists at ETS decided to integrate m-rater with c-rater, thereby solving two of m-rater's shortcomings—(a) because there was no banking system, m-rater keys were stored on a local machine or shared drive, and (b) data files were prepared from student responses and batch-scored on the local

machine. This integration was accomplished during the first half of 2007. As a result, scoring keys for m-rater items are now embedded in c-rater scoring models and are banked with other c-rater scoring models.

The m-rater/c-rater integration led to a redesign of the entire KeyBuilder application. M-rater scoring models are now written in Alchemist. Enhancements to Alchemist enable it to accommodate m-rater scoring models; the term *KeyBuilder* is now applied collectively to these enhancements.

For equation items (items whose response is an equation or expression), there is a small number of generic m-rater keys that score most of the tasks that appear on mathematics assessments. (The target was 80% of the tasks; for tasks that cannot be scored by one of the generic keys, either a task-specific handcrafted key or a new generic key is written.) To build a scoring model for a specific item, a content specialist uses Alchemist to enter a model (correct) equation, select the appropriate generic key, and indicate the features of the response that are to be scored. KeyBuilder for equations was completed in 2008, and, for the most part, I was able to use it to quickly write the m-rater scoring models for the equation items. (After the preliminary work is done, a typical model requires about 10 minutes to write in Alchemist.)

For graph items (items whose response is a graph), the model author constructs model sentences that match important concepts (e.g., “The slope of the line is 3”) using a collection of about 25 graph functions—functions that take as input one or more graphical elements of the examinee’s response and return a feature of the graphical elements (e.g., the slope of a line) or a Yes/No response based on features of the graphical elements (e.g., “Are Line 1 and Line 2 parallel?”). The responses are then scored by a single generic graph key based on the original KeyBuilder executable file. The necessary enhancements to Alchemist to support model sentence construction for graph items had not been completed by the time the CBAL items were scored. (These enhancements were completed in November 2009.) The generic graph key had been completed, however. As a result, with help from technical staff, I was able to build scoring models for graph items.

### **Writing and Validating M-rater Scoring Models**

ETS has a standard procedure for validating automated scoring models for the c-rater and e-rater® scoring engines, based on the extent to which the automated scores agree with human scores. First, a set of student responses is double human scored. Then the responses are divided



into two subsets, the development set and the validation set. The development set is used by the model builder to build the model. After the model has been completed, the validation set is scored, and the automated scores are compared with the human scores, using the following statistical measures:

- The mean and the standard deviation for each human rater and for the automated scoring engine
- The standardized difference between the means—the difference in the means divided by the standard deviation of the (first) human rater
- Unweighted kappa
- Quadratic-weighted kappa
- Pearson correlation
- Absolute and adjacent agreement rates

These statistics must meet the following conditions for the model to be considered validated (the first three conditions apply to both the human/human and the human/automated statistics):

- The standardized difference must not be greater than 0.15 in absolute value.
- The quadratic-weighted kappa must not be less than 0.7.
- The correlation must not be less than 0.7.
- The difference between the human/human quadratic-weighted kappa and the human/automated quadratic-weighted kappa must not be greater than 0.1.

This procedure has worked well for c-rater and e-rater, but does not seem particularly relevant for m-rater. Since the scoring of equations and graphs is completely objective, the reliability of the automated scores can be measured against an absolute standard of right and wrong instead of the relative standard of whether or not the human score is matched. Furthermore, since m-rater should be able to score mathematics responses with 100% accuracy, one would expect complete agreement between m-rater scores and human scores. Finally, a set of scored responses is not required to build a scoring model, unless one wants the m-rater scoring model to identify common incorrect responses.

This means that m-rater scoring models can be written without human scored student responses (as was done in CBAL Mathematics). It also means that the validation process for m-rater is both simpler and more rigorous than the validation process for c-rater and e-rater. The m-rater validation process requires only a single human score, which is compared to the m-rater score; there is no need to calculate the various statistical measures. But the validation process is more rigorous in that there is no tolerance for disagreement; if the human score and the m-rater score are different for even a single response, the two scores must be examined to determine the source of the discrepancy.

As a result of these considerations, the following procedure has been developed for writing and validating m-rater scoring models.

1. The scoring rubrics are carefully reviewed for clarity and tightness.
2. Simulated responses are written for each model response at each score point.
3. The scoring model is written, using the simulated scored responses to debug the model.
4. The scoring model is used to score a set of student responses.
5. The scored student responses are copied to a spreadsheet, and duplicate responses are eliminated.
6. The distinct responses are human-scored (using spreadsheet functions when possible).
7. The human scores are compared with the m-rater scores. If there are any discrepancies between the human and the m-rater scores, these responses are examined in detail to determine the source of the discrepancy.

(This procedure is being re-evaluated in 2011 and will be revised as appropriate, based on experience with the automated scoring of CBAL items.)

Occasionally, there are a set of student responses that have already been double human-scored. In 2009 none of the tasks scored by m-rater were human scored before being automatically scored, but in 2008 the m-rater tasks were human-scored before m-rater scoring models were written. The CBAL team followed the above procedures and then compared the m-rater scores with each set of human scores, looking at any discrepancies in detail. The details of these findings are in Fife (2009). For the most part, when there was a discrepancy between the

m-rater score and one of the human scores, the m-rater score was correct and the human score was incorrect.

### CBAL M-rater Mathematics Tasks

Sixteen tasks from the spring 2009 PAAs were suitable for scoring by m-rater. (See Figure 1.) Thirteen of these tasks were equation tasks and three were graph tasks. Of the 13 equation tasks, four were conditionally scored—that is, each student’s response was scored conditionally on that student’s response to one or more previous tasks. For example, in the task Carnival 2 (see Figure 2), the student is shown the graph of a line; in part *a* the student is asked to indicate the slope of the line, in part *b* the student is asked to indicate the *y*-intercept of the line, and in part *c* the student is asked to write the equation of the line. If a student answers one or both of parts *a* and *b* incorrectly, but the student’s answer to part *c* is consistent with the student’s responses to parts *a* and *b*, then the student receives full credit for part *c*. Thus, whether a student’s response to part *c* is correct may depend on the student’s responses to parts *a* and *b*.

Two of the four conditionally scored equation tasks were scored conditionally on numeric responses, while the other two were scored conditionally on graphic responses. There were no conditionally scored graph tasks in the PAAs.

|                      | Equation Tasks | Graph Tasks |
|----------------------|----------------|-------------|
| Absolutely Scored    | 9              | 3           |
| Conditionally Scored | 2              |             |
|                      | 2              |             |

**Figure 1.** CBAL Mathematics tasks.

|          |                 |
|----------|-----------------|
| Task     | Question Number |
| Carnival | 2               |

Can't go back to the previous question.

The calculator should be available here and to the end of the task.

|               |      |     |      |
|---------------|------|-----|------|
| Testing Tools |      |     |      |
| Back          | Calc | Pro | Next |

The graph represents the data in the table. All rides cost the same amount.

|  |         |         |         |
|--|---------|---------|---------|
| Number of rides ( $x$ )                  | 1       | 2       | 3       |
| Total cost (including admission) ( $y$ ) | \$10.00 | \$12.00 | \$14.00 |

**Carnival Cost**

Let  $x$  = the number of rides and  $y$  = the total cost

- What is the slope of the line?
- What is the  $y$ -intercept of the line?
- Write an equation of the line that contains the data points.

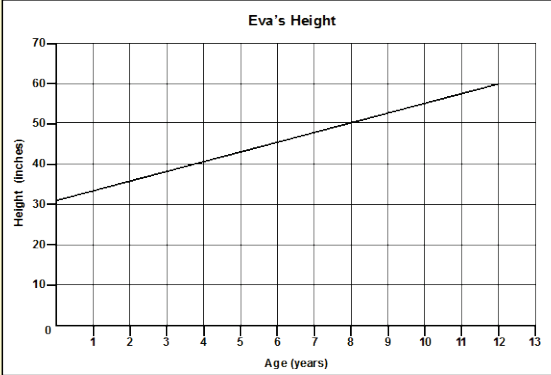
**Figure 2. Carnival, Item 2.**

**Absolutely scored equation tasks.** For seven of the nine absolutely scored equation tasks, it was possible to write the m-rater scoring model using the existing generic keys in KeyBuilder. Two of the tasks, however, required a new generic key; these were Heights 6 and Heights 9. In these tasks, the correct response is a linear equation of the form  $h = ma + b$  with a range of acceptable values for  $m$  and  $b$ . For example, for Heights 6, the response  $h = ma + b$  is correct if  $2 \leq m \leq 3$  and  $30 \leq b \leq 32$ . (See Figure 3.) The existing Linear Equations in the Plane (LEP) key, used for scoring the other linear equation tasks, could not handle this situation; so a new revised LEP key was written, in which the parameters in the model equations have tolerances as well as values. For example, with the revised LEP key, the model equation for Heights 6 is  $h = 2.5a + 31$ , with a tolerance of 0.5 for the slope and 1 for the  $y$ -intercept.

It should be noted that most of these tasks were not scored dichotomously, but had fairly complicated partial-credit rubrics. In no case did these rubrics present a problem for KeyBuilder, other than the case mentioned above.

|                  |                 |            |  |
|------------------|-----------------|------------|--|
| Task             | Question Number | 60 minutes | Testing Tools  |
| Heights & Growth | 6               |            | <input type="button" value="Back"/> <input type="button" value="Calc"/> <input type="button" value="Pro"/> <input type="button" value="Next"/> |

The line of best fit of Eva's height measurements for ages 4 to 12 years has been extended to age 0.



| Age (years) | Height (inches) |
|-------------|-----------------|
| 0           | 30              |
| 4           | 38              |
| 8           | 50              |
| 12          | 62              |

Let

$a$  = age in years and

$h$  = height in inches.

Based on this graph, write an equation using  $a$  and  $h$  that shows how Eva's height could be found using her age.

**Figure 3. Heights and Growth, Item 6.**

**Graph tasks.** Three of the tasks asked the student to draw a graph, using ETS's graph editor. Two of the tasks asked the student to draw a line, while the remaining task asked the student to plot three points. Scoring models were written in Alchemist for all three tasks, as explained above.

**Conditionally scored equation tasks.** Two of the equation tasks (including Carnival 2, mentioned above—see Figure 2) were scored conditionally on numeric responses to previous tasks. Neither c-rater nor KeyBuilder was designed to handle conditional scoring based on previous tasks. The c-rater support team was able to score responses conditionally by using as input a response file containing the relevant previous responses as well as the response being scored, making it appear to c-rater as a single response. There was no capability in c-rater or elsewhere to combine responses to several tasks into one response file, so the combining had to be done manually. A batch file was prepared in which the various responses were concatenated. A typical response to the three parts of Carnival 2 might have looked like this:

<ID: 1001><FIELD: 1>

2

<ID: 1001><FIELD: 2>

8

<ID: 1001><FIELD: 3>

<eq>y=2\*x+8</eq>

The reformatted response would have looked like this:

<ID: 1001><FIELD: 1>

<eq>M=2, B=8, y=2\*x+8</eq>

The model equation would then be

<eq>y=M\*x+B</eq>

While this ad hoc solution was sufficient for the CBAL project and may be adequate for batch scoring most tasks, real-time scoring requires enhancements that automate the procedure. The necessary enhancements were built in 2010 as part of an ETS R&D allocation project for CBAL titled *Automated Scoring of CBAL Mathematics Tasks*.

The remaining two equation items were scored conditionally on graph responses to previous tasks (see, for example, All Wet 6—Figure 4). In each case, the graph response was a line, and the conditionally scored task asked for the equation of the line. For these tasks, it was not sufficient to concatenate the student’s response to the graph task with the response to the equation task. The key for the equation task requires the slope and the  $y$ -intercept of the graph response, while the actual response to the graph task consists of the coordinates of the two points that the student plotted in the graph editor to generate the line, together with other characters that indicate the parameters used to configure the graph and, sometimes, the coordinates of two points that generate a stimulus line. For example, a student’s response to All Wet 6 might look like the following:

<ID: 1001><FIELD: 1>

["M\_WET06\_GRID01", \_grid:=[0.0,5.0,1.0,0.0,20.0,2.0], Line:=[SOURCE\_RES,line, [0.0,0.0], [5.0,4.0]], StimulusLine:=[SOURCE\_RES,line, [0.0,0.0], [5.0,20.0]]]

<ID: 1001><FIELD: 2>

<eq>y=0.8x</eq>

The relevant part of this response is the fact that the student's line passed through the points (0,0) and (5,4)—the highlighted part. To produce a file that m-rater could score, I imported the response file into Excel and used the functionality of Excel to extract the  $x$  and  $y$  coordinates of the two points and to calculate from them the slope and the  $y$ -intercept of the line that the student plotted. I then created a response file containing the information m-rater needed to score the responses.

<ID: 1001><FIELD: 1>

<eq>M=0.8, B=0, y=0.8\*x</eq>

|         |                 |            |  |                    |
|---------|-----------------|------------|--|--------------------|
| Task    | Question Number | 60 minutes | Student cannot return to the previous screen | Testing Tools      |
| All Wet | 6               |            |  | Back Calc Pro Next |

Grant Hackett holds the world record for the fastest 1,500-meter swim. His average speed was approximately 4 miles/hour.

If Jennifer Figge had swam 5 hours per day for the entire 2,400 miles, her average speed would have been 20 miles/hour.

Assume that Figge swam 5 hours each day at a constant rate of 20 miles per hour. The line on the graph shows her distance  $d$  as a function of time  $t$ .

Assume Hackett swam for 5 hours at a constant rate of 4 miles/hour.

a. Add a line to the graph that represents Hackett's distance as a function of time.

b. What is the equation of the line for Hackett?

Figure 4. An equation response is scored conditionally on a graphic response.

It should be possible to use the graph functions in KeyBuilder to do the necessary calculations and then to pass that information to the relevant equation keys. Because it was necessary to prepare a batch file for scoring anyway, the c-rater support team did not spend the resources to automate the calculation within KeyBuilder. But to support real-time scoring, this procedure will need to be automated.

Once the format of the batch file was set, it was not difficult to use KeyBuilder to write scoring models for the conditionally scored tasks, in spite of the fact that KeyBuilder was not built with conditional scoring in mind.

### **Testing the Scoring Models**

As the scoring models were written, they were tested against simulated scored responses, as described above. In the course of debugging these models, I discovered a bug in the LEP key. If an otherwise correct response uses the variables  $x$  and  $y$  instead of the variables that should be used, LEP will score the response as correct. For example, in All Wet 11, the correct response is  $d = 4 + 20b$ , but the response  $y = 4 + 20x$  would have been scored as correct.

There is a fairly simple workaround for this bug—create a new concept for the incorrect equation. That is, suppose Concept 1 is the correct equation  $d = 4 + 20b$ , and there is a scoring rubric assigning 1 point if Concept 1 is matched. Create a new Concept 2 to be the equation  $y = 4 + 20x$  and a new scoring rubric assigning 0 points if Concept 2 is matched. In the list of scoring rules in Alchemist, place the rule for Concept 2 before the rule for Concept 1. The response  $y = 4 + 20x$  will match both concepts, but will be assigned 0 points because the rule for a Concept 2 match appears in the scoring rules list before the rule for a Concept 1 match.

The revised LEP key does not have this bug (nor do any of the other generic keys). Eventually, the c-rater support team hopes to combine the LEP key and the revised LEP key and, at the same time, fix this bug.

### **Cleaning the Data**

Before the actual student responses for the equation items could be scored, the data needed to be manually cleaned. The PAAs did not use an equation editor to capture student responses to the equation tasks; instead, students entered their responses in a text box, with no restrictions on what characters they could enter. (The CBAL Mathematics team has determined



that ETS's equation editor is difficult for seventh and eighth graders to use.) As a result, the responses needed to be cleaned and formatted before m-rater could understand them:

- Any text that the student had entered had to be deleted.
- Any spaces in the equation had to be deleted.
- An asterisk (\*) had to be inserted to replace each instance of implicit multiplication.
- Sometimes parentheses had to be inserted to make the response unambiguous.

It is worth noting that, in 2010, MathFlow, an equation editor from Design Science, was used to capture student responses. As a result, it was not necessary to clean the data. See Fife (2011) for details about MathFlow.

Another consequence of the student's unrestricted ability to enter characters is the fact that students could enter the right equation with the wrong variables. It was then necessary to decide how to score these responses. Other than the previously mentioned bug involving  $x$  and  $y$ , m-rater will score a response as incorrect if the response does not contain the correct variables in the correct case. If credit (or partial credit) is to be given for such responses, then separate concepts need to be written for each possible combination of incorrect variables.

It is possible, even with a text box to capture the student's response, to restrict the characters that a student can enter. Thus, it is possible to prevent the incorrect variables error. What is relevant here is whether this error is procedural or conceptual. This issue was discussed by Fife, Graf, Ohls, and Marquez (2008). They give two examples of the incorrect variables error, one in which the error is procedural and one in which it is conceptual:

Task: Write the equation of a line with a slope of  $-3$ .

Response:  $y = -3x + b$

The student clearly understands the concept being tested. The incorrect variables error is procedural and can be prevented by restricting the letters that the student is allowed to enter.

Task: Write a linear equation in the form  $ax + b = 0$  with a solution of  $x = -3$ .

Response:  $-3a + b = 0$

The student has some idea of how to attack the problem, but either does not know what to do next or does not understand the nature of the response that is expected. Here the incorrect variables error is conceptual; it might be reasonable, therefore, to allow this error by allowing the student to enter the characters  $a$  and  $b$  as well as  $x$  and  $y$  (Fife et al., 2008, pp. 22–23).

## Scoring Student Responses

After the student response data had been cleaned, the student responses were scored using the m-rater scoring models. The responses were then human scored and the human scores compared with the m-rater scores, as described in the section on validation of scoring models. There were only two discrepancies; two responses for Carnival 2c were scored incorrectly because the data had not been cleaned properly. In each case, an asterisk had not been inserted to indicate multiplication. When the asterisk was inserted in the data and the responses were rescored, the m-rater score agreed with the human score. (See Table 1.)

**Table 1**

### *Improperly Cleaned Responses*

| Student's response  | Properly cleaned response |
|---------------------|---------------------------|
| $y = 10 + (x - 1)2$ | $y = 10 + (x - 1) * 2$    |
| $y = 2x + 8$        | $y = 2 * x + 8$           |

## Automated Scoring of Solution Steps—Literature Review

The automated scoring of equations and graphs has been around more than a decade and has been well documented; see, for example, Bennett et al. (2000). Initially, these automated scoring systems were used to score the response as right or wrong, but more recently ETS's m-rater has been used to provide partial credit and to identify common errors (Fife et al., 2008; Shute & Underwood, 2006). But m-rater is still only used to score a single response; it has not been used to score a sequence of responses that constitute the solution steps to a problem.

Some early research at ETS involved the scoring of solutions to constructed response algebra word problems using an automated system known as GIDE (Bennett, 1993; Bennett & Sebrechts, 1994; Martinez & Bennett, 1992; Sebrechts, Bennett, & Katz, 1993; Sebrechts, Bennett, & Rock, 1991). For each problem to be scored, a set of solution steps was determined; these steps were viewed as goals that an ideal solution should achieve. A sample of student responses was collected to train the system to recognize when a goal had been achieved. The system could reproduce human scores with a reasonable degree of reliability. Extensive initial work, however, was required for each problem to prepare the system to score that problem.

Singley and Bennett (1998) investigated the use of schema theory to develop automated scoring models for scoring solution steps. This approach was not as labor-intensive as the GIDE approach.

There has been some recent research on the development of tutoring systems that can help students solve problems, but the problems must be highly structured so that the automated system can evaluate the student's response. For example, at Carnegie Mellon University, Corbett, McLaughlin, and Scarpinato (2000) have developed cognitive tutors that match a student's work with a cognitive model to assess whether or not the student has mastered a certain aspect of the solution. But the student must answer the questions in a highly structured format that the tutor can map to the cognitive model. Similar remarks apply to a system called Ms. Lindquist, also developed at Carnegie Mellon University, by Heffernan and Koedinger (2002).

Nguyen and Kulm (2005) discussed the importance of web-based assessment; they describe a web-based assessment system that features randomization (on-the-fly variants) and automated adapted feedback based on the student errors. But their system exclusively uses multiple-choice and numeric-response items. There is no scoring of equations, either as an end-product or as an intermediate step in arriving at the solution.

Ashton, Beevers, Korabinski, and Youngson (2006) described a system for scoring solution steps in which the problem is scaffolded. If the student answers the question incorrectly, solution steps for answering the question are listed. The student then provides the solution for each step. Each step can then be scored, conditionally on the student's response to the preceding steps. This has two advantages: the student's error can be diagnosed, and partial credit can be assigned. Ashton et al. (2006) gave the following example:

Find the equation of the line tangent to the graph of the function  $g(x) = x^2 + 3x + 5$  when  $x = 1$ .

The correct response is  $y = 5x + 4$ . If instead a student responds, for example,  $y = 4x + 5$ , the student is presented with a sequence of steps:

Step 1: If  $y = x^2 + 3x + 5$ , what is the value of  $y$  when  $x = 1$ ?

Step 2: What is  $dy/dx$ ?

Step 3: What is the slope of the tangent line at  $x = 1$ ?

Step 4: What is the equation of the tangent line at  $x = 1$ ?

Suppose the student then provides these responses:

Step 1: 9

Step 2:  $2x + 3$

Step 3: 4

Step 4:  $y = 4x + 5$

We can see that the student has answered Steps 1 and 2 correctly, Step 3 incorrectly, and Step 4 correctly based on the student's responses to the previous steps.

Ashton et al. (2006) showed that this approach can produce improved student performance on standardized assessments in the United Kingdom. However, as they themselves point out (p. 97), "in many cases, breaking the question into smaller parts gave away the strategy, which itself was a learning point."

Livne, Livne, and Wight (2007) described a system developed at the University of Utah that assigns partial credit to a response by parsing the response, generating a syntax tree, and then comparing the syntax tree of the response to the syntax tree of the ideal response. While this method can attempt to identify the source of error in a response, it does not look at the steps leading up to the incorrect response.

All of these techniques attempt to provide a diagnostic assessment of a student's incorrect response. In some cases, the problem is scaffolded so that each step is isolated and can be scored; in other cases, a diagnosis is inferred from the response using a cognitive or analytical model of some sort. But an inference may or may not be reliable, and the scaffolding can give away the store. In the future, we hope to develop the mathematical framework to score individually a series of solution steps with the view toward assessing the extent to which the steps are leading to the correct solution.

### **Summary and Future Work**

Because of ETS's investment in building KeyBuilder, m-rater scoring models for equation and graph tasks can now be written quickly and accurately by content specialists, with 100% reliability of scoring. If there are equation tasks that cannot be scored by the existing generic keys, either task-specific handcrafted keys or a new generic key can be written.

Conditional scoring of tasks presents a different challenge, not to the writing of the scoring models, but to the creation of the appropriate response files. The ad hoc approach used in

2009 will not work for all items that require conditional scoring, nor will it work for real-time scoring. In 2010, the c-rater support team made the necessary enhancements to c-rater to support conditional scoring.

Another significant challenge for the eventual operational use of CBAL is the lack of an appropriate interface for capturing equation responses. One of the 2009 CBAL projects involved a study of existing equation editors to see if there is one that is more user-friendly than ETS's and that ETS can license. As a result of that study, the CBAL Mathematics team recommended obtaining a developer's license for a product called MathFlow, by Design Science. In 2010, ETS successfully used MathFlow in CBAL pilot assessments.

Finally, an important capability for CBAL would be the ability to evaluate a student's solutions steps (instead of just scoring the final answer). This may not require major technological development as much as it requires a better understanding of the mathematics of solution steps. In the future, the CBAL Mathematics team hopes to conduct the basic research to determine how a scoring engine such as m-rater can score solution steps.

## References

- 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*, 37(1), 93–119.
- Bennett, R. E. (1993). Toward intelligent assessment: An integration of constructed response testing, artificial intelligence, and model-based measurement. In N. Frederiksen, R. J. Mislevy, & I. Bejar (Eds.), *Test theory for a new generation of tests* (pp. 99–123). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Bennett, R. E., & Gitomer, D. H. (2009). Transforming K-12 assessment: Integrating accountability testing, formative assessment, and professional support. In C. Wyatt-Smith & J. Cumming (Eds.), *Educational assessment in the 21st century* (pp. 43–61). New York, NY: Springer.
- Bennett, R. E., Morley, M., & Quardt, D. (2000). Three response types for broadening the conception of mathematical problem solving in computerized tests. *Applied Psychological Measurement*, 24, 294–309.
- Bennett, R. E., & Sebrechts, M. M. (1994). *The accuracy of automatic qualitative analyses of constructed-response solutions to algebra word problems* (ETS Research Report No. 94-04). Princeton, NJ: ETS.
- Corbett, A., McLaughlin, M., & Scarpinato, K. C. (2000). Modeling student knowledge: Cognitive tutors in high school and college. *User Modeling and User-Adapted Interactions*, 10, 81–108.
- Fife, J. H. (2009). *CBAL mathematics scoring models: 2008 project report*. Unpublished manuscript.
- Fife, J. H. (2011). *Equation editors and the automated scoring of mathematics tasks*. Manuscript in preparation.
- Fife, J. H., Graf, E. A., Ohls, S., & Marquez, E. (2008). *Identifying common misconceptions: An analysis of the mathematics intervention module (MIM) data* (ETS Research Memorandum No. RM-08-16). Princeton, NJ: ETS.
- Graf, E. A., Harris, K., Marquez, E., Fife, J. H., & Redman, M. (2010, March). Highlights from the Cognitively Based Assessment of, for, and as Learning (CBAL) project in mathematics. *ETS Research Spotlight*, 3, 19–30.

- Heffernan, N. T., & Koedinger, K. R. (2002). An intelligent tutoring system incorporating a model of an experienced human tutor. In S. A. Cerri, G. Gouardères, & F. Paraguaçu (Eds.), *Proceedings of the Sixth International Conference on Intelligent Tutoring Systems* (pp. 596–608). Biarritz, France: Springer-Verlag.
- Livne, N. L., Livne, O. I., & Wight, C. A. (2007). Can automated scoring surpass hand grading of students' constructed responses and error patterns in mathematics? *Journal of Online Learning and Teaching*, 3(3). Retrieved from <http://jolt.merlot.org/vol3no3/livne.htm> on August 30, 2011.
- Martinez, M. E., & Bennett, R. E. (1992). A review of automatically scorable constructed-response item types for large-scale assessment. *Applied Measurement in Education*, 5(2), 151–169.
- Nguyen, D. M., & Kulm, G. (2005). Using web-based practice to enhance mathematics learning and achievement. *Journal of Interactive Online Learning*, 3(3). Retrieved from <http://www.ncolr.org/jiol/issues/pdf/3.3.1.pdf>.
- Sebrechts, M. M., Bennett, R. E., & Katz, I. R. (1993). *A research platform for interactive performance assessment in graduate education* (ETS Research Report No. RR-93-08). Princeton, NJ: ETS.
- Sebrechts, M. M., Bennett, R. E., & Rock, D. A. (1991). *Machine-scorable complex constructed-response quantitative items: Agreement between expert system and human raters' scores* (ETS Research Report No. RR-91-11). Princeton, NJ: ETS.
- Shute, V., & Underwood, J. (2006). Diagnostic assessment in mathematics problem solving. *Technology, Instruction, Cognition and Learning*, 3, 151–166.
- Singley, M. K., & Bennett, R. E. (1998). *Validation and extension of the mathematical expression response type: Applications of schema theory to automatic scoring and item generation in mathematics* (ETS Research Report No. RR-97-19). Princeton, NJ: ETS.