



Research Report

ETS RR-13-26

Automated Scoring of Mathematics Tasks in the Common Core Era: Enhancements to M-rater in Support of *CBAL*[™] Mathematics and the Common Core Assessments

James H. Fife

December 2013

ETS Research Report Series

EIGNOR EXECUTIVE EDITOR

James Carlson
Principal Psychometrician

ASSOCIATE EDITORS

Beata Beigman Klebanov
Research Scientist

Heather Buzick
Research Scientist

Brent Bridgeman
Distinguished Presidential Appointee

Keelan Evanini
Managing Research Scientist

Marna Golub-Smith
Principal Psychometrician

Shelby Haberman
Distinguished Presidential Appointee

Gary Ockey
Research Scientist

Donald Powers
Managing Principal Research Scientist

Gautam Puhan
Senior Psychometrician

John Sabatini
Managing Principal Research Scientist

Matthias von Davier
Director, Research

Rebecca Zwick
Distinguished Presidential Appointee

PRODUCTION EDITORS

Kim Fryer
Manager, Editing Services

Ruth Greenwood
Editor

Since its 1947 founding, ETS has conducted and disseminated scientific research to support its products and services, and to advance the measurement and education fields. In keeping with these goals, ETS is committed to making its research freely available to the professional community and to the general public. Published accounts of ETS research, including papers in the ETS Research Report series, undergo a formal peer-review process by ETS staff to ensure that they meet established scientific and professional standards. All such ETS-conducted peer reviews are in addition to any reviews that outside organizations may provide as part of their own publication processes. Peer review notwithstanding, the positions expressed in the ETS Research Report series and other published accounts of ETS research are those of the authors and not necessarily those of the Officers and Trustees of Educational Testing Service.

The Daniel Eignor Editorship is named in honor of Dr. Daniel R. Eignor, who from 2001 until 2011 served the Research and Development division as Editor for the ETS Research Report series. The Eignor Editorship has been created to recognize the pivotal leadership role that Dr. Eignor played in the research publication process at ETS.

**Automated Scoring of Mathematics Tasks in the Common Core Era: Enhancements to
M-rater in Support of *CBAL*TM Mathematics and the Common Core Assessments**

James H. Fife

Educational Testing Service, Princeton, New Jersey

December 2013

Find other ETS-published reports by searching the ETS ReSEARCHER
database at <http://search.ets.org/researcher/>

To obtain a copy of an ETS research report, please visit
<http://www.ets.org/research/contact.html>

Action Editor: Keelan Evanini

Reviewer: Isaac Bejar and Randy Bennett

Copyright © 2013 by Educational Testing Service. All rights reserved.

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



Abstract

The m-rater scoring engine has been used successfully for the past several years to score *CBAL*TM mathematics tasks, for the most part without the need for human scoring. During this time, various improvements to m-rater and its scoring keys have been implemented in response to specific CBAL needs. In 2012, with the general move toward creating innovative tasks for the Common Core assessment initiatives, in traditional testing programs, and with potential outside clients, and to further support CBAL, m-rater was enhanced in ways that move ETS's automated scoring capabilities forward and that provide needed functionality for CBAL: (a) the numeric equivalence scoring engine was augmented with an open-source computer algebra system; (b) a design flaw in the graph editor, affecting the way the editor graphs smooth functions, was corrected; (c) the graph editor was modified to give assessment specialists the option of requiring examinees to set the viewing window; and (d) m-rater advisories were implemented in situations in which m-rater either cannot score a response or may provide the wrong score. In addition, 2 m-rater scoring models were built that presented some new challenges.

Key words: automated scoring, graph response, mathematics test item, equation editor, MathML, local extremum, cubic spline, computer algebra system

Table of Contents

	Page
Computer-Algebra Systems	2
Graphs of Smooth Functions	4
Setting the Viewing Window in the Graph Editor	12
M-rater Scoring Advisories.....	16
M-rater Scoring Models	22
<i>Smart Phones</i> Part 2 Item 2	22
<i>Heights and Growth</i> Part 2 Item 4	27
Implications for Further Research	34
Examinee Performance When Examinees Must Select the Viewing Window	34
How to Score Curves?	34
How to Score Line-of-Best-Fit Questions?	34
M-rater Advisories: Too Many False Positives?	34
What Is the Best Interface for Entering Equations?	35
Does Entering Mathematics Questions Online Change the Construct Being Tested?	35
Conclusion	35
References.....	37
Notes	40
Appendix.....	41

List of Tables

	Page
Table 1. Cut Points for <i>Dams and Drought</i> Item	31
Table 2. Human/M-rater Agreement Using Simulated Responses to Set Cut Points	31
Table 3. Human/M-rater Agreement Using Actual Student Responses to Set Cut Points	32

List of Figures

	Page
Figure 1. A CBAL mathematics item with the graph editor.....	5
Figure 2. The item in Figure 1 with a response.	6
Figure 3. A twice-differentiable cubic spline through a given set of data points.	7
Figure 4. A Hermite cubic spline defined using the arithmetic mean.	9
Figure 5. A cubic spline defined using the arithmetic mean with no local maximum at $x = 4$	10
Figure 6. A Hermite cubic spline defined using the harmonic mean.....	11
Figure 7. The CBAL item in Figure 1 with a configuration screen.....	14
Figure 8. The CBAL item in Figure 1 with the examinee-produced viewing window.	14
Figure 9. <i>Smart Phones</i> Part 2 Item 2.	23
Figure 10. The <i>Smart Phones</i> item with a better choice of horizontal gridlines.	24
Figure 11. The distribution of x_1 and y_1 values in examinee responses, with x_1 ranging from 0 to 24 in increments of 2 and y_1 ranging from 0 to 1200 in increments of 100.....	24
Figure 12. The distribution of x_1 and y_1 values in examinee responses, with x_1 ranging from 11.0 to 13.0 in increments of 0.2 and y_1 ranging from 100 to 300 in increments of 20.....	25
Figure 13. The distribution of x_1 and y_1 values in examinee responses, with x_1 ranging from 11.0 to 13.0 in increments of 0.1 and y_1 ranging from 100 to 300 in increments of 10.....	25
Figure 14. <i>Heights and Growth</i> Part 2 Item 4.....	28
Figure 15. <i>Dams and Drought</i> —The data and the line of best fit $y = -10.4173x + 329.3009$	29
Figure 16. <i>Dams and Drought</i> —The data and the lines $y = -12x + 350$ and $y = -9x + 310$	29
Figure 17. <i>Dams and Drought</i> —The data with the line $y = -12x + 310$	30
Figure 18. A sample response to <i>Heights and Growth</i> Part 2 Item 4.	33

An automated scoring engine for scoring mathematics responses to constructed-response tasks was developed at ETS in the mid-1990s (Bennett, Morley, & Quardt, 2000; Bennett, Steffen, Singley, Morley, & Jacquemin, 1997). Now called *m-rater*, this scoring engine has been used successfully for the past several years to score mathematics tasks in the Cognitively Based Assessment of, for, and as Learning (CBAL™) assessment, for the most part without the need for human scoring (Breyer, Williams, Fife, & Lewis, 2012; Fife, 2011). The goal of the CBAL 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, 2010; Bennett & Gitomer, 2009). Assessments are being developed in mathematics, reading, writing, and science. 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 *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, the mathematics responses being scored by m-rater.

The mathematics component of CBAL is based on relevant research in cognitive science and mathematics education, as described in Graf (2009). Some of the early work in CBAL mathematics task development is described in Graf, Harris, Marquez, Fife, and Redman (2009, 2010). Further work is discussed in Fife, Graf, and Ohls (2011) and Cayton-Hodges et al. (2012).

During the time that m-rater has been used to score CBAL mathematics tasks, various improvements to m-rater and its scoring keys have been implemented in response to specific CBAL needs (Fife, 2011). In 2012, with the general move toward creating innovative tasks for the Common Core assessment initiatives, in traditional testing programs, and with potential outside clients, and to further support CBAL, m-rater was enhanced in ways that move ETS's automated scoring capabilities forward. In addition, two m-rater scoring models were built that presented some new challenges.

In 2012, m-rater was enhanced in four areas:

1. The numeric equivalence scoring engine was augmented with an open-source computer algebra system.
2. A design flaw in the graph editor, affecting the way the editor graphs smooth functions, was corrected.

3. The graph editor was modified to give assessment specialists the option of requiring examinees to set the viewing window.
4. M-rater advisories were implemented in situations in which m-rater either cannot score a response or may provide the wrong score.

Each of these enhancements will be described in turn, followed by a discussion of the two scoring models that were built and the particular challenges that they presented. The paper will conclude with a discussion of some areas of possible further research and development.

Computer-Algebra Systems

The m-rater engine scores two types of responses to constructed-response (CR) tasks: mathematical expressions (or equations) and graphs. (M-rater also scores numeric responses, but, for the purposes of this discussion, one can think of numeric responses as simple expressions.) When the response to a CR task is an expression, m-rater determines if the examinee's response is mathematically equivalent to the correct expression. There are basically two automated methods for determining if two expressions are equivalent—a computer algebra system (CAS) can be used to determine symbolically if the two expressions are equivalent, or the two expressions can be numerically evaluated at sufficiently many points so that one can be reasonably confident that the expressions are equivalent. ETS's first scoring engine for what were then called *Mathematical Expression* items used a rudimentary CAS developed from open-source code (Bennett et al., 1997). It was determined, however, that computer algebra systems were too slow for immediate scoring, as is required for computer adaptive tests, so the code was rewritten to use numerical evaluation (Bennett et al., 2000).

Since then, hardware and software have improved quite a bit; there has also been extensive research on the use of computer algebra systems in mathematics education (Drijvers, 2003; Kramarski & Hirsch, 2003; Pointon & Sangwin, 2003) and assessment (Sangwin, 2002, 2003). In particular, Sangwin (2003) has stated that he believes that “in the near future *all* computer aided assessment systems will link computer algebra and assessment to perform ... automatic ... marking of mathematics ...” (p. 2) [emphasis in the original]. In the same paper, he remarked that “Some current commercial testing products compare answers by substituting a number of random values *which is clearly more limited*” (p. 4) [emphasis added]. In Sangwin

(2002) he remarked that he believes that “all *reputable* computer aided assessment systems in the near future will contain ... marking systems [that use a CAS]” (p. 1) [emphasis added].

As CBAL assessment specialists began to develop more complicated CR tasks and as ETS Research made plans to extend the level of mathematics tasks scorable by m-rater pursuant to the Common Core assessment initiatives, the consensus developed that higher mathematics items would be easier to score using a CAS. At the level of mathematics at which numerical evaluation has been used by m-rater (Algebra I, mostly), the method has roughly the same level of accuracy as symbolic manipulation (Bennett et al., 2000). But for higher levels of mathematics, numerical evaluation is limited. For example, it would be extremely difficult to determine the accuracy of the equation $d(x^3)/dx = 3x^2$ or the equation $\int_0^1 x^3 dx = \frac{1}{4}$ using numerical evaluation. By switching to a CAS, the full power of the CAS is available to score mathematics equations at any level.

Using a CAS has other advantages, as well:

- If the correct response to an item is an equation that is not a function (for example, the equation of the sphere $x^2 + y^2 + z^2 = 1$), the author of a numeric evaluation scoring key may need to parameterize the equation to ensure that proper points are chosen for evaluation. Even for curves (two-dimensional equations), finding a parameterization can be difficult. For example, if $n > 2$, there is no rational parameterization of the equation $x^n + y^n = 1$; this is a consequence of Fermat’s Last Theorem. This need for parameterization limits the nature of items that numeric evaluation scoring engines can score; depending on the level of mathematics involved, this limitation could be severe. With a CAS, parameterizations are not needed, and such responses are easy to score.
- It is theoretically possible for a numeric evaluation scoring engine to score a response incorrectly. For example, suppose the correct response to an item is the equation $y = x + 1$ and an examinee responds $y = (x^2 - 1)/(x - 1)$. The second equation is not equivalent to the first, and therefore should be scored as incorrect. But the second equation differs from the first only at the point $x = 1$. Therefore, unless $x = 1$ happens to be one of the points selected by the scoring key for evaluation, a numeric evaluation scoring engine will score the second equation as correct.

For these reasons, a computer algebra system was added to m-rater, augmenting the numeric-evaluation scoring engine. There are several third-party proprietary computer algebra systems available (e.g., Mathematica, Maple), but the mathematics community seems to have adopted the open-source software package known as Sage as the default CAS for use in research (Denny, 2013). Sage is a Python-based package that incorporates several existing open-source software packages to accomplish some of its functions. In particular, functionalities required by m-rater use a specific CAS called *SymPy* (Galochkin, 2011). Dmitry Galochkin, a software developer at ETS, wrote a new key for scoring equations using the Python-based SymPy library (www.sympy.org). Galochkin called the new key *SYMPY* and modeled it after the existing numeric-evaluation keys, but, in fact, it functions quite differently. Unlike the existing m-rater keys, which are written in C++, the SYMPY key uses a Java-based version of Python called *Jython* to run Python scripts, which are precompiled into Java classes (Galochkin, 2011). Model sentences in SYMPY are written in SymPy syntax, but because SymPy syntax can be somewhat formidable, Galochkin also wrote what are called *wrapper functions* that take as arguments the entries in the response fields and encode the SymPy syntax. Using Galochkin (2011), the Alchemist¹ user's manual was revised (Fife, 2012) to include a section on writing scoring models using the new SYMPY key.

Because the old numeric-evaluation keys have been retained, m-rater can still be used with legacy scoring models, and new models can even be written using the legacy keys if that is desired.

Graphs of Smooth Functions

Besides expressions, the other major class of responses that m-rater can score are graphs. As with expressions, examinees must be given an editor in which to enter their responses. Because the automated scoring of free-hand graphs is difficult (Lukoff, 2010), the ETS graph editor was designed so that an examinee only needs to click points on a coordinate grid. As the points are clicked, they are automatically connected with a curve, with the examinee having previously clicked a button to indicate what type of curve should connect the points. Four types of curves are currently supported—straight lines, smooth curves, piecewise-linear curves, and no connecting curve at all (that is, the points plotted by the examinee are left as points). To plot a line, the examinee clicks two points; the editor draws the line containing the two points. To plot a piecewise-linear curve, the examinee clicks the singular points; the editor draws line segments

between each pair of adjacent points. To plot a smooth curve, the examinee clicks as many points as are required to generate the desired curve; the editor connects the plotted points with a smooth curve.

For example, Figure 1 shows a CBAL item in which the examinee is asked to draw the graph of a line. The examinee clicks the button labeled *Line* and then clicks any two points on the line, say the points (4,22) and (6,18); the editor then draws the line containing the two points (see Figure 2).

The screenshot shows the CBAL MATH interface. At the top, it displays 'Moving Sidewalks Section 1', 'Question # 5 of 12', and 'Timer 59 minutes'. The main title is 'CBAL MATH'. On the right, there are navigation buttons: 'Sidewalk', 'Calc', 'STOP', 'Back', and 'Next'.

The problem is titled 'Rider B on Sidewalk'. It includes a diagram of a moving sidewalk with two riders. Below the diagram is a table with the following data:

Time (seconds)	Distance of Rider B from Gate (feet)
4	22
6	18
10	10
13	4

The graph editor on the right contains the instruction: 'Draw a graph that shows Rider B's distance from the gate over time from the beginning of the sidewalk to end of the sidewalk. Click on the Line button to start.' The graph has a vertical axis labeled 'Distance of Rider B from gate (feet), y' ranging from 1 to 35, and a horizontal axis labeled 'Time (seconds), x' ranging from 1 to 20. A 'Line' button is visible on the right side of the graph area, along with 'Undo' and 'Start Over' buttons.

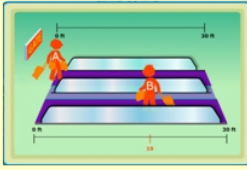
Figure 1. A CBAL mathematics item with the graph editor.

Moving Sidewalks Section 1 Question # 5 of 12 Timer 57 minutes

CBAL MATH

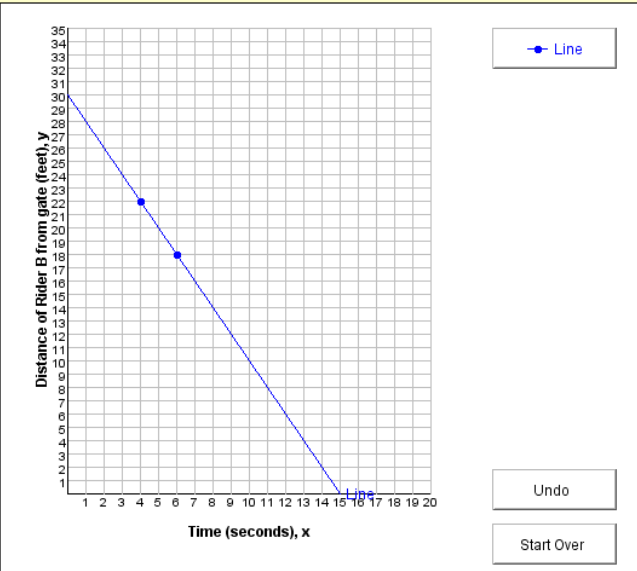
Sidewalk Calc STOP Back Next

Rider B on Sidewalk



Time (seconds)	Distance of Rider B from Gate (feet)
4	22
6	18
10	10
13	4

Draw a graph that shows **Rider B's** distance from the gate over time from the beginning of the sidewalk to end of the sidewalk. Click on the Line button to start.



Distance of Rider B from gate (feet), y

Time (seconds), x

Line

Undo

Start Over

Figure 2. The item in Figure 1 with a response.

Drawing a line containing two points is a straightforward task to program into the editor; drawing a smooth curve generated by a collection of points is more complicated. The general problem of fitting a smooth curve to discrete data points has been studied extensively in the literature. Most approaches involve defining the curve piecewise; that is, using different formulas for the curve between each pair of consecutive points in such a way that the pieces of the curve fit together to form a single smooth curve. More precisely, if $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$ are the data points, with $x_0 < x_1 < \dots < x_n$, then for each $i = 1, \dots, n$, let $f_i : [x_{i-1}, x_i] \rightarrow \mathbf{R}$ be a continuous function such that $f_i(x_{i-1}) = y_{i-1}$ and $f_i(x_i) = y_i$. The function f defined by $f(x) = f_i(x)$ for $x_{i-1} \leq x \leq x_i$ passes through each of the given data points and is continuous. If in addition each f_i is differentiable and $f'_i(x_i) = f'_{i+1}(x_i)$ for $i = 1, \dots, n-1$, then f is differentiable also. Such a function is called a *spline*.

In applications, the various functions f_i are frequently defined to be cubic polynomials. Because cubic polynomials have four coefficients, these provide enough parameters so that

equality of the second derivatives $f_i''(x_i) = f_{i+1}''(x_i)$ can also be required. In this case, the spline f will be twice differentiable (and its second derivative will be continuous).

Because twice-differentiable cubic splines have this additional degree of smoothness provided by the continuity of the second derivative, they are a particularly appealing way to fit a smooth curve to discrete data points when the extra smoothness of the curve is a desirable feature (see Figure 3). But ETS's goal in designing the graph editor was somewhat different; the goal when graphing smooth curves in the editor was to duplicate, as closely as possible, the procedure students use when graphing functions on paper. On paper, students plot some points, likely including the local extrema, and then connect the points with a smooth curve, making certain that the curve has extreme points at the desired locations. So, for example, if an examinee plotted the points in Figure 3 in the graph editor, one would want the curve drawn by the editor to have local maxima at $x = 1$ and $x = 4$, and local minima at $x = 2$ and $x = 8$. A necessary condition for these points to be local extrema is that the derivatives at these points equal 0. Therefore, to guarantee that our cubic spline has local extrema at the desired points, it is not enough to require that the derivatives of the f_i 's be equal at the data points; one must be able to specify the *values* of these derivatives at the data points. This will be in lieu of requiring continuity of the second derivatives. A cubic spline obtained in this way is called a *Hermite* cubic spline.

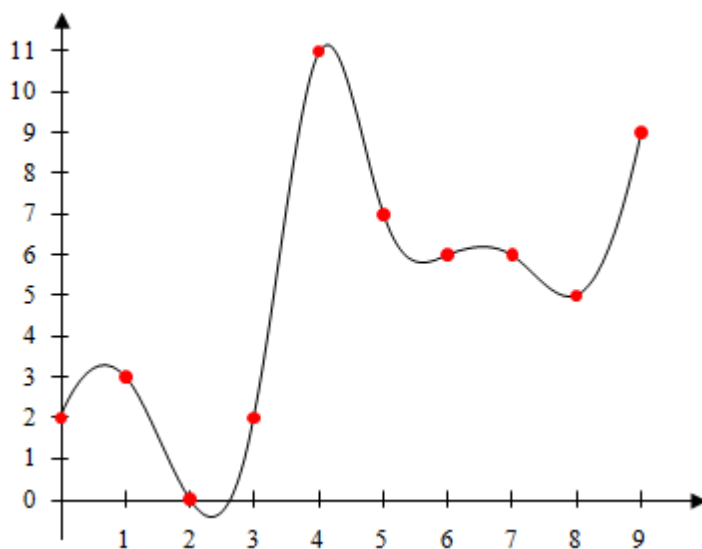


Figure 3. A twice-differentiable cubic spline through a given set of data points.

So, as before, let $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$ be the data points, with $x_0 < x_1 < \dots < x_n$, and let $m_1, \dots, m_{n-1} \in \mathbf{R}$ be real numbers. For each $i = 1, \dots, n$, let $f_i : [x_{i-1}, x_i] \rightarrow \mathbf{R}$ be the cubic polynomial $f_i(x) = a_i + b_i(x - x_{i-1}) + c_i(x - x_{i-1})^2 + d_i(x - x_{i-1})^3$ with the properties that $f_i(x_{i-1}) = y_{i-1}$ and $f_i(x_i) = y_i$ for $i = 1, \dots, n$ and $f_i'(x_{i-1}) = m_{i-1}$ and $f_i'(x_i) = m_i$ for $i = 2, \dots, n-1$. There are several ways of handling the functions f_1 and f_n ; for our purposes, it is sufficient to let these functions be quadratic polynomials such that $f_1'(x_1) = m_1$ and $f_n'(x_{n-1}) = m_{n-1}$.

Under these conditions, a straightforward calculation shows that the coefficients a_i, b_i, c_i , and d_i are defined as follows: for each $i = 1, \dots, n$, let $h_i = x_i - x_{i-1}$, $k_i = y_i - y_{i-1}$, and $s_i = k_i/h_i$, and let $m_0 = f_1'(x_0)$ and $m_n = f_n'(x_n)$. Then $a_i = y_{i-1}$, $b_i = m_{i-1}$, $c_i = (3s_i - 2m_{i-1} - m_i)/h_i$, and $d_i = (m_{i-1} + m_i - 2s_i)/h_i^2$; see, for example, Dougherty, Edelman, and Hyman (1989); Fritsch and Carlson (1980); and Hyman (1983). Because f_1 and f_n are quadratic polynomials, it follows that $d_1 = d_n = 0$, and hence $m_0 = 2s_1 - m_1$ and $m_n = 2s_n - m_{n-1}$.

It remains to determine, for our purposes, the appropriate values of the derivatives m_i . For $i = 1, \dots, n-1$, we want f to have a local maximum at x_i if $y_i > y_{i-1}$ and $y_i > y_{i+1}$, and we want f to have a local minimum at x_i if $y_i < y_{i-1}$ and $y_i < y_{i+1}$. Equivalently, we want f to have a local maximum at x_i if $s_i > 0$ and $s_{i+1} < 0$, and we want f to have a local minimum at x_i if $s_i < 0$ and $s_{i+1} > 0$. In other words, we want f to have a local extremum at x_i if s_i and s_{i+1} have opposite signs; that is, if $s_i s_{i+1} < 0$. Because a necessary condition for f to have a local extremum at x_i is that $f'(x_i) = 0$, we will set $m_i = 0$ if $s_i s_{i+1} < 0$.

Additionally, we want f to be constant on the interval $[x_{i-1}, x_i]$ if $y_i = y_{i-1}$; that is, if $s_i = 0$. Thus, we want both $m_i = 0$ and $m_{i-1} = 0$ if $s_i = 0$, or, equivalently, we want $m_i = 0$ if either $s_i = 0$ or $s_{i+1} = 0$; that is, if $s_i s_{i+1} = 0$. Thus, we set $m_i = 0$ if $s_i s_{i+1} \leq 0$.

If $s_i s_{i+1} > 0$, then m_i should be a number between s_i and s_{i+1} . In the original design of the graph editor, m_i was defined to be the arithmetic mean of s_i and s_{i+1} when $s_i s_{i+1} > 0$. Thus, in the original design of the graph editor, m_i was defined by

$$m_i = \begin{cases} \frac{s_i + s_{i+1}}{2} & \text{if } s_i s_{i+1} > 0 \\ 0 & \text{if } s_i s_{i+1} \leq 0 \end{cases} \quad (1)$$

for $i = 1, \dots, n-1$. Figure 4 shows the Hermite cubic spline through the same points as the twice-differentiable spline in Figure 3 but defined as described above, using the arithmetic mean.

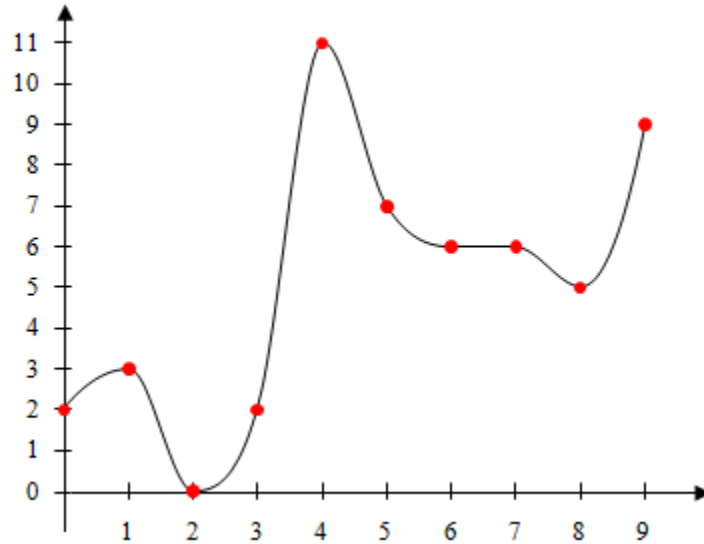


Figure 4. A Hermite cubic spline defined using the arithmetic mean.

While a zero derivative is necessary for a function to have a local extremum, it is not sufficient, and it turns out that Hermite cubic splines defined according to Equation 1 may not always have local extrema at points where $m_i = 0$. For example, Figure 5 shows a Hermite cubic spline, defined using the arithmetic mean, for which $y_4 > y_3$ and $y_4 > y_5$, and hence $m_4 = 0$, but the spline does not have a local extremum at x_4 . It appears that the spline is so steep from $x = 2$ to $x = 3$ that the curve overshoots the expected local extremum at $x = 4$. It turns out that this is much the case; in fact, we have the following theorem:

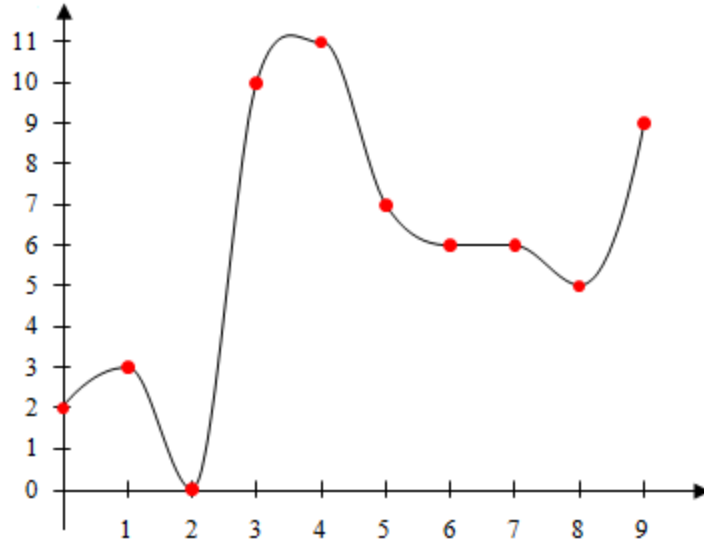


Figure 5. A cubic spline defined using the arithmetic mean with no local maximum at $x = 4$.

Theorem 1. Let f be a Hermite cubic spline through the $n + 1$ points $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$, with $x_0 < x_1 < \dots < x_n$, where m_i is defined as in Equation 1 for $i = 1, \dots, n - 1$. Let $k = 1, \dots, n - 1$ be such that $y_k > y_{k-1}$ and $y_k > y_{k+1}$.

(a) If $k > 1$ and $s_{k-1} > 5s_k$, the spline f does not have a local maximum at x_k . The spline has a local maximum at a point $u < x_k$.

(b) If $k < n - 1$ and $s_{k+2} < 5s_{k+1}$, the spline f does not have a local maximum at x_k . The spline has a local maximum at a point $u > x_k$.

(c) If $1 < k < n - 1$, $s_{k-1} \leq 5s_k$, and $s_{k+2} \geq 5s_{k+1}$, or if $k = 1$ and $s_3 \geq 5s_2$, or if $k = n - 1$ and $s_{n-2} \leq 5s_{n-1}$, the spline f has a local maximum at x_k . The spline has no other local maximum in the interval (x_{k-1}, x_{k+1}) .

For example, in Figure 5, $s_3 = 10$ and $s_4 = 1$. Because $s_3 > 5s_4$, the spline f does not have a local maximum at $x_4 = 4$, in spite of the fact that $y_4 > y_3$ and $y_4 > y_5$.

This theorem is proved in the appendix. A similar theorem holds for local minima.

It turns out that the trick to finding a Hermite cubic spline with the desired properties lies in how m_i is defined at those points where $m_i \neq 0$. In particular, Butland (1980) has shown that if

m_i is defined to be the harmonic mean of s_i and s_{i+1} when $s_i s_{i+1} > 0$, then the Hermite cubic spline will always have the desired properties:

Theorem 2 (Butland). *Let f be a Hermite cubic spline through the $n+1$ points (x_0, y_0) , $(x_1, y_1), \dots, (x_n, y_n)$, with $x_0 < x_1 < \dots < x_n$, where m_i is defined by*

$$m_i = \begin{cases} \frac{2s_i s_{i+1}}{s_i + s_{i+1}} & \text{if } s_i s_{i+1} > 0 \\ 0 & \text{if } s_i s_{i+1} \leq 0 \end{cases}$$

for $i = 1, \dots, n-1$.

(a) *If $k = 1, \dots, n-1$ is such that $y_k > y_{k-1}$ and $y_k > y_{k+1}$, the spline f has a local maximum at x_k . The spline has no other local maximum on the interval (x_{k-1}, x_{k+1}) .*

(b) *If $k = 1, \dots, n-1$ is such that $y_k < y_{k-1}$ and $y_k < y_{k+1}$, the spline f has a local minimum at x_k . The spline has no other local minimum on the interval (x_{k-1}, x_{k+1}) .*

Figure 6 shows the Hermite cubic spline through the same points as the spline in Figure 5 but defined using the harmonic mean instead of the arithmetic mean.

Theorem 2 is also proved in the appendix.

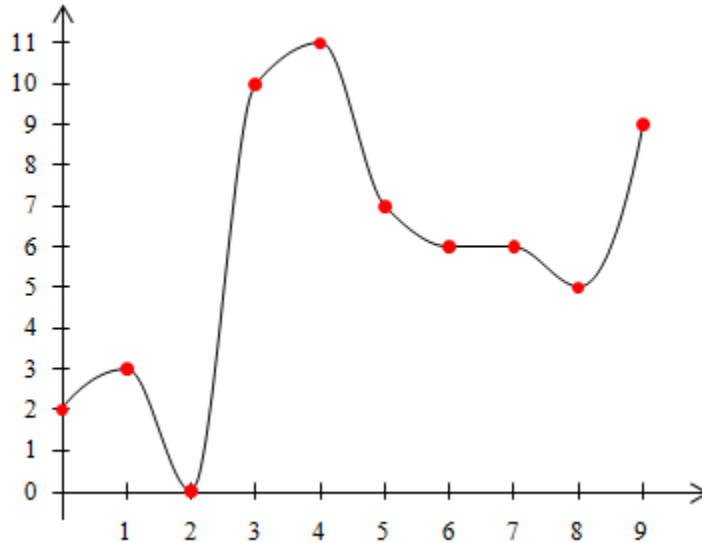


Figure 6. A Hermite cubic spline defined using the harmonic mean.

The ETS graph editor was originally designed with the derivatives m_i defined using the arithmetic mean, as in Theorem 1. As a result, the editor would generate the curve in Figure 5. The code has now been changed so that the derivatives are defined using the harmonic mean, as in Theorem 2; it follows from Theorem 2 that the curve will always have the desired properties regarding local extrema, as in Figure 6.

Setting the Viewing Window in the Graph Editor

The *viewing window* in the graph editor is the portion of the xy -plane that is visible in the editor. The viewing window is the space in which the examinee draws his or her graph; it determines the portion of the graph that can be shown. If the viewing window is too small, interesting features of the graph will be missed; if the viewing window is too large, the resulting small scale may obscure some features. For the graph editor in Figure 1, the viewing window is defined by the inequalities $0 \leq x \leq 20$ and $0 \leq y \leq 35$, with gridlines every 1 unit. This viewing window was selected and fixed during the item authoring stage. However, selecting the appropriate viewing window is an important skill that students need to have mastered (Ball & Stacey, 2001; Pierce & Stacey, 2002). Certainly, in a paper-and-pencil setting, if the examinee were asked to draw a graph on a blank sheet of paper, the examinee would need to select a range for the x - and y -axes. Even when students are using a graphing calculator to graph a function, they still need to be able to select an appropriate viewing window (Doerr & Zangor, 2000; Drijvers & Trouche, 2008; Stacey, 2005; Stacey, McCrae, Chick, Asp, & Leigh-Lancaster, 2000). According to Stacey (2005), “developing a strong concept of the viewing window (which involves ideas of domain and range, scale and zooming) [is] critical to good use of graphing functionality.” Consequently, it is desirable to be able to require examinees to set the viewing window of the graph editor in an item. Additionally, because the testing program might want to assess the examinee’s choice of viewing window, it is desirable that the parameters selected by the examinee, which define the viewing window, be scorable elements.

Another aspect of the graph in Figure 1 that was fixed during the item authoring stage is the axis labels—*Time (seconds)* for the x -axis and *Distance of Rider B from the gate* for the y -axis. But, again, the ability to select or determine the correct axis labels is an important part of the construct being measured, and the capability to require examinees to enter these labels for themselves, and to score their entries, is a desirable feature.

Based on these considerations, the graph editor has been modified so that item authors and content specialists have the option, for each item, of requiring examinees to specify the viewing window and the axis labels before the graphing window will be displayed in that item. When this option is chosen, fields for the examinee to enter minimum and maximum values for x and for y and the labels for the x - and y -axes appear in the space where the graphing window would normally be displayed. (There is a check to prevent the examinee from entering a maximum value that is less than the minimum value.) The examinee clicks a button, and the graphing window appears with the examinee's choice of viewing window and axis labels. See Figures 7 and 8.


Because examinees cannot be expected to select grid points that correspond to correct responses, it is necessary, for items in which examinees set the viewing window, that snap-to-grid be disabled. This means that the scoring rubrics for these items must specify a level of precision in the responses; this level of precision depends on the grid width in the examinee's graph. It follows that if two different examinees have different grid widths, their responses will not be comparable, even if they have the same minimum and maximum values for x and y .

Similarly, if an examinee selects gridlines that are inappropriate (see, for example, the discussion of the *Smart Phones* item below), the inappropriate selection of gridlines could make the item more difficult. Either of these situations can interfere with what the item is supposed to measure and can make analysis of student responses problematic. For this reason, examinees are not required (or allowed) to set the grid width or the gridlines when setting the viewing window. The approximate number of gridlines is established by the item author and/or content staff and is the same for all examinees. For each examinee's graph, the editor calculates a reasonable grid width and appropriate gridlines, based on the examinee's minimum and maximum values of x and y and the approximate number of gridlines established for that item. Here, *reasonable* means that the grid width equals 1, 2, or 5 times a (positive or negative) power of 10. As a result, gridlines such as those in the *Smart Phones* item below will never be generated. So that it all works out, it may be necessary to adjust the minimum and maximum values of x and y , as well. In Figures 7 and 8, the approximate number of gridlines is set at 20 for both axes.

Moving Sidewalks Section 1 Question # 4 of 9 Timer 58 minutes

CBAL MATH

Rider B on Sidewalk



Time (seconds)	Distance of Rider B from Gate (feet)
4	22
6	18
10	10
13	4

Draw a graph that shows Rider B's distance from the gate over time from the beginning of the sidewalk to the end of the sidewalk. First enter values to determine the viewing window and enter axis labels. Then click the Create Graph button.

Range of X-axis:
 min:
 max:

Range of Y-axis:
 min:
 max:

X-axis label:

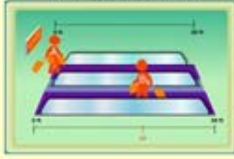
Y-axis label:

Figure 7. The CBAL item in Figure 1 with a configuration screen.

Moving Sidewalks Section 1 Question # 4 of 9 Timer 58 minutes

CBAL MATH

Rider B on Sidewalk



Time (seconds)	Distance of Rider B from Gate (feet)
4	22
6	18
10	10
13	4

Draw a graph that shows Rider B's distance from the gate over time from the beginning of the sidewalk to the end of the sidewalk. Click the Line button to start.

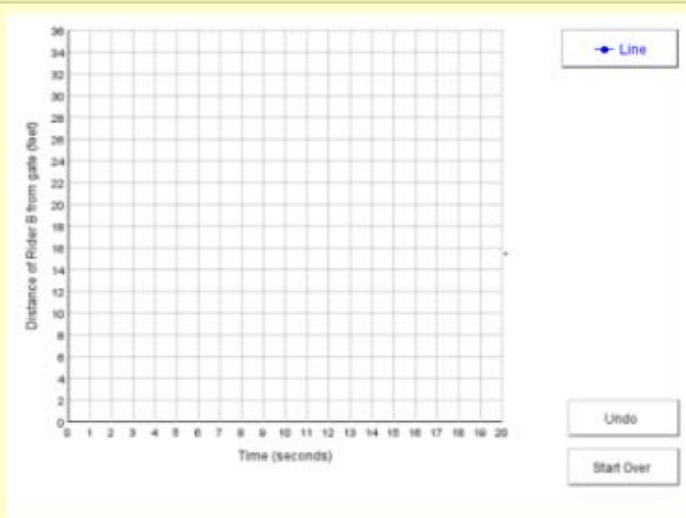


Figure 8. The CBAL item in Figure 1 with the examinee-produced viewing window.

Here are the formulas used to determine the grid width and the actual minimum and maximum values of x and y . Let the examinee's minimum and maximum values for x be x_{\min} and x_{\max} , and let n be the approximate number of gridlines as set by the item authors and reviewers. (Similar calculations hold for the y -axis.) Let $r = x_{\max} - x_{\min}$. The grid width g is the smallest integer greater than or equal to r/n that equals 1, 2, or 5 times a power of 10. The grid width g and the actual minimum and maximum values of x are determined by the following equations:

$$\begin{aligned}
 a &= \left\lceil \log\left(\frac{r}{n}\right) \right\rceil \\
 b &= \left\lceil \frac{r}{n \times 10^a} \right\rceil \\
 c &= \frac{b}{2} \left(\frac{2.5 - b}{|2.5 - b|} + 1 \right) + \frac{5}{2} \left\lceil \frac{b}{5} \right\rceil \left(\frac{b - 2.5}{|b - 2.5|} + 1 \right) \\
 g &= c \times 10^a \\
 \text{actual } x_{\min} &= \left\lfloor \frac{x_{\min}}{g} \right\rfloor \times g \\
 \text{actual } x_{\max} &= \left\lceil \frac{x_{\max}}{g} \right\rceil \times g
 \end{aligned}$$

Gridlines will occur at integer multiples of g .

Note: In these formulas, $\lfloor x \rfloor$ means the greatest integer less than or equal to x , and $\lceil x \rceil$ means the least integer greater than or equal to x . So, for example, $\lfloor 2.5 \rfloor = 2$ and $\lfloor -2.5 \rfloor = -3$.

Also, $|x|$ means the absolute value of x and \log means the base-10 logarithm.

The parameters that set the viewing window are already reported to m-rater as part of the response; the graph key in m-rater will be modified to score these fields. Item authors and reviewers can require the examinee to enter axis labels or not, as they wish, on an item-by-item basis. Examinees can be provided with a drop-down list from which to select a label, or they can be provided with a text box in which to enter the label as free text. Either way, the student's entry can be scored; a free-text entry would be scored by ETS's c-rater short-answer scoring engine and, thus, must satisfy c-rater's requirements.

M-rater Scoring Advisories

Breyer et al. (2012) investigated issues related to human/m-rater agreement. In particular, they examined situations in which humans and m-rater disagree. While the overall human/m-rater agreement was very high (as would be expected), there were some responses for which the humans agreed with each other but disagreed with m-rater. As reported in Breyer et al., there are five general reasons why humans and m-rater sometimes disagreed:

- Sometimes the humans made a mistake.
- Sometimes the humans did not understand complicated rubrics.
- Humans are more forgiving of typographical errors and other nonconstruct-related errors than is m-rater (which is not forgiving at all).
- Sometimes the equation editor that was being used was unable to generate content Mathematical Markup Language (MathML), which is necessary for m-rater scoring.²
- Sometimes the equation editor generated incorrect content MathML.

The first two of these bullets represent instances in which the m-rater score is clearly correct and the human score is wrong. The third bullet, however, represents instances in which the human score, while technically incorrect, may be a better measure of the examinee's proficiency than the m-rater score (which is likely to be 0), and the last two bullets represent instances in which, due to MathML conversion problems, the m-rater score could be incorrect.

Regarding the third bullet, three distinct types of nonconstruct-related errors are common:

- Keystroke errors—For example, an examinee types an extra decimal point ($p = .0.2 + 0.8m$ instead of $p = 0.2 + 0.8m$). Usually, such a response will be syntactically incorrect, and m-rater will be unable to score it.
- Incorrect variables—For example, an examinee might give an equation in x and y even though the item asked for an equation in s and t . If the equation is otherwise correct, human scorers may choose to give partial credit or perhaps even full credit.
- Text entered into equations—A common error is for examinees to attempt to include text when entering equations. The text is usually interpreted as additional variables, which renders the equation incorrect.

One approach that would eliminate both problems with incorrect variables and problems with text in equations would be to restrict the characters that examinees can enter in the equation

editor, so that, for example, examinees could not enter letters that were not appropriate variables for the item. Unfortunately, this does not seem possible with the equation editor that CBAL is currently using; however, another equation editor, for which this input restriction is possible, is being investigated for implementation. If this investigation is successful and CBAL is able to implement the new editor, then these types of errors will be eliminated; examinees will only be allowed to enter letters that are relevant variables to the item. Note that *relevant variables* are not necessarily the same as *correct variables*. One might want to allow students to enter certain incorrect variables if that error is relevant to the construct being measured (Fife, Graf, Ohls, & Marquez, 2008).

Because m-rater scores the content MathML output of the equation editor, m-rater will not be able to score a response if the editor cannot generate content MathML. Breyer et al. (2012) found several responses for which content MathML could not be generated but for which the human scorers awarded at least some credit for the response. One example occurred with responses to an item with the prompt, “Write an expression for the number of minutes it takes to make b big greeting cards.” The correct response is the expression $12b$, but several students attempted to enter “ $12b$ min” into the equation editor. The editor interpreted *min* to be the function that returns the minimum of its arguments; because in the expression $12b$ *min* there are no numbers or variables that can be the arguments of the minimum function, the response cannot be encoded into content MathML.

Finally, as mentioned earlier, incorrect content MathML is sometimes generated. This occurs when a mathematical expression can have more than one mathematical meaning. The editor must decide which meaning to give the expression, and it did not always choose the meaning that CBAL students intended. For example, for one item, the correct response was $6s$. In one administration of the item, several students responded $s(6)$; the editor interpreted this entry to mean that s was a function and $s(6)$ was the value of the function at the argument 6.

M-rater advisories have now been implemented for responses for which m-rater would otherwise give a score of 0 but for which, based on the research described above, it is thought that humans might give at least partial credit. These advisories are being issued in the following cases:

- M-rater cannot parse the response.
- Content MathML cannot be generated.
- Unexpected (incorrect) variables are present.

- The editor may have generated incorrect content MathML.
- The presentation MathML contains a line break.

It is expected that the first bullet will target most responses in which the examinee has made a keystroke error. The third bullet will target responses with incorrect variables and also responses in which the examinee has attempted to enter text.

The fourth bullet is based on known examples of incorrect content MathML. The incorrect content MathML found in the Breyer et al. study occurred in two types of situations. One was the $s(6)$ problem above. The other situation occurred when the examinee used a comma as a separator in a large number. In these responses, the editor interpreted the comma as a serial comma, indicating the presence of two separate expressions. For example, the linear equation $m = 20,000p$ would be interpreted as the two expressions $m = 20$ and $000p$.

For each of these situations, it was possible to identify features of the content MathML that indicated that the content MathML might be incorrect. To see how this works, here is an example of a correct response from the equation editor for the examinee response $6x + s$:

```
<eqeditor>
  <math>
    <mrow>
      <mn>6</mn>
      <mi>s</mi>
      <mo>+</mo>
      <mi>x</mi>
    </mrow>
  </math>
  <math>
    <apply>
      <plus/>
      <apply>
        <times/>
        <cn>6</cn>
        <ci>s</ci>
      </apply>
      <ci>x</ci>
    </apply>
  </math>
</eqeditor>
```

The `<eqeditor>` tag indicates a response from the equation editor CBAL uses. Inside the `<eqeditor>` tag are two `<math>` tags; the first encloses the presentation MathML, and the second encloses the content MathML. The presentation MathML is a straightforward representation of the expression as entered by the examinee, but the content MathML represents

the actual mathematics encoded in the expression. There are several things to note. The first is that the second `<math>` tag has only one child, an `<apply>` tag. This `<apply>` tag means that the expression performs an operation on some number of arguments. The first child tag of the `<apply>` tag specifies the operation, and the remaining child tags are the arguments of the operation. In this case, the first child tag is the `<plus/>` tag, indicating that the operation of addition is involved; the two additional child tags indicate that the addition operation is being applied to two quantities. The first of these quantities is itself a product, so the corresponding tag is another `<apply>` tag. This `<apply>` tag also has three child tags; the first indicates that the operation is multiplication, and the second and third indicate the quantities being multiplied.

Note that each `<apply>` tag has at least three child tags—the first specifies the operation, and the remaining specify the quantities the operation is applied to. Now consider the response from the equation editor for the examinee response $s(6)$:

```
<eqeditor>
  <math>
    <mrow>
      <mi>s</mi>
      <mo>( </mo>
      <mn>6</mn>
      <mo>)</mo>
    </mrow>
  </math>
  <math>
    <apply>
      <ci>s</ci>
      <cn>6</cn>
    </apply>
  </math>
</eqeditor>
```

Again, the presentation MathML represents the expression as it was entered by the examinee.

But, in the content MathML, the `<apply>` tag should have three child tags, the first specifying the multiplicative operation, and the others specifying the terms being multiplied; the content MathML should look like this:

```
<math>
  <apply>
    <times/>
    <ci>s</ci>
    <cn>6</cn>
  </apply>
</math>
```

Instead the `<apply>` tag has only two child tags. The first tag specifies an operation that only has one argument; that is, s is being treated as a function of one variable.

It follows that what we have called *the $s(6)$ problem* produces content MathML containing an `<apply>` tag with only two child tags. Thus, the presence of an `<apply>` tag in content MathML with only two child tags indicates that the content MathML may be incorrect; this condition triggers one of the m-rater advisories.

Of course, an `<apply>` tag with only two child tags could arise in a perfectly legitimate way if the examinee has entered an expression involving a function of one variable, such as the square root function or, in more advanced assessments, a trigonometric or logarithmic function. Currently there are no CBAL items whose responses require such expressions; however, in the future, it will be necessary to revise the code so that standard functions of one variable do not trigger the advisory.

As stated above, the other situation leading to incorrect content MathML is a response containing a separator comma. For example, consider the response from the equation editor for the examinee response $m = 20,000p$:

```
<eqeditor>
  <math>
    <mrow>
      <mi>m</mi>
      <mo>=</mo>
      <mn>20</mn>
      <mo>,</mo>
      <mn>000</mn>
      <mi>p</mi>
    </mrow>
  </math>
  <math>
    <apply>
      <eq/>
      <ci>m</ci>
      <cn>20</cn>
    </apply>
    <apply>
      <times/>
      <cn>000</cn>
      <ci>p</ci>
    </apply>
  </math>
</eqeditor>
```

In this example, even the presentation MathML does not get the expression entirely correct, in that it fails to treat 20,000 as a single number. More serious, though, is the fact that in the content MathML, the `<math>` tag has two `<apply>` tags, one for the equation $m = 20$ and the other for the product $000p$. A `<math>` tag with more than one `<apply>` tag indicates a response that has been interpreted as a series of expressions, and, therefore, this condition indicates content MathML that may be incorrect. Thus, a `<math>` tag with more than one `<apply>` tag also triggers an m-rater advisory.

In this case, we do not have a concern that the advisory could be triggered by a legitimate response. Because m-rater cannot score a response containing multiple expressions or equations, a legitimate response to an item intended for m-rater scoring will never contain more than one expression.

Note that there is no way to preprocess a response to remove the commas. Examinees enter their responses directly in the equation editor, which records the responses in presentation MathML, converts the presentation MathML to content MathML, and returns the presentation and content MathML. The content MathML is then processed for m-rater scoring. By the time m-rater receives the response (in the form of content MathML), the damage has already been done.

Note also that if we are able to implement a new equation editor that allows us to prevent examinees from entering certain characters, we will be able to prevent examinees from entering commas altogether, and this particular problem will no longer occur.

Finally, in the course of implementing advisories related to the first four bullets noted above, it was discovered that if the examinee has entered a line break in the response, the content MathML that is generated will not encode the line break; instead, it will juxtapose the two lines. Thus, if the examinee enters

$$\begin{aligned}x &= 2 \\y &= 3\end{aligned}$$

the equation editor will generate content MathML for the expression $x = 2y = 3$. A later version of the editor may have corrected this problem (and the problem may not afflict other editors), but that does not matter because m-rater cannot score multiline responses, anyway.

Whenever a response falls into one of these categories, m-rater will now issue an advisory suggesting that the response should be human scored.

M-rater Scoring Models

Several items in the 2012 CBAL mathematics multistate pilot data were scored by m-rater, but most of these items had been previously administered and scored using m-rater; hence, existing models could be used for scoring. Two items, however, required new scoring models. These two items were *Smart Phones Part 2 Item 2* and *Heights and Growth Part 2 Item 4*. Both of these scoring models had some interesting features that are worth noting.

Smart Phones Part 2 Item 2

The item is shown in Figure 9. This is a graphing item in which the examinee is asked to plot four points. The interesting feature here is that the snap-to-grid feature of the graph editor was disabled. When the snap-to-grid feature is enabled, the examinee can only click grid points; when the examinee clicks somewhere inside the grid, the point snaps to the nearest grid point. For example, in the item in Figure 1, if the examinee wants to click the two points (4, 22) and (6, 18), the examinee can click anywhere on the graph close to (4, 22), and the point (4, 22) will be recorded. The examinee does not need to click (4, 22) exactly. As a result, examinee responses are precise, and scoring models can be written with the assumption that the examinee will provide precise responses.

The downside to snap-to-grid, however, is that when snap-to-grid is enabled, the item can only ask that grid points be plotted. As a result, items that use real data are problematic; they are frequently impossible to configure in a way that meets the grid-point constraint. An example is the *Smart Phones* item in Figure 9. If the graph editor were configured so that the four points to be plotted are grid points, then there would need to be a gridline at every unit along the y-axis; hence, there would need to be 1,500 horizontal gridlines. Clearly, this would make the graph unreadable. So, for this task, snap-to-grid was disabled. As a result, however, examinees cannot be expected to plot points with precision. To plot the point (12, 168), for example, the examinee must estimate where 168 is along the y-axis; as a result, an examinee may, for example, plot the point (12, 165) or (12, 171) instead of (12, 168).³

Smartphones Part 2 Question # 2 of 8 Timer

CBAL MATH

Calc STOP Back Next

The table below shows the number of Android™ apps if that number were to double every 6 months after February 1, 2011.

Date	Number of Months since August 1, 2010	Number of Thousands of Android™ Apps
Aug 1, 2010	0	42
Feb 1, 2011	6	84
Aug 1, 2011	12	168
Feb 1, 2012	18	336
Aug 1, 2012	24	672
Feb 1, 2013	30	1344

The graph below shows the two points corresponding to the numbers cited in the article. Add the additional points shown in the table on the left to the graph.

Figure 9. Smart Phones Part 2 Item 2.

Actually, this task was made more difficult by the poor choice of horizontal gridlines. With the graph in its current configuration, the student must estimate where along the line, from 0 to 375, the number 168 falls—an interesting mathematical problem but not the skill that the item is intended to measure. A better choice of horizontal gridlines would have been to have a gridline every 100 units; see Figure 10. As a result of the current configuration, the level of precision in the student responses was quite low. For example, let (x_1, y_1) be the point a particular examinee plots for (12, 168). Figure 11 shows the distribution of the values of x_1 and y_1 in the examinee responses collected in the pilot.⁴ As one can see, the values for both x_1 and y_1 have a fairly large range, though most of the values cluster around 12 and 168. More detailed views, however, show that, while almost all of the x_1 values fall between 11.8 and 12.2, the y_1 values are more spread out, with more values falling between 140 and 150 and between 190 and 200 than between 160 and 170; see Figure 12 and Figure 13.

Smartphones Part 2 Question # 2 of 8 Timer

CBAL MATH Calc STOP Back Next

The table below shows the number of Android™ apps if that number were to double every 6 months after February 1, 2011.

Date	Number of Months since August 1, 2010	Number of Thousands of Android™ Apps
Aug 1, 2010	0	42
Feb 1, 2011	6	84
Aug 1, 2011	12	168
Feb 1, 2012	18	336
Aug 1, 2012	24	672
Feb 1, 2013	30	1344

The graph below shows the two points corresponding to the numbers cited in the article. Add the additional points shown in the table on the left to the graph.

Figure 10. The Smart Phones item with a better choice of horizontal gridlines.

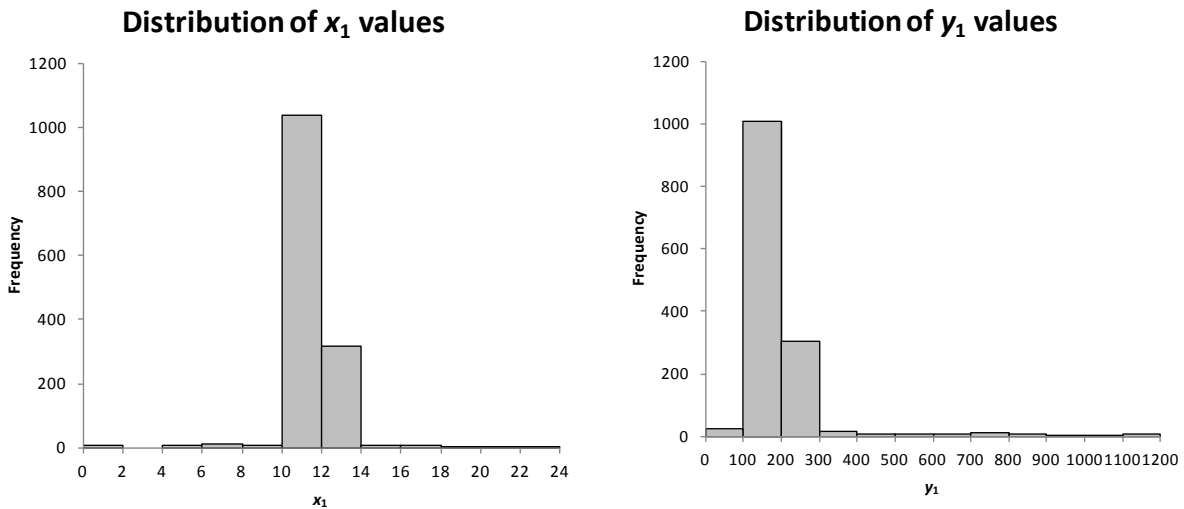


Figure 11. The distribution of x_1 and y_1 values in examinee responses, with x_1 ranging from 0 to 24 in increments of 2 and y_1 ranging from 0 to 1200 in increments of 100.

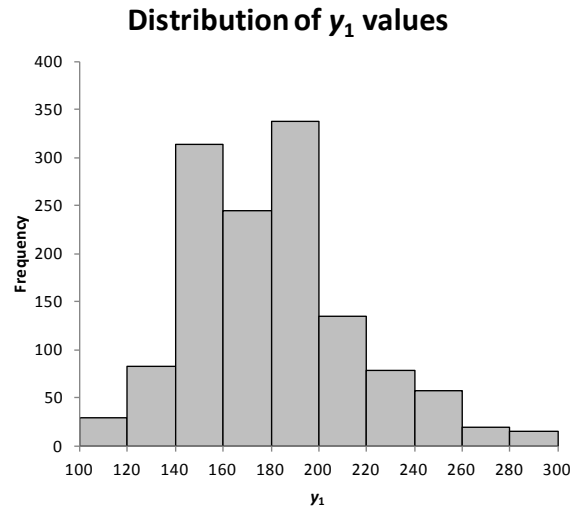
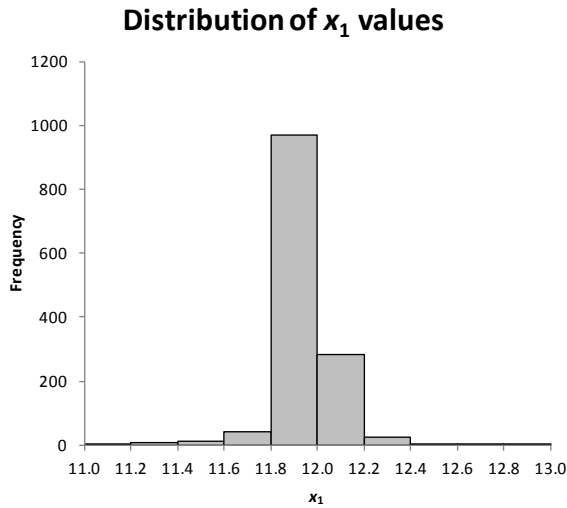


Figure 12. The distribution of x_1 and y_1 values in examinee responses, with x_1 ranging from 11.0 to 13.0 in increments of 0.2 and y_1 ranging from 100 to 300 in increments of 20.

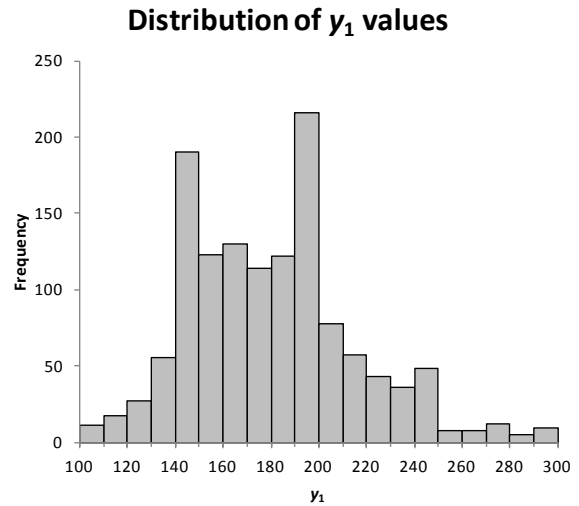
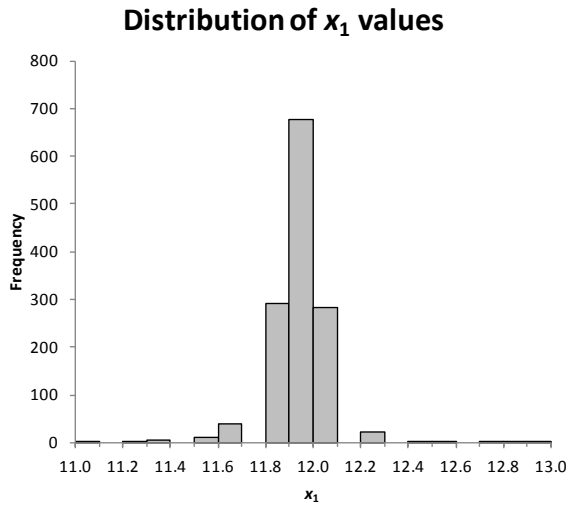


Figure 13. The distribution of x_1 and y_1 values in examinee responses, with x_1 ranging from 11.0 to 13.0 in increments of 0.1 and y_1 ranging from 100 to 300 in increments of 10.

Writing the scoring rubrics. In any event, the fact that, even in the best of circumstances, examinees could not be expected to plot the points with precision meant that the scoring model for this item needed to establish a tolerance for each point, within which responses would be considered correct. For example, any point (x, y) for which $|x - 12| \leq \delta$ and $|y - 168| \leq \varepsilon$ for

suitable δ and ε would be considered as being a correct plotting of the point (12, 168). When we had collected about 400 responses, histograms were prepared, similar to the ones here, showing the distribution of the x and y values corresponding to the four points to be plotted. From these histograms, members of the CBAL mathematics team established the tolerances for each point.

Writing scoring rubrics presented other challenges, as well. What if an examinee plotted more than four points? An examinee could plot as many as 15 points. One early version of the rubrics would have given full credit to a response containing four correct points and 11 incorrect points. Or, what if an examinee tried to plot the same point more than once? Although the graph editor does not allow an examinee to plot *exactly* the same point more than once, the tolerances established in the scoring rubrics create a rectangle around each of the target points, so that any point plotted inside that rectangle counted as a correct point. Thus, a scoring rubric that says, “Four points are plotted and they are all correct” would give full credit to a response that effectively tries to plot the same point as four different instances within the target rectangle.

The mathematics team finally agreed on the following rubrics:

- 2 points if each of the four points is plotted correctly, with no additional points
- 1 point if three of the four points are plotted correctly, with no additional points
- 0 points otherwise

Implementing the scoring rubrics. To implement these rubrics, they were translated into *concepts* and *scoring rules* (Fife, 2012). A concept is a feature of the response whose presence justifies awarding the response full or partial credit. For example, one concept might be the correct equation, numeric response, or graph; another could be a response that is to be awarded partial credit. A scoring rule is a rule of the form

Any n of Concepts x, \dots, z is worth m points.

For example, a scoring rule might say, “Any 2 of Concepts 1, 2, and 3 is worth 1 point.” This framework of concepts and scoring rules was originally developed for c-rater (Sukkarieh & Blackmore, 2009) and adapted for m-rater when m-rater was integrated with c-rater (Fife, 2011).

For the *Smart Phones* item, the scoring rubrics were translated into these concepts and scoring rules:

Concepts

1. Point 1 plotted within tolerance
2. Point 2 plotted within tolerance
3. Point 3 plotted within tolerance
4. Point 4 plotted within tolerance
5. More than 4 points plotted

Scoring rules

1. Any 1 of {5} is worth 0 points.
2. Any 4 of {1-4} is worth 2 points.
3. Any 3 of {1-4} is worth 1 point.

Heights and Growth Part 2 Item 4

This item is shown in Figure 14. The examinee is shown a scatter plot and is asked to draw a “line of best fit” for the data. The examinee is not expected to calculate the actual least-squares regression line; rather, he or she is expected to plot a line that visually seems to fit the data reasonably well.

We have had mixed experience with both the human scoring and the automated scoring of such items. We administered a similar task in 2008 as part of the extended task *Dams and Drought*. The responses were double human scored on a 3-point scale (0–2). But, the scoring rubrics were not clear as to what would constitute a 2-point response and what would constitute a 1-point response. As a result, the human-human agreement was poor; the proportion of exact agreement was $A = 0.66$, Cohen’s kappa was $\kappa = 0.56$, and the quadratic-weighted kappa was $QWK = 0.56$ ($n = 32$).

Heights and Growth Part 2 Question # 4 of 5 Timer 59 minutes CBAL MATH

The table shows the height, in inches, for a sample of girls at age 2 and corresponding heights when fully grown.

Height at 2	31.0	31.5	33.0	33.5	34.5	35.0	36.0
Height When Fully Grown	60.0	61.9	63.5	64.4	65.8	66.3	66.9

Drag the handles (•) on the blue line to create a line of best fit for the data in the table.

Figure 14. Heights and Growth Part 2 Item 4.

Because the score awarded to a response depends on how close the response line is to the actual line of best fit, a precise measure of closeness is required to score the responses with m-rater. Because m-rater calculates the slope m and the y -intercept b of each response, establishing independent tolerances on m and b for each score point and scoring the responses based on these tolerances might seem like a reasonable approach. This approach, however, could lead to inappropriate scores. For example, the line of best fit for the data in the *Dams and Drought* item is the line $y = -10.4173x + 329.3009$; Figure 15 shows the data and the line of best fit. Figure 16 shows the data and two other lines: $y = -12x + 350$ and $y = -9x + 310$. Both of these responses were assigned a score of 2 by content experts. Hence, if scores were based on tolerances for m and b , then -9 and -12 would be within the tolerance for m , and 310 and 350 would be within the tolerance for b . It follows that the response $y = -12x + 310$ would be assigned a score of 2. But, see Figure 17; not surprisingly, content experts assigned a score of 0 to this response.

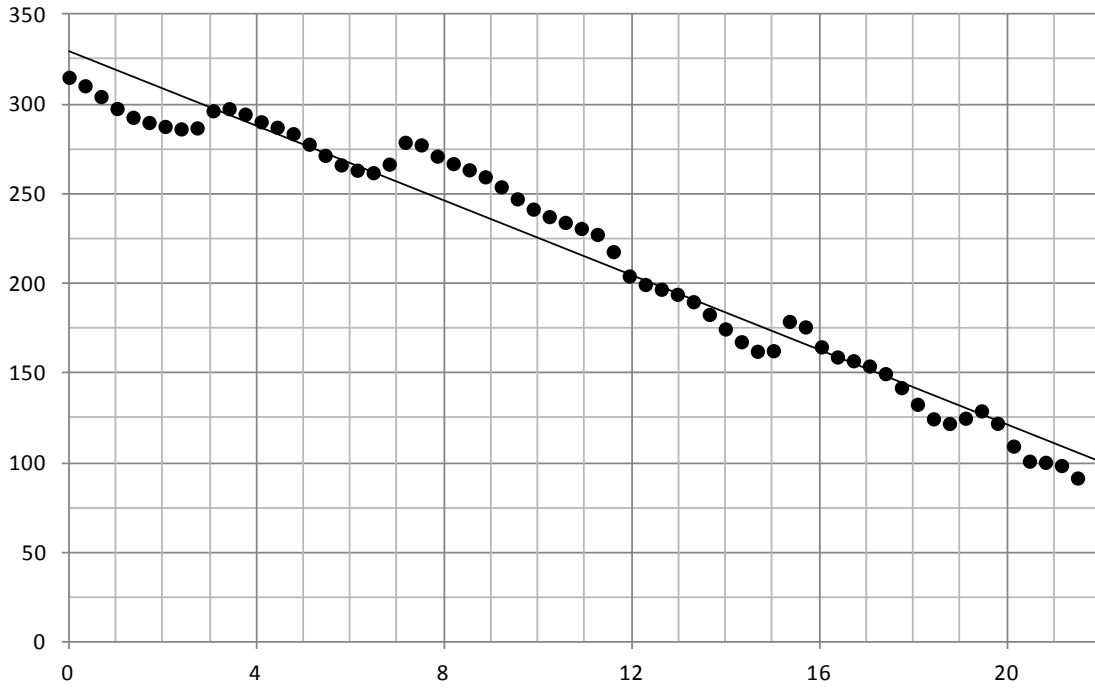


Figure 15. Dams and Drought—The data and the line of best fit $y = -10.4173x + 329.3009$.

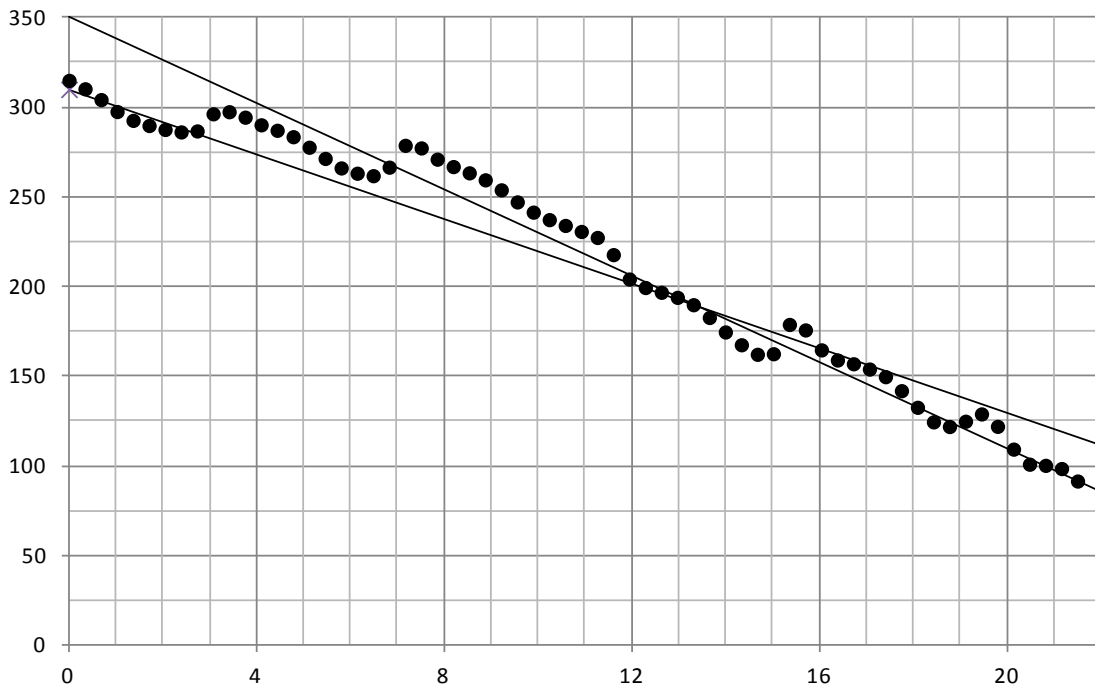


Figure 16. Dams and Drought—The data and the lines $y = -12x + 350$ and $y = -9x + 310$.

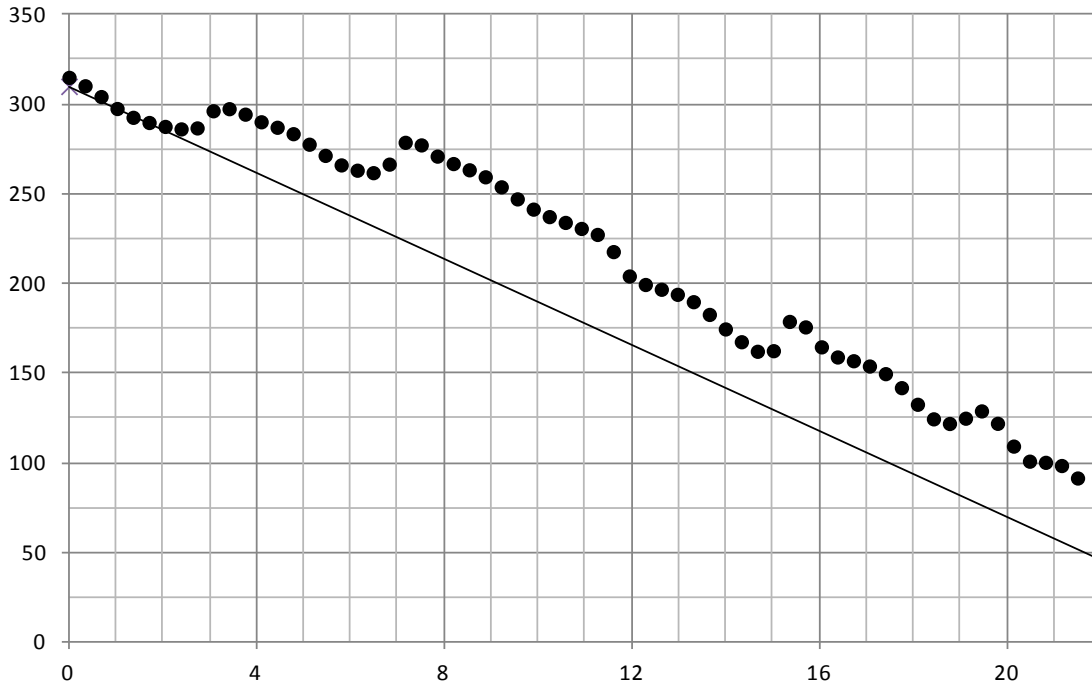


Figure 17. Dams and Drought—The data with the line $y = -12x + 310$.

An alternative approach might be to create an m-rater scoring model based on the relative difference between the root mean square deviation (RMSD) of the examinee's line and the RMSD of the actual line of best fit. Let n denote the number of data points on the scatter plot and let (x_i, y_i) denote the i^{th} data point. If the examinee's response is the line $y = mx + b$, then the RMSD of the examinee's response line is

$$\sqrt{\frac{1}{n} \sum_{i=1}^n (mx_i + b - y_i)^2}$$

The actual line of best fit is (by definition) the line for which the RMSD is the least. Let r be the RMSD for the line of best fit. Then the relative difference between the RMSD of the examinee's line and the RMSD of the line of best fit is

$$d = \frac{\sqrt{\frac{1}{n} \sum_{i=1}^n (mx_i + b - y_i)^2} - r}{r}$$

A large collection of simulated responses was generated and d calculated for each response. Members of the CBAL mathematics team then established cut points for each of the score levels. It was decided to score on a 4-point scale (0–3), so three cut points were selected; see Table 1. Thus, a response would receive a score of 3 if $d < 0.36$, a score of 2 if $0.36 \leq d < 0.6012$, and so on. Because the human/human agreement for this rubric was poor, and because the human scoring and the m-rater scoring were on different scales, no attempt was made to determine human/m-rater agreement.

Table 1
Cut Points for Dams and Drought Item

Score point	Cut point
3	0.36
2	0.6012
1	1.096

A similar item was administered in December 2009 as part of the *Bigfoot* task. Again, the responses were double human scored on a 3-point scale (0–2), and they were scored by an m-rater model based on cut points in the d -metric established by members of the CBAL mathematics team, using simulated response. This time, the human/human agreement was quite good ($A = 0.94$, $\kappa = 0.88$, $QWK = 0.91$; $n = 110$), but the human/m-rater agreement was poor (see Table 2). Clearly, the cut points based on simulated responses did not align with human scoring. However, when the human scores of the actual student responses were used to establish cut points in the d -metric, it was possible to simulate m-rater scores that yielded human/m-rater agreements in line with the human/human agreement (see Table 3).

Table 2
Human/M-rater Agreement Using Simulated Responses to Set Cut Points

	Human 1/m-rater	Human 2/m-rater
Exact agreement	0.41	0.42
Cohen’s kappa	0.29	0.29
Quadratic-weighted kappa	0.60	0.59

Table 3***Human/M-rater Agreement Using Actual Student Responses to Set Cut Points***

	Human 1/m-rater	Human 2/m-rater
Exact agreement	0.97	0.93
Cohen's kappa	0.95	0.86
Quadratic-weighted kappa	0.96	0.89

The next time a line-of-best-fit item was administered was in December 2010. *Heights and Growth* Part 2 Item 4 was administered to 125 examinees. The responses were double human scored on a dichotomous scale; no m-rater scoring model was written at that time. When the *Heights and Growth* task was included in the 2012 mathematics multistate pilot, it was decided to write a scoring model using the d -metric based on the human scores from December 2010. Unfortunately, the human/human agreement this time was not good. Rater 1 was a good bit more generous than Rater 2; of the 125 responses, there were 22 for which Rater 1 gave a score of 1 and Rater 2 gave a score of 0. (There were no responses for which Rater 2 gave a score of 1 and Rater 1 gave a score of 0.)⁵ While the proportion of agreement was not bad ($A = 0.82$), Cohen's kappa was quite low ($\kappa = 0.56$). A third human rater has scored the responses on which the first two raters disagreed; in effect, the final human score was H1 when H1 and H2 agreed and H3 when H1 and H2 did not agree. When the final human scores were used to establish the d -metric cut point, it was possible to produce simulated m-rater scores that yielded a human/m-rater agreement of $A = 0.93$ and $\kappa = 0.75$.

However, the final human scores were not consistent; there were five identical responses of which three had a final human score of 1 and two had a final human score of 0. The inconsistency was both internal with Rater 1 and external between Rater 2 and Rater 3. Rater 2 gave all five responses a score of 0. Rater 3 scored the three responses for which Rater 1 gave a score of 1 and agreed with Rater 1 (disagreeing with Rater 2) on all three scores. If the two scores of 0 were changed to 1, the human/m-rater agreement increased to $A = 0.94$ and $\kappa = 0.80$.

At this point, a member of the CBAL mathematics team wrote analytic rubrics for this item based on the y -coordinates of the left and right endpoints of the line segment plotted—that is, the y -coordinates of the points on the line whose x -coordinates are 30.5 and 36.0 (see Figure 18). If y_L and y_R denote the y -coordinates of the left and right endpoints, respectively, then, according to these analytic rubrics, a response is scored as correct if $60 \leq y_L \leq 61$ and

$67 \leq y_R \leq 68$ or if $61 \leq y_L \leq 62$ and $66 \leq y_R \leq 67$. (The response in Figure 18 satisfies the first of these two conditions and, hence, would be scored as correct.) These scoring rubrics can be easily encoded in an m-rater model, so it was decided to base the scoring model on these rubrics rather than on the d -metric and human scores.

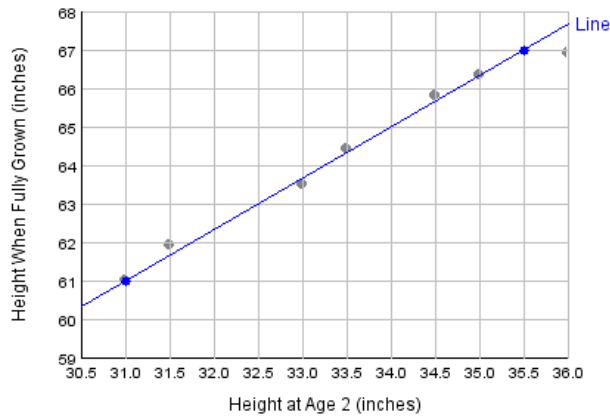


Figure 18. A sample response to *Heights and Growth Part 2 Item 4*.

One final note regarding this item: although it is not visible, the graph editor was configured incorrectly when the item was authored on-screen. As can be seen from Figure 14 and Figure 18, the viewing window is defined by the inequalities $30.5 \leq x \leq 36.0$, with gridlines every 0.5 unit, and $59 \leq y \leq 68$, with gridlines every 1 unit. But, in the underlying code in the graph editor, the parameters were set so that x ranged from 0 to 11 with gridlines every 1 unit, and y ranged from 0 to 234 with gridlines every 26 units. Responses were reported to m-rater based on the configuration hidden in the code and not the configuration that was visible on the screen; thus, a transformation had to be applied to the response data before those data could be scored. Applying this transformation was not difficult, but it did increase the chance of error. For a subsequent administration, the item was reauthored with the correct configuration in the underlying code.

Implications for Further Research

The work described in this paper has several implications regarding further research:

Examinee Performance When Examinees Must Select the Viewing Window

How well will examinees perform when they must select the viewing window in a graph item? What will be the effect on the examinee scores? Will items requiring that the viewing window be set be as informative as items with a fixed viewing window? Care must be taken the first time this feature is used. It might be useful to create parallel versions of tasks, with one version requiring that the examinee set the viewing window and the other version not.

How to Score Curves?

The graph editor allows examinees to enter curves by plotting selected points; the editor then connects the points with a smooth curve. But, the graphing key does not score general curves; it only scores graphs of quadratic functions. This graphing key was based on a key developed for a previous state assessment program, which only asked for the plotting of points, lines, graphs of piecewise-linear functions, and graphs of quadratic functions. How should general curves be scored? What features of the curve do we want to score? Do we need the capability of graphing curves that have cusps? We need to answer these questions in the context of specific design considerations and specific items. We may soon have an opportunity to look at these questions in CBAL as the mathematics component expands its efforts from middle school to high school algebra.

How to Score Line-of-Best-Fit Questions?

What is the best way to do this scoring? If the item has analytic rubrics, then those rubrics can be encoded in an m-rater scoring model. If the item has holistic rubrics, it may be possible to use the human scores to set cut points on the d -metric, but the reliability of human scores appears to vary widely for this type of item.

M-rater Advisories: Too Many False Positives?

M-rater advisories can now be issued for certain categories of responses which m-rater either cannot score or would always score as 0. These categories include responses that m-rater cannot parse, responses for which the equation editor cannot produce content MathML, and responses for which the equation editor is likely to produce incorrect content MathML. Human

scoring will be recommended on the assumption that humans will often choose to give the response some credit.

What Is the Best Interface for Entering Equations?

Many of the problems leading to advisories are due to problems with the equation editor that CBAL currently uses. Are there better choices for an interface in which examinees can enter their responses? As stated earlier, a different equation editor is being investigated for use with CBAL; this editor can be configured to restrict examinee input. Additionally, the advent of tablet computers and of handwriting-recognition software suggests other possibilities.

Does Entering Mathematics Questions Online Change the Construct Being Tested?

When the response to a task requires writing an equation, how does asking examinees to enter the equation on a computer instead of writing it on paper change what is being measured? Gallagher, Bennett, Cahalan, and Rock (2002) found no evidence that the use of an equation editor negatively affected student performance, but the students in their study were prospective graduate students in quantitative fields. So far, there seems to have been little attention paid to the issue of how middle school students relate to equation editors.

Conclusion

Several important enhancements were made to m-rater in 2012—the scoring engine was enhanced with the addition of a computer algebra system, enhancements were made to the graph editor to correct a problem with the graphing of smooth functions and to allow assessment specialists to require examinees to determine the viewing window before responding to an item, and advisories are now issued when m-rater either cannot score a response or is likely to produce an incorrect score due to MathML conversion problems.

When examinees determine the viewing window in the graph editor, snap-to-grid must be disabled because examinees cannot be expected to necessarily select a viewing window for which the points to be plotted are always grid points. In 2012, the CBAL initiative administered for the first time a graphing item with snap-to-grid disabled. The scoring rubrics needed to specify tolerances for each point to be plotted (an x -tolerance and a y -tolerance). These tolerances were selected by test developers based on the distribution of responses for a sample of examinees.

Line-of-best-fit items are another class of questions whose automated scoring requires the examination of a sample of responses. Over the past several years, we have tried various methods of scoring these items. The items are easy to score if there are analytic rubrics that give determinative scores for each response. In the absence of such rubrics, the best method may be to use human scores to establish cut points on the d -metric, although the reliability of this method depends on the reliability of the human scoring.

As CBAL expands its focus to include high school algebra, and as Common Core assessments are developed, new task types with their own challenges will emerge. One of these challenges will be to develop the specifications for scoring graphs of smooth functions. Another challenge, both for new task types and for old task types, is the proper use of m-rater advisories. Finally, the search continues for the best interface for capturing student responses when the response is an equation and for a better understanding of the measurement implications of asking examinees to respond on a computer instead of on paper.

References

- Ball, L., & Stacey, K. (2001). New literacies for mathematics: A new view of solving equations. *The Mathematics Educator*, 6(1), 55–62.
- Bennett, R. E. (2010). Cognitively Based Assessment of, for, and as Learning (CBAL): A preliminary theory of action for summative and formative assessment. *Measurement: Interdisciplinary Research & Perspective*, 8(2), 70–91.
- 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., Steffen, M., Singley, M. K., Morley, M., & Jacquemin, D. (1997). Evaluating an automatically scorable, open-ended response type for measuring mathematical reasoning in computer-adaptive tests. *Journal of Educational Measurement*, 34, 162–176.
- Breyer, F. J., Williams, F. E., Fife, J. H., & Lewis, C. (2012). *Comparing m-rater and human scorers on CBAL mathematics constructed-response items*. Princeton, NJ: Educational Testing Service.
- Butland, J. (1980). A method of interpolating reasonable-shaped curves through any data. *Proc. Computer Graphics*, 80, 409–422.
- Cayton-Hodges, G. A., Marquez, E., van Rinj, P., Keehner, M., Laitusis, C., Zapata-Rivera, D., ... Hakkinen, M. T (2012, May). *Technology enhanced assessments in mathematics and beyond: Strengths, challenges, and future directions*. Paper presented at the Invitational Research Symposium on Technology Enhanced Assessments, Washington, D.C.
Retrieved from http://www.k12center.org/events/research_meetings/tea.html
- Denny, J. K. (2013). SAGE: Open source mathematics software system. *The College Mathematics Journal*, 44(2), 149–155.
- Doerr, H. M., & Zangor, R. (2000). Creating meaning for and with the graphing calculator. *Educational Studies in Mathematics*, 41(2), 143–163.

- Dougherty, R. L., Edelman, A., & Hyman, J. M. (1989). Nonnegativity-, monotonicity-, or convexity-preserving cubic and quintic Hermite interpolation. *Mathematics of Computation*, 52(186), 471–494.
- Drijvers, P. H. (2003). *Learning algebra in a computer algebra environment: Design research on the understanding of the concept of parameter*. Retrieved from <http://igitur-archive.library.uu.nl/dissertations/2003-0925-101838/inhoud.htm>
- Drijvers, P., & Trouche, L. (2008). From artifacts to instruments: A theoretical framework behind the orchestra metaphor. In G. W. Blume & M. K. Heid (Eds.), *Research on technology and the teaching and learning of mathematics: Vol. 2. Cases and perspectives* (pp. 363–392). Charlotte, NC: Information Age.
- Fife, J. H. (2011). *Automated scoring of CBAL mathematics tasks with m-rater* (Research Memorandum No. RM-11-12). Princeton, NJ: Educational Testing Service.
- Fife, J. H. (2012). *How to write m-rater scoring models for numeric, equation, and graph items*. Princeton, NJ: Educational Testing Service.
- Fife, J. H., Graf, E. A., & Ohls, S. (2011). *Constructed-response mathematics tasks study* (Research Report No. RR-11-35). Princeton, NJ: Educational Testing Service.
- Fife, J. H., Graf, E. A., Ohls, S., & Marquez, E. (2008). *Identifying common misconceptions: An analysis of the mathematics intervention module (MIM) data* (Research Memorandum No. RM-08-16). Princeton, NJ: Educational Testing Service.
- Fritsch, F. N., & Carlson, R. E. (1980). Monotone piecewise cubic interpolation. *SIAM Journal of Numerical Analysis*, 17(2), 238–246.
- Gallagher, A., Bennett, R. E., Cahalan, C., & Rock, D. A. (2002). Validity and fairness in technology-based assessment: Detecting construct-irrelevant variance in an open-ended, computerized mathematics task. *Educational Assessment*, 8(1), 27–41.
- Galochkin, D. (2011). *Symbolic math key for c-rater math item evaluation*. Unpublished manuscript, Educational Testing Service, Princeton, NJ.
- Graf, E. A. (2009). *Defining mathematics competency in the service of cognitively based assessment for grades 6 through 8* (Research Report No. RR-09-42). Princeton, NJ: Educational Testing Service.
- Graf, E. A., Harris, K., Marquez, E., Fife, J., & Redman, M. (2009). *Cognitively Based Assessment of, for, and as Learning (CBAL) in mathematics: A design and first steps*

- toward implementation (Research Memorandum No. RM-09-07). Princeton, NJ: Educational Testing Service.
- 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.
- Hyman, J. M. (1983). Accurate monotonicity preserving cubic interpolation. *SIAM Journal of Statistical Computing*, 4(4), 645–654.
- Kramarski, B., & Hirsch, C. (2003). Using computer algebra systems in mathematical classrooms. *Journal of Computer Assisted Learning*, 19, 35–45.
- Lukoff, B. (2010). *The design and validation of an automatically-scored constructed-response item type for measuring graphical representational Skill* (Unpublished doctoral dissertation). Stanford University, School of Education, Stanford, CA.
- Pierce, R., & Stacey, K. (2002). Algebraic insight: The algebra needed to use computer algebra systems. *Mathematics Teacher*, 95, 622–627.
- Pointon, A., & Sangwin, C. J. (2003). *An analysis of undergraduate core material in the light of hand held computer algebra systems*. Retrieved from <http://web.mat.bham.ac.uk/C.J.Sangwin/Publications/handheldcas.pdf>
- Sangwin, C. J. (2002). *Assessing higher skills with computer algebra marking*. Retrieved from http://www.jisc.ac.uk/media/documents/techwatch/tsw_02-04.pdf
- Sangwin, C. J. (2003). *Assessing mathematics automatically using computer algebra and the Internet*. Retrieved from <http://web.mat.bham.ac.uk/C.J.Sangwin/Publications/tma03.pdf>
- Stacey, K. (2005). Accessing results from research on technology in mathematics education. *Australian Senior Math Journal*, 19(1), 8–15.
- Stacey, K., McCrae, B., Chick, H., Asp, G., & Leigh-Lancaster, D. (2000). Research-led policy change for technology-active senior mathematics assessment. In J. Bana & A. Chapman (Eds.), *Mathematics education beyond 2000. Proceedings of the 23rd annual conference on the Mathematics Education Research Group of Australasia* (pp. 572–579). Perth, Western Australia: MERGA.
- Sukkariéh, J. Z., & Blackmore, J. (2009). C-rater: Automatic content scoring for short constructed responses. In *Proceedings of the Twenty-Second International FLAIRS Conference* (pp. 290–295). Menlo Park, CA: AAAI Press.

Notes

- ¹ Alchemist is ETS's online software tool for writing m-rater scoring models.
- ² MathML is a markup language, similar to XML or HTML, that can capture the presentation and the content of a mathematical expression. There are actually two versions of MathML. Presentation MathML captures how an expression is displayed without regard to its mathematical content; content MathML captures the mathematical meaning of an expression without regard to its display. CBAL uses a third-party equation editor in which examinees enter their responses. The editor records the response in presentation MathML, converts the presentation MathML into content MathML, and then returns both the presentation and the content MathML. For scoring, the content MathML is converted into a form that m-rater can score.
- ³ As explained earlier, snap-to-grid will need to be disabled whenever the viewing window is to be determined by the examinee, because, depending on how the examinee selects the gridlines, the correct responses (the responses that the examinee needs to plot) may not correspond to grid points.
- ⁴ The graph editor was configured so that the students could plot the points in any order, and the points are not labeled to indicate which point goes with which point. As a result, the first point plotted by an examinee was not necessarily intended to be the point (12,168). Furthermore, examinees could plot more than four points, and some did plot more than four points. For the purposes of this analysis, I only looked at the first four points each examinee plotted, and I assumed that the point whose x -coordinate was closest to 12 was the intended (12,168).
- ⁵ Rater 1 and Rater 2 were the same two individuals for all responses.

Appendix
Proofs of Theorems 1 and 2

Theorems 1 and 2 follow from the following theorem:

Theorem 0. Let $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$ be $n+1$ points in the plane, with $x_0 < x_1 < \dots < x_n$. Let f be a Hermite cubic spline through the points such that $m_i = 0$ if $s_i s_{i+1} \leq 0$ for $i = 1, \dots, n-1$. (If $s_i s_{i+1} > 0$, then m_i can be any number between s_i and s_{i+1} .) Let $k = 1, \dots, n-1$ be such that $y_k > y_{k-1}$ and $y_k > y_{k+1}$.

(a) If $m_k > 3s_k$, the cubic polynomial f_k defined on the interval $[x_{k-1}, x_k]$ has a local minimum at x_k .

(b) If $m_k \leq 3s_k$, the cubic polynomial f_k defined on the interval $[x_{k-1}, x_k]$ has a local maximum at x_k .

(c) If $m_{k+1} < 3s_{k+1}$, the cubic polynomial f_{k+1} defined on the interval $[x_k, x_{k+1}]$ has a local minimum at x_k .

(d) If $m_{k+1} \geq 3s_{k+1}$, the cubic polynomial f_{k+1} defined on the interval $[x_k, x_{k+1}]$ has a local maximum at x_k .

Before proving Theorem 0, I mention the following lemma and its corollary:

Lemma. If a quadratic polynomial f has two roots, $x = r$ and $x = s$, then $f'((r+s)/2) = 0$.

Corollary. If a cubic polynomial has two critical points, $x = r$ and $x = s$, then

$$f''((r+s)/2) = 0.$$

Proof of Theorem 0. Because $y_k > y_{k-1}$ and $y_k > y_{k+1}$, it follows that $s_k > 0$ and $s_{k+1} < 0$. Hence, $s_k s_{k+1} < 0$ and, therefore, $f'_k(x_k) = f'_{k+1}(x_k) = m_k = 0$. Because $f_i''(x) = 2c_i + 6d_i(x - x_i)$, it follows that $f_k''(x_k) = 2c_k + 6d_k h_k$ and $f_{k+1}''(x_k) = 2c_{k+1}$. Because $m_k = 0$, it follows from the formulas for c_i and d_i given previously that $c_k = (3s_k - 2m_{k-1})/h_k$, $c_{k+1} = (3s_{k+1} - m_{k+1})/h_{k+1}$, and $d_k = (m_{k-1} - 2s_k)/h_k^2$. Therefore,

$$f_k''(x_k) = \frac{2m_{k-1} - 6s_k}{h_k} \text{ and } f_{k+1}''(x_k) = \frac{6s_{k+1} - 2m_{k+1}}{h_{k+1}}.$$

If $m_{k-1} > 3s_k$, it follows that $f_k''(x_k) > 0$. Because $f_k'(x_k) = 0$ and $f_k''(x_k) > 0$, it follows from the second derivative test that f_k has a local minimum at x_k . This proves part (a).

If $m_{k-1} < 3s_k$, it follows that $f_k''(x_k) < 0$. Because $f_k'(x_k) = 0$ and $f_k''(x_k) < 0$, it follows from the second derivative test, again, that f_k has a local minimum at x_k .

If $m_{k-1} = 3s_k$, then $f_k''(x_k) = 0$ and the second derivative test does not apply, but I can still show that, restricted to the interval $[x_{k-1}, x_k]$, the cubic polynomial f_k has a local maximum at x_k . I shall show this by contradiction; suppose that it does not. The polynomial f_k attains its maximum on the interval $[x_{k-1}, x_k]$ at a point $u \in [x_{k-1}, x_k]$. If $u = x_k$, then f_k would have a local maximum at x_k . Because I am assuming it does not, it follows that $u \neq x_k$. Because

$f_k(x_{k-1}) = y_{k-1} < y_k = f_k(x_k)$, the polynomial cannot attain its maximum at x_{k-1} , either, and hence $u \neq x_{k-1}$. Therefore, $x_{k-1} < u < x_k$. Hence, $f_k'(u) = 0$. Because $f_k'(x_k) = 0$, it follows from the corollary to the lemma that $f_k''((u + x_k)/2) = 0$. Because $f_k''(x_k) = 0$ and $(u + x_k)/2 \neq x_k$, this implies that the linear polynomial f_k'' has two roots, which is not possible. Thus, restricted to the interval $[x_{k-1}, x_k]$, the polynomial f_k has a local maximum at x_k . This completes the proof of part (b).

The proofs of parts (c) and (d) are similar. ■

There is also a version of Theorem 0 that applies when $y_k < y_{k-1}$ and $y_k < y_{k+1}$. Its statement and proof are left to the reader.

Proof of Theorem 1. To prove part (a), suppose $k > 1$ and $s_{k-1} > 5s_k$. Because $y_k > y_{k-1}$, it follows that $s_k > 0$; hence, $s_{k-1} > 5s_k > 0$. Thus, $s_{k-1}s_k > 0$ and, therefore,

$$m_{k-1} = \frac{s_{k-1} + s_k}{2} > \frac{5s_k + s_k}{2} = 3s_k.$$

Therefore, by part (a) of Theorem 0, the cubic polynomial f_k , defined on the interval $[x_{k-1}, x_k]$, has a local minimum at x_k . It follows that the spline cannot have a local maximum at x_k .

The polynomial f_k attains a maximum on the interval $[x_{k-1}, x_k]$ at a point $u \in [x_{k-1}, x_k]$.

Because f_k has a local minimum at x_k , and because $y_{k-1} < y_k$, it follows that $x_{k-1} < u < x_k$.

Thus, the spline has a local maximum at a point $u < x_k$. This proves part (a).

The proof of part (b) is similar, using part (c) of Theorem 0.

To prove part (c), first suppose $1 < k < n-1$, $s_{k-1} \leq 5s_k$, and $s_{k+2} \geq 5s_{k+1}$. As in part (a), $s_k > 0$. If $y_{k-2} \geq y_{k-1}$, then $s_{k-1} \leq 0$ and, hence, $m_{k-1} = 0 < 3s_k$. Otherwise, $y_{k-2} < y_{k-1}$. In this case, $s_{k-1} > 0$ and an argument similar to that in the proof of part (a) shows that $m_{k-1} \leq 3s_k$.

Therefore, by part (b) of Theorem 0, the cubic polynomial f_k , defined on the interval $[x_{k-1}, x_k]$, has a local maximum at x_k .

Similarly, because $s_{k+2} \geq 5s_{k+1}$, it follows that $m_{k+1} \geq 3s_{k+1}$, and, hence, by part (d) of Theorem 0, the cubic polynomial f_{k+1} , defined on the interval $[x_k, x_{k+1}]$, has a local maximum at x_k . Therefore, the spline f has a local maximum at x_k .

Now, suppose $k=1$ and $s_3 \geq 5s_2$. Because $m_0 = 2s_1 - m_1$, $m_1 = 0$, and $s_1 > 0$, it follows that $m_0 = 2s_1 < 3s_1$. Thus, f_1 , defined on the interval $[x_0, x_1]$, has a local maximum at x_1 .

Because $s_3 \geq 5s_2$, it follows as above that the cubic polynomial f_2 has a local maximum at x_1 .

Therefore, the spline f has a local maximum at x_1 .

The case $k=n-1$ and $s_{n-1} \leq 5s_{n-1}$ is similar.

Finally, suppose the spline f has another local maximum at a point $u \in (x_{k-1}, x_{k+1})$, where $u \neq x_k$. We may assume without loss of generality that $u < x_k$. Then $x_{k-1} < u < x_k$, and, therefore, $f'_k(u) = 0$. The cubic polynomial f_k attains a minimum on the interval $[u, x_k]$ at some point $v \in [u, x_k]$. Because f_k has local maxima at u and x_k , it follows that $u < v < x_k$, and, therefore, $f'_k(v) = 0$. But this means that the quadratic polynomial f'_k has three roots, u , v , and x_k , which is not possible. Therefore, the spline f cannot have another local maximum on the interval (x_{k-1}, x_{k+1}) .

This completes the proof of part (c), and, with it, the proof of Theorem 1. ■

Proof of Theorem 2. To prove part (a), suppose that $y_k > y_{k-1}$ and $y_k > y_{k+1}$. Because $y_k > y_{k-1}$, it follows that $s_k > 0$. If $k > 1$ and $y_{k-2} \geq y_{k-1}$, then, as in the proof of Theorem 1(c), $m_{k-1} < 3s_k$.

On the other hand, if $y_{k-2} < y_{k-1}$, then $s_{k-1} > 0$ and, therefore,

$$m_{k-1} = \frac{2s_k s_{k-1}}{s_k + s_{k-1}} < \frac{2s_k (s_k + s_{k-1})}{s_k + s_{k-1}} = 2s_k < 3s_k .$$

Finally, if $k = 1$, then $m_0 = 2s_1 < 3s_1$ as before, because $m_1 = 0$. So, in all cases $m_{k-1} < 3s_k$, and, therefore, by Theorem 0, the cubic polynomial f_k , defined on the interval $[x_{k-1}, x_k]$, has a local maximum at x_k .

Similarly, the cubic polynomial f_{k+1} , defined on the interval $[x_k, x_{k+1}]$, has a local maximum at x_k . It follows that the spline f has a local maximum at x_k .

The proof that f does not have another local maximum in the interval (x_{k-1}, x_{k+1}) is identical to the corresponding proof in Theorem 1(c). This completes the proof of part (a).

The proof of part (b) is similar, using the version of Theorem 0 that is applicable when $y_k < y_{k-1}$ and $y_k > y_{k+1}$. This completes the proof of Theorem 2. ■