MathPlayer: Fully Accessible Web-based Math

Neil Soiffer
neils@dessci.com
Design Science

Ricky Cancro
rickyc@saltire.com
Saltire Software

Dana Li
stitchlowyni@hotmail.com
Design Science

Abstract: *MathPlayer is an add-in to Microsoft Internet Explorer (IE) that renders MathML visually. It also contains a number of features that make mathematical expressions accessible to people with print-disabilities (blindness, low-vision, dyslexia, etc). These include the ability to convert the math into customizable speech or braille. MathPlayer integrates with many screen readers including JAWS and Window-Eyes. MathPlayer also works with a number of TextHELP!'s learning disabilities products.*

1. Introduction

Mathematical expressions used on web pages typically take one of three forms: images, low-level HTML tagging, or MathML[27]. The use of images is very common, but suffers from problems such as the baseline of the math not matching the surrounding text, font sizes and colors not matching the user's preference/rest of the document, and poor quality if the page is printed. The use of low level HTML tagging for italics, superscripts, and subscripts produces readable but low-quality results because spacing around operators typically does not follow the rules of standard mathematical typography. Furthermore, this technique is limited to simple math expressions. MathML is the World Wide Web Consortium's (W3C) recommendation for including math in XML. MathML offers many advantages over the other techniques including:

- Browsers will automatically match the font, size, color, and baseline of the expression with the rest of the document;
- Mathematical structure is maintained so that the math can be searched, translated into many formats including speech, and can be used in computation;
- Programs and users can interact with the expression to enlarge it, show or elide terms in it, or synchronously highlight it while it is spoken.

Maple[21], Mathematica[36], and most other mathematical software support importing and/or exporting MathML. This allows people to copy expressions on Web pages into mathematical software for further computation, graphing, or other exploration.

The only way to make an expression accessible if an image is used is to use an `alt` attribute on the `img` tag. Unfortunately, alternative text rarely accompanies math expressions and when it does, it often fails to disambiguate the expression. Eg, "x to the n plus 1" could be either x^{n+1} or $x^n + 1$. Furthermore, the alternative text can not be translated into braille because braille has separate literary and math code; a proper translation requires knowledge of the expression.

The focus of this note is on MathPlayer's ability to render MathML and support access to the expression for people with print disabilities (blindness, low-vision, and certain learning disabilities such as dyslexia). We begin with some background on the usefulness of using MathML both for accessibility and for general use such as computation. The following section provides an overview of earlier work accomplished by other researchers working on math accessibility. This work inspired many of the ideas contained in MathPlayer. A major difference between MathPlayer and previous work is that MathPlayer is designed to integrate into user's existing software (both browsers and AT) rather than having them use a standalone tool when they encounter math in a document. The remainder of this note describes MathPlayer's accessibility features and work that will be included in future versions of MathPlayer.

2. The Power of MathML

MathML is used in a number of different applications such as in XML-based work flows (eg, publishing, patent offices[15]), and in Web pages. In Web pages, the use of MathML has a number of advantages over images. By using MathML, the math:

- is displayed with a size, font, and foreground and background color that automatically matches the rest of the document;
- aligns with the rest of the text;
- is accessible;
- is searchable;
- can be copied and pasted into numerous applications for re-use or computation.

Although there are an infinite number of mathematical expressions, it is extremely common to find certain expressions used repeatedly. The re-use of math in a Web page can occur when the MathML for an equation is copied and pasted into a document (homework, journal paper…). It can also occur if the reader wishes to search for the expression[24]. Although general purpose search engines are an essential part of the Web, they have not expanded to searching math yet. However, a number of research efforts have had some success with searching for expressions, and at least one (MathDex[7]) is available for general use.

Another example of reuse that MathML affords is copying and pasting an expression into one of the many applications that import MathML for computation or plotting[35]. This can be useful to further explore a problem in a research paper or simply to avoid the tedium of retyping a homework problem into the computation or plotting system. Most of these systems also support export of MathML, so the results can be posted as a Web page without retyping the math.

Numerous classes have made their materials available online. These range from class/tutorials in Algebra[9] to Calculus courses[18] to science and engineering courses[4]. The later references include links to Mathematica[36] notebooks and Java applets to enhance the written material. Distance learning courses are another source of online material. Both Blackboard and WebCT[2] have some support for MathML in their system, but for authoring material and for student responses on tests, etc. Schools that use these systems typically are required to abide by federal, state, and university mandates for accessibility. The emphasis on accessibility of online material is a recent phenomenon. In the last few years some large universities such as those in the California State University system have launched major efforts[34] to ensure that all of their materials are accessible.

3. Prior Work

A number of projects have focused on math accessibility. Early work includes MathTalk[30], MAVIS[16], AsTeR[26], and TRIANGLE[12]. MathTalk was envisioned as a standalone product as was MAVIS, although MAVIS's Nemeth Code[25] braille translation was integrated with Scientific Notebook[20].

TeX is a common format for writing documents containing math. According to a study by Design Science[33], TeX accounts for about 30% of all material submitted to scientific, technical, and math publishers (the rest is Microsoft Word+MathType or Equation Editor). Because TeX is a text-based linear format, it is marginally accessible with standard AT software. However, TeX is not easy to learn and sonification of its commands can be hard to understand. AsTeR is T. V. Raman's seminal work on math accessibility. It reads LaTeX documents and converts them into a form that can be spoken and navigated. AsTeR is built on top of EMACS, which has limited its adoption. TRIANGLE uses a notation that is similar to TeX, but with special characters replacing TeX's math commands, which makes it easier to learn and understand.

More recent work has involved MathML. Two larger accessibility projects that involve MathML are UMA[17] and Lambda[19]. Both projects have a strong focus on two-way translation between MathML and multiple braille math codes. They also include some standalone software (e.g., MathGenie[29]) for voicing, navigating, and/or translating math to braille.

4. MathPlayer

MathPlayer is a free plug-in for IE that displays MathML in Web pages. MathPlayer's accessibility features were introduced in MathPlayer 2.0 in 2004 and were enhanced in a 2.1 release in early 2007. MathPlayer supports speech generation and seamlessly works with many AT software products such as JAWS[10], Window-Eyes[14], and Read&Write[32]. We are working with several vendors of DAISY players[5] to incorporate MathPlayer into their products so that those players fully implement the 2007 MathML extension to DAISY[28].

4.1 MathPlayer 2.1

In a Web page, MathPlayer displays math that matches the surrounding document's font properties such as its size and foreground and background color. If the user adjusts the font properties, the math will be readjusted also.

Individual expressions can be magnified. This is useful because typically a document will be sized so that the text can be read comfortably. However, mathematical typography uses multiple font sizes. For example, subscripts and superscripts are typically reduced in size. Because of this, some parts of an expression may not be easily readable even though MathPlayer uses a font size that matches the rest of the document. Magnifying an individual expression allows the user to read small scripts without enlarging the entire document. This is shown below:

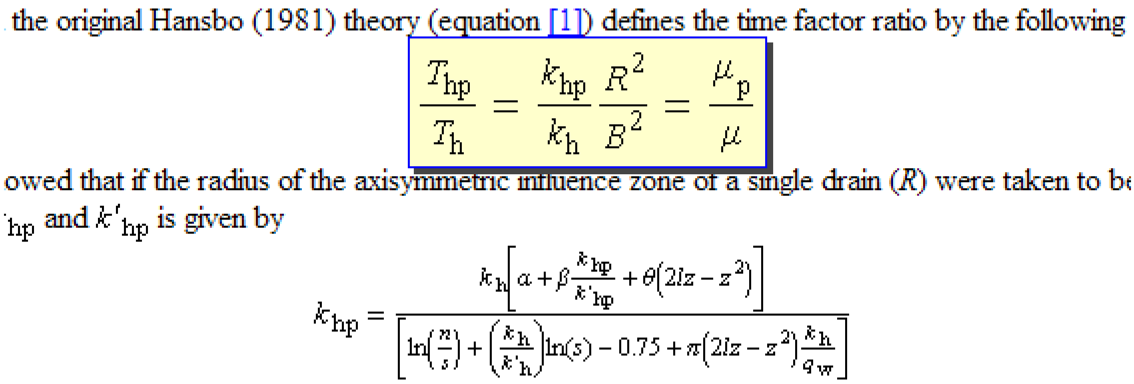

Figure 1: Example of Magnification in MathPlayer

For accessibility, MathPlayer's most important feature is its ability to convert MathML into a speech string that can be spoken by the Text to Speech (TTS) Engine used by the AT software. The

MathPlayer interface for speech currently being used by most AT software restricts communication to simple text strings – no speech engine control strings can be used for prosody. However, two differing amounts of pauses are achieved by the judicious use of commas and periods. MathPlayer's speech interface supports SAPI4[22], SAPI5[23], and SSML[3] tagged strings and we are working with AT vendors to make use of those interfaces.

For some people with low vision or learning disabilities, their reading is enhanced by listening to what they are reading and having the words highlighted as they are being spoken. This is referred to as *synchronized highlighting*. Many high-end screen magnifiers and learning disability tools make use of synchronized highlighting. We have worked with several vendors to integrate MathPlayer's synchronized highlighting into their products. Two examples of MathPlayer's display using synchronized highlighting are shown below:

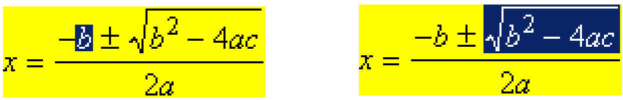

Figure 2: Two examples of synchronized highlighting

Several studies[8] have shown that synchronized highlighting is helpful for text, but none has been done for math and it is currently not known whether highlighting the notational structures such as the square root in the second example is useful or not. To allow for experimentation, MathPlayer allows AT software to control the amount of highlighting that MathPlayer does.

MathPlayer supports several forms of navigation, although AT vendors currently do not make use of them. The supported forms include moving through the tree/expression structure of the math, moving by words that are spoken, and a hybrid approach. We are working with several vendors to determine the best interface for their needs so that MathPlayer's navigation options are available to users.

Braille is a linear notation. It typically uses either a 6 dot pattern (2x3) or sometimes an 8 dot pattern (2x4). This means that a single braille character is not capable of representing the vast number of mathematical symbols that are used, let alone the 2-dimensional notation used in math. Various notations have been developed that use multiple braille characters to represent math symbols. Unlike printed mathematical notation which has an almost universal acceptance, various countries and groups within a country have adopted their own braille notation.

MathPlayer 2.1 supports the Dotsplus[11] notation. Dotsplus is a 2D braille notation useful for embossing an entire page. Dotsplus uses braille for numbers and letters, but uses graphics (tightly-spaced raised dots) to represent other characters (eg, $\pm$) and uses 2-dimensional notation for the math. This makes Dotsplus relatively easy to learn, but is somewhat confusing for experienced braille readers who are not used to moving their fingers vertically to read braille. It also means that Dotsplus cannot be authored directly.

4.2 Future Features

Two important features have been implemented since the last release: support for many different braille notations and customizable speech. In addition to this, line breaking has been added. Line breaking is important to accessibility because the use of large fonts may cause lines to be too large to fit on a single line and need to wrap sensibly just as they do for text.

MathPlayer is able to generate braille math in a number of different formats. DotsPlus was mentioned above, but MathPlayer can now also generate other braille math codes suitable for display on refreshable braille displays. Refreshable braille displays are devices that attach to the front of a keyboard and have a series of pins arranged in 2x4 braille cells; typically they are 40 – 80 braille characters wide. These pins are raised and lowered according to what words/lines of text are being read. MathPlayer defines a simple COM interface that is easy to implement so that potentially any MathML-to-braille translator can work with it. To date, only one translator has been hooked up (it generates Nemeth[25] code), but based on discussions with other translator authors we expect at least two more translators to implement the interface in the coming year.

There are many ways to speak a math expression. Speech varies by local custom, subject area, and expertise/familiarity of the listener. Expertise/familiarity affects the amount of verbosity and ambiguity/disambiguity that may be desired. Ambiguity often happens with fewer words, but context or knowledge of the subject may allow the listener to disambiguate what was said. Hence, the ability to customize what is spoken is important as it affects both the understandability and the efficiency of the speech. MathPlayer uses a rule-based system to generate different styles of speech. In addition to the text of the speech, embedded speech cues and synchronization points can be generated for SAPI4, SAPI5, SSML and other TTS standards.

An important side effect of allowing customization is that MathPlayer's speech can be generated in different languages. MathPlayer uses the language of the document to choose the language used for speaking the math. Speech is generated via a set of rules that match MathML and convert it to text.

Two different styles of speech have been implemented in English so far: one based on MathPlayer's current style of speech and one based on MathSpeak[13]. Implementing these styles of speech brought about some unique challenges. For example, over 2,800 Unicode characters have been mapped to spoken text. While some characters are simple to put into words (e.g., Greek letters) others are not (e.g.,⥯).

As well as implementing different speech styles, translations for Spanish and Chinese have also been made. Languages often have regional variations that affect speech for mathematical expressions. For example, France and Belgium use different words for the numbers 80-99 and Mexico and Spain use different words for "over" when speaking a fraction. Asian languages such as Chinese and Japanese differ from Western languages in more than just the words used. For example, in Chinese, the denominator is spoken before the numerator in a fraction; many other mathematical notations are also spoken in an order different from that of the Western Languages. These differences are easily handled by MathPlayer's rule-based approach to speech.

The current release of MathPlayer only works with XHTML documents. However, PDF distribution of documents is quite common and we have developed an Adobe Acrobat plug-in that accesses MathML in a PDF document and does most of what MathPlayer for IE does. The format that MathPlayer uses is part of a draft specification by AIIM to define a Universal Access PDF document format standard (PDF/UA[1]) that includes math.

5. Future Work

MathPlayer's speech rules are based upon a pattern matcher/rule system. The rules are able to specify synchronization points and prosody in addition to text to speak. The rules provide a great deal of flexibility and allow users to match structures such as limits and integrals so that they are spoken in the customary manner rather than treating them as general expressions with limits and/or scripts. The downside to this power is that learning the rule language is too complicated for normal users who may just want to change the words used for a symbol or the words spoken for beginning/ending a 2D notation such as a fraction. We are considering methods that allow rule writers to specify changeable parts of the right hand side of a rule. These specifications would be automatically picked up and placed into an easy to use user dialog in MathPlayer.

We are also pursuing making math accessible in other formats. As previously mentioned, we have developed a prototype that works with Adobe Acrobat if MathML is embedded in the PDF. We modified MathType[6] so that it can generate PDF with embedded MathML. However, considerable work remains to make the modification robust. An even more powerful technique for PDF accessibility would be to use OCR-like technology to find and interpret math in PDF documents. Although more error-prone than authoring it correctly in the first place, an OCR-based technique would work for the vast number of legacy PDF documents. There has been some promising work in this area by the team working on InftyReader[31].

Another common format for distributing documents is Microsoft Word. We are investigating methods to make mathematical expressions in Word and PowerPoint accessible. MathType can translate its math expressions into MathML. Once they are in MathML, we can leverage the work done for MathPlayer in IE to speak, synchronize, and navigate the expression.

No studies have been done on effectiveness of synchronized highlighting for math, but experts in the field have said they feel it should be about as effective as it is for literary text. We are part of a Department of Education grant with the University of Kentucky that will test the effectiveness of MathPlayer and synchronized highlighting in several schools in Kentucky. We hope to report on that work in 2009.

6. Acknowledgements

Work on MathPlayer was supported in part by a SBIR grant from the National Science Foundation and by the Portland Saturday Academy whose summer intern program brought Dana Li to Design Science. We would also like to thank Eduardo Tabacman for his assistance in the Spanish translation.

7. References

[1] Association for Information and Image Management (AIIM).*PDF/UA, Universal Accessibility*, from http://www.aiim.org/standards.asp?ID=27861

[2] Blackboard.*Blackboard & WebCT*, from http://www.webct.com/

[3] Burnett, D. C., Walker, M. R., & Hunt, A. (2004). *Speech Synthesis Markup Language (SSML) Version 1.0*, from http://www.w3.org/TR/speech-synthesis/

[4] Carter, C. (2005). *Mathematics for Materials Scientists and Engineers*, from http://ocw.mit.edu/OcwWeb/Materials-Science-and-Engineering/3-016Fall-2005/CourseHome/index.htm

[5] DAISY Consortium.*DAISY Tools*, from http://www.daisy.org/tools/

[6] Design Science.*MathType*, from www.dessci.com/mathtype

[7] Design Science. (2006). *MathDex*, from www.mathdex.com

[8] Elkind, J. (1993). Using Computer-Based Readers to Improve Reading Comprehension of Students with Dyslexia. *Annals of Dyslexia*, *43*, 238-259.

[9] Fisher, C. (2007). *One Mathematical Cat, Please!*, from http://www.onemathematicalcat.org/

[10] Freedom Scientific.*JAWS for Windows*, from www.freedomscientific.com

[11] Gardner, J. (1993). *Dotsplus---better than Braille?* Paper presented at the International Conference on Technology and Persons with Disabilities, Los Angeles, CA.

[12] Gardner, J., Stewart, R., Francioni, J., & Smith, A. (2002). *TIGER, AGC, AND WIN-TRIANGLE, REMOVING THE BARRIER TO SEM EDUCATION.* Paper presented at the CSUN International Conference on Technology and Persons with Disabilities, Los Angeles, CA.

[13] gh.*MathSpeak*, from http://www.gh-mathspeak.com/

[14] GW Micro.*Window-Eyes*, from www.gwmicro.com

[15] Karleen, P. (2000). *A Case Study of the Use of MathML at the U.S. Patent Office.* Paper presented at the MathML and Math on the Web, Champaign, Il.

[16] Karshmer, A. I., Gupta, G., Geiiger, S., & Weaver, C. (1998). Reading and writing mathematics: the MAVIS project In *Proceedings of the third international ACM conference on Assistive technologies* (pp. 136-143). Marina del Rey, California, United States ACM Press.

[17] Karshmer, A. I., Gupta, G., Pontelli, E., Miesenberger, K., Ammalai, N., Gopal, D., et al. (2004). UMA: a system for universal mathematics accessibility. In *Proceedings of the ACM SIGACCESS conference on Computers and accessibility* (pp. 55-62). Atlanta, GA, USA: ACM Press.

[18] Kleitman, D. J. (2005). *18.013A Calculus with Applications*, from http://ocw.mit.edu/OcwWeb/Mathematics/18-013ASpring-2005/CourseHome/index.htm

[19] *Lambda Project*. from http://www.lambdaproject.org

[20] MacKichan Software.*Scientific Notebook*, from www.mackichan.com

[21] MapleSoft. (2007). Maple: MapleSoft.

[22] Microsoft.*Speech API 4.0 SDK Documentation*, from http://www.microsoft.com/speech/download/old/sdk40a.asp

[23] Microsoft.*Speech API 5.1 SDK Documentation*, from http://www.microsoft.com/speech/download/sdk51/

[24] Miner, R. The Importance of MathML to Mathematics Communication. *Notices of the AMS, 52*(5), 532-538.

[25] Nemeth, A. (1972). *The Nemeth Braille Code for Mathematics and Science Notation* (1972 Revision ed.). Frankfort: American Printing House for the Blind.

[26] Raman, T. V. (1994). *Audio System for Technical Readings*: Cornell University.

[27] Ron Ausbrooks, S. B., David Carlisle, Stéphane Dalmas, Stan Devitt, Angel Diaz, Max Froumentin, Roger Hunter, Patrick Ion, Michael Kohlhase, Robert Miner, Nico Poppelier, Bruce Smith, Neil Soiffer, Robert Sutor, Stephen Watt. (2003). *Mathematical Markup Language (MathML) Version 2.0 (Second Edition)*, from http://www.w3.org/TR/2003/REC-MathML2-20031021/

[28] Soiffer, N., & Kahl, K. (2007). *Specification for a Digital Talking Book Modular Extension for Mathematics*, from www.daisy.org/projects/mathml/mathml-in-daisy-spec.html

[29] Stanley, P., & Karshmer, A. I. (2006). *Translating MathML into Nemeth Braille code.* Paper presented at the International Conference on Computers Helping People (ICCHP), Linz, Austria.

[30] Stevens, R. D., Edwards, A. D. N., & Harling, P. A. (1997). Access to mathematics for visually disabled students through multi-modal interaction. *Human-Computer Interaction (Special issue on Multimodal Interfaces)*, 47-92.

[31] Suzuki, M., Tamari, F., Fukuda, R., Uchida, S., & Kanahori, T. (2003). *INFTY: an integrated OCR system for mathematical documents.* Paper presented at the Proceedings of the 2003 ACM symposium on Document engineering, Grenoble, France.

[32] TextHELP!*Read&Write 7.1*, from www.texthelp.com

[33] Topping, P. (2006). *MathML Workflows in STM Publishing*, from http://www.dessci.com/en/reference/white_papers/mathml_workflows.htm

[34] University, C. S.*Accessible Technology Initiative*, from http://www.calstate.edu/accessibility

[35] W3C.*MathML Software*, from http://www.w3.org/Math/implementations.html

[36] Wolfram, S. (2003). *The Mathematica Book*: Wolfram Media, Incorporated.