Steve Maddox

Supporting Students
with **Disabilities** series

Mathematical equations in Braille

Steve Maddox
School of Physics and Astronomy
University of Nottingham
steve.maddox@nottingham.ac.uk

Introduction

Physics lecture notes include a great deal of mathematics. When two blind students enrolled in our physics course, we needed to provide a complete set of undergraduate physics notes in Braille.

Although commercially available software can produce mathematical Braille, it requires an expert user to edit and produce readable Braille. The automatic translation features work well for converting literary text to Braille, but were not reliable for any mathematics beyond the simplest school level equations.

We investigated several ways to produce mathematical Braille from LaTeX files, and chose to use the Duxbury Braille Translator (DBT) [1]. Although we found a way to produce Braille lecture notes, the system is far from reliable, and improved software is required.

Braille

A single Braille character is a 2x3 cell of dots and spaces, e.g. ⠿ ⠾ ⠷, and each letter has a single cell representation. For example, the letter d is ⠙. At the simplest level, producing a Braille document from a standard text file simply requires a font change, so that each character is replaced by the equivalent Braille cell, and the file can be printed by a Braille embosser. This character substitution is called Grade 1 Braille. Since there are only 63 possible cells, it is necessary to combine cells into pairs or triples to represent other characters. For example, a capital letter D is ⠠ ⠙. The number 4 is also ⠙, but can also be written as ⠲. If the meaning is ambiguous the letter prefix (⠰) and the number prefix (⠼) should be used, so 4d would be ⠼ ⠙ ⠰ ⠙.

This Grade 1 Braille leads to very bulky documents, so a system of contractions (or abbreviations) is used to save space. This is called Grade 2 Braille. The contractions are context dependent, with characters having different meaning if they are at the beginning, middle, or end of a word, or if they are isolated characters to be read as a whole word.

Grade 3 Braille uses even more contractions (similar to text messages), but there is no official standard.

Mathematics in Braille

There are several different standards for mathematical Braille, including: UK RNIB; Dotsplus; Nemeth; and GS Braille. Nemeth and GS Braille are American standards, not

commonly used in the UK. So, we considered two software packages, Duxbury Braille Translator (DBT) which tries to produce braille following the RNIB standards, and Dotsplus which uses its own system.

The RNIB has published a standard system for representing mathematics in Braille [2]. The system presents equations as a linear sequence of Braille cells, with a large number of extra Braille characters to represent operators and layout of the equation. The DBT software attempts to produce standard mathematical Braille, as defined by the RNIB.

As an example, the equation:

$$\int_0^x \frac{1}{2\sigma} e^{\frac{t^2}{2\sigma}} \, dx$$

would become

⠮⠡⠂⠲⠼⠣⠰⠨⠆⠌⠢⠬⠒⠩⠂⠖⠆⠴⠩⠾⠉⠙⠭

The Dotsplus [3] system retains the visual layout of normal typeset equations, using standard Braille characters for the variables, and a new set of characters which don't use the standard 6 point Braille cell. An example of this is the equation

$$|r| < 1, \qquad \sum_{n=1}^{\infty} gr^{n-1} = \frac{g}{1-r}$$

which would become

$$|r| < 1, \qquad \sum_{n=1}^{\infty} gr^{n-1} = \frac{g}{1-r}$$

The symbols for infinity, -, =, sigma and the line of fraction line are embossed as they are visually. This means that a non-standard Braille printer is required to produce the hardcopy output.

We chose to use the RNIB system because it was more similar to the Braille that the students had used in school, and also the Nottingham University Alternative Formats Service4 already had a copy of the software.

We chose to work from LaTeX files, since I had most experience with that for of typesetting equations, and at least some lecturers already had LaTeX versions of their lecture notes. Originally we thought that a further advantage would be that a standard screen reader can read simple LaTeX, and so equations would be immediately accessible without producing hard-copy Braille. Unfortunately, this was not possible in practice, because the students had more than enough work to follow the physics without having to learn the intricacies of typesetting equations in LaTeX. The students' workload was compounded because of our difficulties in producing Braille lecture notes quickly enough.

We also investigated a third option for producing Braille equations directly from LaTeX. W. Park has written a publicly available Braille style file [5] for LaTeX and python scripts to convert standard LaTeX into a printable version of Grade 2 Braille. Unfortunately these deal only with literary Braille, and do not attempt to handle mathematics at all. In principle, it is a well defined problem to convert LaTeX equations into RNIB Braille equations, but lack of time prevented us from making significant progress in modifying the style file to deal with anything but the simplest of equations. A further complication is that the output from the LaTeX is simply a postscript file. Although this can be printed to produce visual Braille, this cannot simply be sent to a Braille embosser to produce usable Braille hardcopy.

Braille Maths using Duxbury DBT

The Duxbury DBT software can read a variety of file formats, including LaTeX, for translating literary text. The Braille output for literary text is generally quite acceptable, but it can not handle many of the features of LaTeX; automatic section numbering, equation numbering, itemized lists and cross-referencing are not implemented by DBT. The software was even less reliable when dealing with equations. The initial version was missing a large number of quite common symbols, (e.g. the > symbol), but the more recent versions now recognize most of the common mathematical symbols. The main limitations now come from the fact that only a limited subset of math mode formatting is recognized. The limited implementation of LaTeX commands means that we needed rewrite the LaTeX to use only the commands that work.

LaTeX for DBT

For many lecture courses, LaTeX files were typed explicitly to be Brailled, so the LaTeX commands could be limited to the set that DBT can handle. For many other courses, we already had LaTeX files that were used to produce printable output. We found that it took almost as long to change the equations as it would to retype them. In order to speed up this conversion, I wrote a perl script which searched for the LaTeX commands that we had identified as unrecognizable by DBT, and replaced them with forms that were compatible with DBT. For example, the LaTeX command to produce a vector, $\verb|\vec{x}|$, is not implemented by DBT, but DBT can produce bold font in maths mode using $\verb|\mathbf{x}|$. The relevant line from the perl script is

```
s/\\vec\{([^\}]*)\}/\\mathbf\{$1\}/g ;
```

The script makes several dozen replacements, and speeded up the conversion from standard LaTeX to a DBT compatible version. Nevertheless, we found that the output still needs manual checking, and in many cases it was simpler to re-type the LaTeX.

Our system in practice

We start from lecturer's notes, or hand written notes from a note-taker who sits in the lectures. Then post-graduate students type the LaTeX for each module. The LaTeX is checked and forwarded to the University Alternative Formats Service, who produce the hard-copy Braille. This is then delivered to the students with added diagrams if necessary.

The system is time-consuming and labour intensive. It consists of a long chain with many critical links that can introduce delays and errors. Although it works, it is far from ideal, and failures happened on a regular basis. We need a better system.

Other options

Screen readers

Screen readers use speech synthesis software to read text from windows on the computer screen. There are several quite sophisticated packages available, but they do have their limitations. In particular, they tend not to work with Java-based windows, and they cannot currently deal with equations. There are some projects under way that aim to produce a screen reader that can read out equations, but a comprehensive package is not yet available.

Braille Displays

Braille displays are devices attached to a PC that raise and lower small pine in a large array to actively form Braille characters. They typically display 2 lines of 40 Braille cells, and use a simple one-to-one ASCII character to Braille cell conversion. Since this is not the way that Grade 2 Braille, or maths Braille works, an extra layer of software would be required to use these devices effectively.

The Fundamental Problem

The fundamental problem with converting a mathematical equation into Braille is how to linearize the representation. The same problem must be solved in order to read out an equation with a speech synthesizer. In a sense, LaTeX is already a solution, since it does use a purely linear representation for typesetting equations, but there are many unnecessary extra complications resulting from its history as a general type-setting language. MathML seems to provide a more general approach that also has the benefit of being compatible with generic web browsers, and this seems a promising way forward.

"Mathematics is not easy to convey using Braille or screen readers and although partial solutions exist, they still have limitations."

MathML

MathML is a standard that provides a low-level specification for describing maths on the web [6]. It can be rendered with many standard web browsers as shown in Fig 1 [7], although some need an additional plug-in.

MathML can use presentational tags to describe how an equation should be laid out on the page, but it can also use semantic tags, which are more closely related to the actual mathematics of the equation. As an example consider the simple equation: $x^2+4x+4=0$

Fig 1 - MathML example web site

Using presentational tags, the MathML code is:
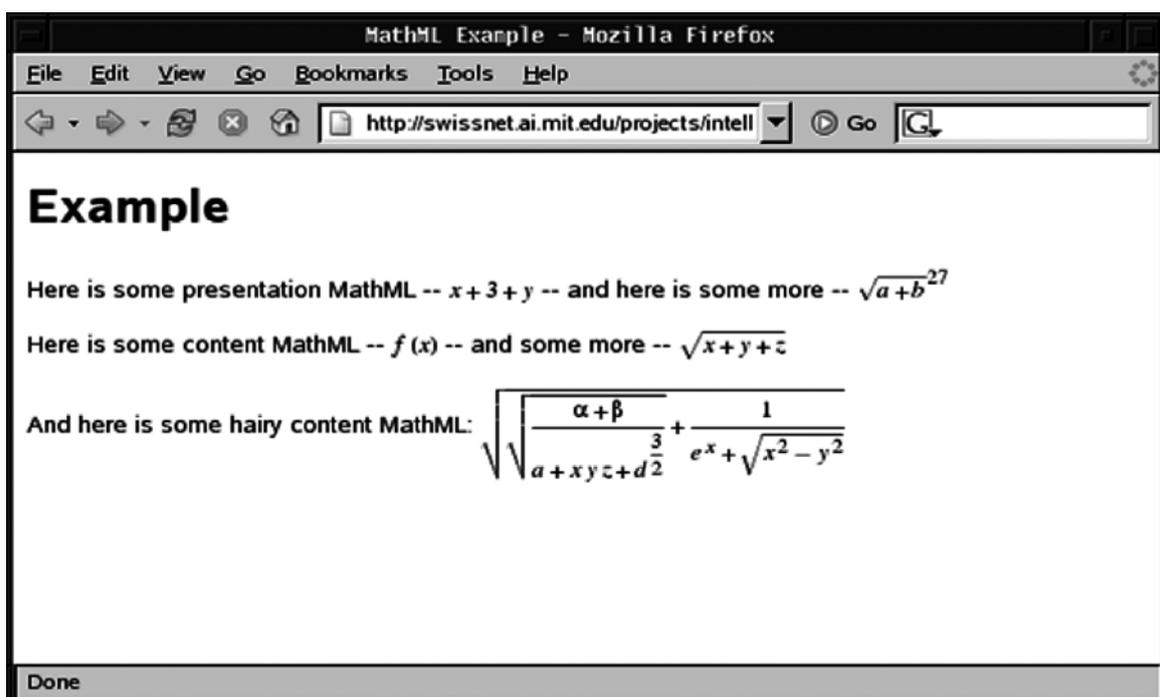
```
<mrow>
  <mrow>
   <msup> <mi>x</mi> <mn>2</mn> </msup>
   <mo>+</mo>
   <mrow> <mn>4</mn>
      <mo>&InvisibleTimes;</mo>
      <mi>x</mi>
   </mrow>
   <mo>+</mo>
   <mn>4</mn>
  </mrow>
  <mo>=</mo>
  <mn>0</mn>
</mrow>
```

Using semantic tags, the code is:

```
<apply>
  <plus/>
  <apply>
   <power/>
   <ci>x</ci>
   <cn>2</cn>
  </apply>
  <apply>
   <times/>
   <cn>4</cn>
   <ci>x</ci>
  </apply>
  <cn>4</cn>
</apply>
```

Several packages already have the capability to output MathML. These include LaTeX to MathML converters and the Word equation editor, although in my experience none of these are 100% reliable at the moment.

In principle, semantic tags should make it easier to interpret an equation because they convey more information about the meaning of equations than the geometrical layout of an equation on the page.

Summary

Mathematics is not easy to convey using Braille or screen readers and although partial solutions exist, they still have limitations. Several groups are actively working on the problems of converting maths to Braille, and producing screen readers for maths.

References

1. Duxbury web site:
   http://www.duxburysystems.com/
   [Accessed 19 April 2007].

2. RNIB, Braille Mathematics Notation 1987. 1987, Peterborough: Royal National Institute for the Blind.

3. Dotsplus web site:
   http://dots.physics.orst.edu/dotsplus.html
   [Accessed 19 April 2007].

4. Nottingham Alternative Formats Service:
   http://www.nottingham.ac.uk/is/services/alternative-formats/
   [Accessed 19 April 2007].

5. LaTeX Braille style file:
   http://www.ctan.org/tex-archive/macros/latex/contrib/braille/braille.html
   [Accessed 19 April 2007].

6. MathML web site:
   http://www.w3.org/Math/
   [Accessed 19 April 2007].

7. MathML example web site:
   http://swiss.csail.mit.edu/projects/intelligent-book/mathml/mathml-example.xml
   [Accessed 19 April 2007]

Editor's Note:

Look out for the follow-up article, in the MSOR Connections Aug 2007 newsletter, by Michael Whapples, a blind student with practical experience of using assistive technology to support him in his studies.

The "Supporting Students with Disabilities series" is a new regular series that is published by MSOR Connections and co-ordinated by Emma Jane Wright (University of Nottingham) and Peter James Rowlett (Nottingham Trent University). If you have any potential articles for this series, please email the editors at: editors@mathstore.ac.uk