RESOURCE REVIEW

Accessible teaching with GNU TeXmacs

Jeroen Wouters, Department of Mathematics and Statistics, University of Reading, Reading, UK.
Email: j.wouters@reading.ac.uk

Abstract

In this article I give a brief overview of some of the challenges in creating accessible documents for STEM education, as well as why and how GNU TeXmacs can be used to address some of these.

Keywords: accessibility, HTML, software, WYSIWYG.

1. Introduction

Equal access to education is an ideal that is supported by many governments and organisations and is enshrined in international treaties (Right to Education Initiative, 2022). Discriminated groups and society in general stand to gain from giving more opportunities to contribute to a wider group of individuals. It is, therefore, not surprising that legislation is put in place to improve equal access to education (HM Government, 2021). Such regulations stipulate that new educational materials have to meet accessibility requirements or, if this is not yet the case, that they should do so in the near future. Such accessibility means, roughly speaking, that the way in which information is transferred should not be an obstacle to the processing of that information.

In practice, this means providing documents in formats and styles that are suitable for users with visual, aural, movement or cognitive disabilities. For example, users with visual disabilities may require a text to be read out to them, while readers with dyslexia can benefit from the use of a concise writing style and a specific layout of the text on the page.

Education in science, technology, engineering and mathematics poses particular challenges in this regard, due to the frequent use of equations, graphs and diagrams that are difficult to convey in a non-visual manner. In the case of plots and diagrams, an accessible format may be a tactile reproduction on embossed paper or an audible representation. For mathematical equations the format may be Nemeth Braille, or HTML with MathJax that can be read out by screen reader software (Cervone, 2012).

2. Screen readers and HTML

We will now focus on users with visual impairment. For these students consuming text can mean that it should be read out loud. Fortunately, this can now be done by computer algorithms called screen readers. For documents with little structure (with a linear flow of text with the occasional heading), these algorithms work quite well. However, the situation becomes more involved when mathematical notation, such as subscripts, superscripts, fractions and square roots, appears. For the screen reader to correctly pronounce such mathematical constructs, they need to be presented in a structured format, so that the software can understand what symbols belong in the numerator and what belongs in the denominator of a fraction, say.

To the author's knowledge, the main methods currently available to provide structured mathematical information to screen readers is either via a tagged PDF or via HTML with MathML. Unfortunately, the popular Chrome browser currently doesn't support MathML, with some progress recently being made by Igalia (https://mathml.igalia.com/). In the meantime, using MathJax to provide the right output for a variety of browsers and screen readers seems to be the best solution. MathJax also provides other accessibility features, such as the ability to magnify parts of the equation.

3. The problem with LaTeX and possible solutions

The formats in which to present information in an accessible way presents one side of the equation. The other side is how to author such content. Currently a lot of the material used in higher STEM education is produced in LaTeX and output as PDF files. Unfortunately, LaTeX discards any structural information in the source once the document is typeset. The resulting output is basically a set of disconnect characters on a page.

Given the large amount of material already written in LaTeX, a lot of effort has been invested in creating accessible work-flows for it, following different strategies. Once ready, LaTeX3 should be able to produce tagged PDFs (Mittelbach, 2020), but is at least several more years from completion and usable solutions are needed right now. Other methods to create tagged pdfs are complicated to use.

LaTeXML and TeX4ht seem to be the most well-developed methods to create HTML output from LaTeX. Although they work well in many situations, due to the complexity of the LaTeX input, they fail regularly, with error messages that can be hard to decipher. Compilation can also take longer than with standard LaTeX. This is obviously not an ideal solution, as it further complicates the already tedious edit-compile-inspect cycle typical to LaTeX.

Since the complexity of LaTeX hampers its convertibility (Poulain, 2014), another solution is moving to a simpler format other than LaTeX, but closely related to it. For example, PreTeXt and RMarkdown can include equations in LaTeX, but they lack the flexibility of LaTeX. Most importantly, these formats still require authoring in raw text with an edit-compile-inspect cycle. If we need to change our work-flows and switch to a different format, why not change to a method that not only makes use of the document structure to make information accessible to a wider audience, but also makes use of that structure to make the authoring experience more pleasant to the writer?

This is what GNU TeXmacs aims to achieve (van der Hoeven, 2020). It is a true WYSIWYG ("what you see is what you get") editor for technical documents that makes use of the structure of the document to facilitate editing.

The fact that TeXmacs internally uses a format that is quite similar to XML also means that conversion to HTML and MathML are reasonably straightforward, certainly when compared to the conversion from LaTeX.

It should be noted that TeXmacs can not only be used to author new mathematical documents. It also features a converter to import existing LaTeX documents into TeXmacs. Its conversion algorithm is less robust and feature-rich than that of LaTeXML, but it gives a good result on well written LaTeX documents. The result may need manual editing, but in the author's opinion, this is worth it, considering the efficiency gains that can be obtained once using TeXmacs.

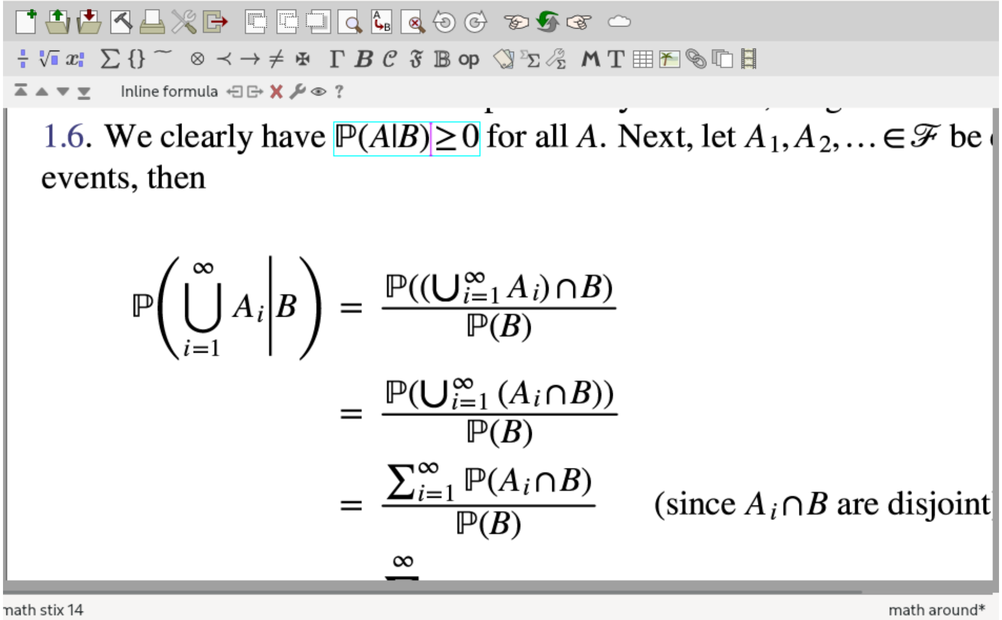

Figure 1. TeXmacs' user interface. Equations can be entered in three different ways: through the toolbars of the graphical user interface, using keyboard shortcuts (e.g. Alt-f for a fraction) or using LaTeX-like commands (e.g. "\frac" for a fraction).

4. Creating accessible documents in TeXmacs

Manuals and tutorials on creating TeXmacs documents can be found on the TeXmacs website. Here we'll only mention the conversion to HTML.

Practically speaking, to output HTML documents with MathJax equations in TeXmacs, go to the "Convert" tab of the "Edit → Preferences" menu. In the "Html" sub-tab, select "Export mathematical formulas as MathJax". Next, from the document you wish to export, select "File → Export → Html". The exported Html can be further styled using CSS, for example to select the font or page margins. A PDF file can similarly be saved via "File → Export → Pdf".

In the author's work-flow for educational documents, both a PDF and HTML version would be created for each item, as some students may still prefer the fixed PDF layout. The files are then attached to items in the University's virtual learning environment (Blackboard). By attaching them we have full control over the rendering of the HTML document, avoiding possible conflicts with Blackboard. Another possibility is to open the HTML file in a text editor and copy/paste the source code into a Blackboard item, although MathJax version 3 seems to be incompatible with Blackboard, so the version should be manually changed to 2 in this case.

From observation in the classroom, both the PDF and HTML formats are used by students, with HTML having the advantage of being better suited for the mobile devices that many students are now using.

TeXmacs can also be useful to create documents suitable for students with other disabilities than visual impairment. For example, the British Dyslexia Association provides guidelines on how to produce dyslexia friendly documents (British Dyslexia Association, 2018). Based on these guidelines, you may want to consider applying the following simple changes from the TeXmacs user interface:

- Go to "Document → Font" and pick a large and well readable font (e.g. Fira Sans or Carlito at 12-14pt size).

• Go to “Document → Paragraph” and choose an appropriately large interline spacing.

• Go to “Document → Colors → Background” and choose a light, but not white, background.

• If necessary, go to “Document → Page → Margins” and adjust the margins, so that about 60-70 characters fit on one line.

Once the above changes have been made, they can be put into a TeXmacs “style file”, which can later be added to other documents with two mouse clicks.

The above style changes will appear in the PDF version of the output, but are not automatically translated to the HTML output. For this, a CSS file can be used. The CSS used by the author is reproduced in the appendix. Users can also adjust the style of a document to their personal preference by using various browser plug-ins.

Inserting alt-text descriptions of images is currently not directly supported from the TeXmacs GUI, but they can be inserted by using the html-attr tag. To create this tag, type \html-attr and press the Return key. Edit the arguments to provide the alt-text: <html-attr|alt|"my alt text"|>. Then, move into the third argument field and press Return again. This will activate the tag. Finally, the image can be inserted into the active tag. The author has published a small plug-in for TeXmacs to simplify this procedure at https://gitlab.act.reading.ac.uk/ss902791/tm-alt-text.

5. Improving accessibility in online sessions

Another way in which the accessibility of STEM education can be improved is by using TeXmacs to deliver live on-line sessions. Previously, the author would use a tablet with a stylus to write equations on a virtual whiteboard. This, however, would occasionally lead to questions to clarify some of the handwritten equations. To increase legibility, it was decided to switch to TeXmacs for on-line teaching. Thanks to its intuitive keyboard short-cuts, the author can enter equations at nearly the same speed as he would write them on a whiteboard. To this end, the Beamer style included in TeXmacs can be used in combination with a large, clear font (e.g. Fira Sans 14pt) to make the equations and text easy to read.

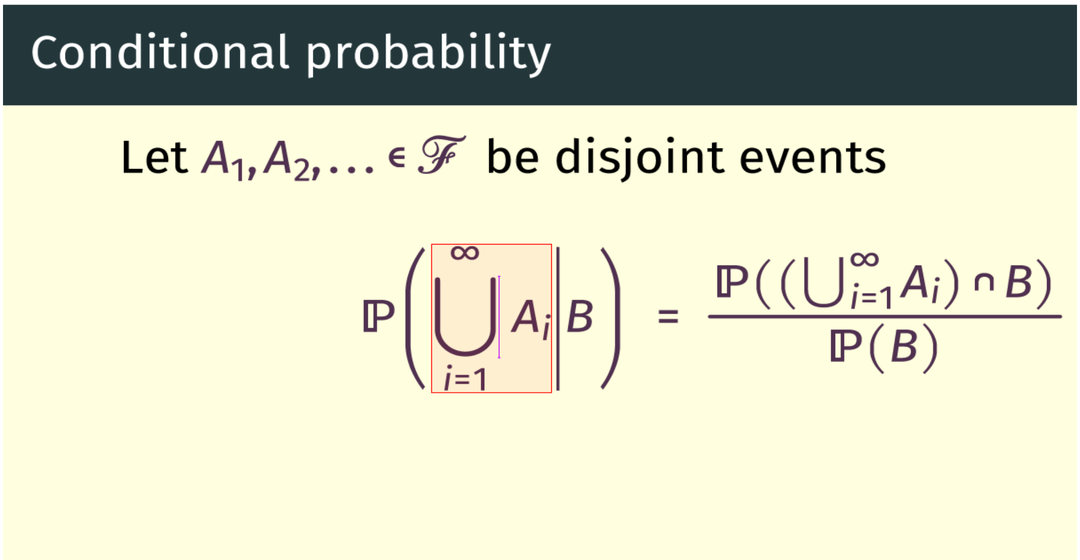

Figure 2. A screenshot showing the use of the Beamer during online teaching sessions.

To start a presentation in TeXmacs, from the user interface:

- Go to “Document → Style” and select “beamer” for TeXmacs' presentation style.
- Go to “Document → Font” and pick a large and well readable font.

After the first on-line session using TeXmacs as a presentation tool, students reacted positively when asked for feedback. A majority preferred this method over the method of drawing on an electronic whiteboard with a stylus.

Using TeXmacs during live sessions has the advantage of producing typeset output which can almost immediately be shared with students. The created content can be copied into a standard (non-beamer) document for some final editing before being converted to PDF and HTML as described above. Another advantage of this approach over other methods is that no additional hardware, such as high-resolution web-cams or styli, is needed. A disadvantage is that creating complicated diagrams requires some practice and is slower than drawing with a stylus.

6. References

Right to Education Initiative, 2022. International law. Available at: https://www.right-to-education.org/page/international-law [Accessed 14 February 2022].

HM Government, 2021. Making online public services accessible. Available at: https://accessibility.campaign.gov.uk/ [Accessed 28 October 2021].

Cervone, D., Krautzberger, P. and Sorge, V., 2016. New Accessibility Features in MathJax, *The Journal on Technology and Persons with Disabilities*, 4, pp. 167-175. Available at: http://hdl.handle.net/10211.3/180124 [Accessed 9 May 2022].

Mittelbach, F. and Rowley, C., 2020. LaTeX Tagged PDF—A blueprint for a large project, *TUGboat*, 41(3), pp.292-298. https://doi.org/10.47397/tb/41-3/tb129mitt-tagpdf

Poulain, F. and Van Der Hoeven, J., 2014. Conservative conversion between LaTeX and TeXmacs. Available at: https://hal.archives-ouvertes.fr/hal-00952926 [Accessed: 28 October 2021].

Van der Hoeven, J., 2020. The Jolly Writer. Your Guide to GNU TeXmacs. Scypress.

British Dyslexia Association, 2018. Dyslexia Style Guide 2018: Creating Dyslexia Friendly Content. Available at: https://www.bdadyslexia.org.uk/advice/employers/creating-a-dyslexia-friendly-workplace/dyslexia-friendly-style-guide [Accessed: 28 October 2021].

7. Appendix: dyslexia friendly CSS

```
html {
    background-color: #fdffe2;
    border: 0px;
    padding: 0px;
    margin: 0px;
}

body {
    font-size: 32px;
    font-family: 'Arial', 'Linux Biolinum', 'Belleza', 'Optima';
    text-align: justify;
    border: 0px;
    padding: 1em;
    margin: 0px;
    margin-left: auto;
    margin-right: auto;
    max-width: 45em;
    line-height: 2.0
}

/*  Add a scroll bar to long equations */
mjx-container {
  display: inline-grid;
  overflow-x: auto;
  overflow-y: hidden;
  max-width: 100%;
}

/* Some equations are put in tables, so add a scroll bar */
table {
    display: block;
    overflow-x: auto;
    white-space: nowrap;
}
```