

OPINION

Accessible equations

David J. Smith, School of Mathematics, University of Birmingham, Edgbaston, Birmingham, B15 2TT, UK. Email: d.j.smith@bham.ac.uk

This opinion piece is modified from a blog post at The Big Conversation, University of Birmingham (<https://blog.bham.ac.uk/bigconversation/2018/03/19/accessible-equations-by-dave-smith-school-of-mathematics/>).

There is a divide in the academic world between those for whom equations and mathematical expressions are their primary tools, and everyone else! Mathematicians, statisticians, operational researchers, computer scientists, physicists and many computational biologists (for the purposes of this opinion piece I will refer to all of us as *mathematicians* for brevity) typically find that widely-used tools such as Microsoft Word and PowerPoint are unsatisfying for the preparation of mathematics. Their main drawbacks are the combination of a clunky point-and-click interface with, to the experienced eye, substandard aesthetics. The most popular alternative system is the typesetting language LaTeX which was mainly developed in the 1970s and 80s. LaTeX enables the production of professional-quality printed documents with equations that most mathematicians consider rather beautiful; it is no surprise therefore that the vast majority of us produce all of our lecture notes and other course materials (handouts, exercises, slides and even examinations) in LaTeX. Typically the implementation that is used is pdfLaTeX, which converts LaTeX ‘source code’ into a PDF file that can be viewed electronically or printed. LaTeX has undoubtedly revolutionised mathematical publishing, and has also influenced the development of HTML (hypertext markup language), the basis for the web.

One of the great appeals of pdfLaTeX is that it gives us complete control over visual layout. However, therein lies a weakness – not everyone consumes lecture material visually, and not all who do see in the same way. The most powerful example of this diversity is the use of screenreaders by blind and visually-impaired students. Equations appearing in PDF files produced by LaTeX are completely unintelligible to a screenreader. A workaround for this difficulty, used with success in my own institution, involves providing the LaTeX source files so that the code itself can be interpreted – ‘backslashes, curly brackets and all’ – by a screenreader. This is certainly workable, but wouldn’t it be preferable if the core set of notes were suitable to be adapted by students to their varying needs? Other examples of how materials may need to be ‘consumed’ differently include the use of large print by students with visual impairments, or the use of sans-serif fonts and coloured backgrounds by students with dyslexia. Special materials can be printed out on request, but wouldn’t it be better if students could simply enlarge text, or experiment with changing the font or background to see what works best for them? The issue of how we consume reading material is most acutely relevant to those with visual impairment, however across all of the academic community we now view content across a range of devices, from monitors to laptops to tablets and phones, all of which require text to be able to resize and reflow according to the dimensions of the screen. We wouldn’t expect online newspapers to come in the form of a downloadable PDF; it seems reasonable that students should have similar expectations of course materials.

HTML-based materials by contrast provide an excellent and up-to-date way to deliver device-friendly, resizable and reformattable content, along with the other advantages of web-based materials, particularly hyperlinking. Virtual Learning Environments such as *Canvas* (used at the University of Birmingham) and equivalent systems can provide a user-friendly platform for colleagues unfamiliar with HTML code to prepare webpages – but what about mathematics? The simplest solution I am

aware of involves integrating the old and the new: LaTeX expressions – which can be included by enclosing within the symbols $\backslash ($ and $\backslash)$ – within an HTML web page. LaTeX expressions can then be interpreted into mathematics through the online service MathJax*, a JavaScript engine developed by the American Mathematical Society and Society for Industrial and Applied Mathematics. MathJax can be included in a webpage by adding a code snippet to the start of the HTML file; in Canvas this is done automatically so one does not even need to write a line of HTML. The result is mathematical expressions that look beautiful but more importantly are then available as MathML for the use of screenreaders. MathJax *accessibility extensions* further enable improved visual and aural rendering by detecting mathematical structure, allowing features such as collapsing and exploring of subexpressions. Having prepared materials in this way one no longer needs to anticipate all of the possible needs of current or future students – the power is with the student to manipulate the content appropriate to their needs. An example Canvas-LaTeX page is available to view†. For the lecturer, converting a set of LaTeX notes to ‘Canvas-LaTeX’ or ‘HTML-LaTeX’ is essentially a task of cutting-and-pasting and then modifying commands such as section headings and figures – something that can be accomplished in a few hours for a 40 lecture course.

I initially posted the comments above to the Big Conversation blog at the University of Birmingham and Twitter, resulting in several replies. One commenter with dyspraxia said that they found that printed notes were much easier to organise and interact with, and asked whether HTML notes would replace pdfLaTeX. HTML notes can also be printed, and indeed I typically print and distribute my online notes, with a few gaps added intentionally, to help with student engagement during lectures. However, it is certainly clear that many would like the option of outputting a pdfLaTeX document too. Along these lines, another commenter highlighted the Python-based documentation package Sphinx, which can output both LaTeX and HTML from a format called restructuredText. It was also pointed out that LaTeX itself should be regarded as an *input system* rather than an output format as such. The package tex4ht can convert .dvi files produced by LaTeX into HTML, although the mathematics output is poor; a much better solution developed by Andrew Stacey‡ introduces a new {internet} document class for LaTeX from which a range of formats can be generated. For those who find LaTeX both powerful and comfortable to use – probably most of the mathematical community – this approach may be very appealing. Further options – highlighted by an anonymous reviewer of the present article – are Pandoc§, which can convert between many formats, including from LaTeX to HTML with MathJax and MathML, the Python package plasTeX** and its associated TeX-to-Braille converter BrITex†† (for more on BrITex, see Whapples, 2007; Rowlett et al., 2010). My current preference is using MathJax $\backslash (\dots \backslash)$ expressions within HTML due to the fact that no compilation or special software is required. There will no doubt be a range of approaches and mathematicians will find the one which best fits their style, workflow, and comfort level with IT.

* <https://www.mathjax.org/>

† <https://canvas.bham.ac.uk/courses/30102/pages/introduction>

‡ <https://github.com/loopspace/latex-to-internet>

§ <https://pandoc.org/>

** <http://tiarno.github.io/plastex/>

†† <http://brltex.sourceforge.net/>

In summary there is a great opportunity to bring our teaching materials into the 21st Century, retaining many of the advantages of ‘traditional’ pdfLaTeX along with much greater accessibility. This change requires a shift in how we view course materials – precise control over visual appearance to suit the lecturer’s aesthetic sensibilities is of low priority to blind, visually-impaired or dyslexic students. Our primary job as educators is to provide *content* (text and equations) the format of which students are then empowered to adapt best to suit their needs.

References

Whapples, M., 2007. Obtaining Braille mathematical documents. *MSOR Connections*, 7(3), pp. 18-21.

Rowlett, P., Rowlett, E., Cliffe, E., Edwards, A., Whapples, M. and Pfluegel, E., 2010. Visual impairment in maths, stats and operational research (MSOR). *MSOR Connections*, 10(2), pp. 45-48.