

Ideas from Oz, for New and Retro-Born Digital Mathematics

Ross Moore

Mathematics Department, Macquarie University, Sydney, Australia

E-mail: ross@maths.mq.edu.au

URL: <http://www.maths.mq.edu.au/staff/ross.html>

Abstract. We look at various techniques to improve the presentation and usability of mathematics in electronic documents, such as abstracts on websites and articles in PDF format. Specific examples of these techniques are available, via the internet, at the website of the Australian Mathematical Society. The author is the web editor for this society.

For mathematics in HTML pages, proper typesetting is possible using Davide Cervone's *jsMath* software. This avoids the use of low-resolution images, and extra fonts can be installed to achieve faster page-building, as well as providing the highest quality printing. Abstract pages can include direct hyperlinks to *MathSciNet*, for all the references of an article that have been reviewed, or a review is pending.

With PDF documents, newly developed features include:

- a. copy/paste of mathematics from the PDF, resulting in either correct symbols in Unicode/UTF-8 encoding or the \TeX source itself;
 - b. moveable pop-up images for figures and tables;
 - c. bookmarks to definitions, theorems, and those equations which are cross-referenced within the article itself;
 - d. display of mathematical symbols within bookmarks, using Unicode characters;
 - e. metadata attachments that accompany the PDF, and are generated by the same typesetting run as the document itself;
 - f. hyperlinks to *MathSciNet* for each bibliography entry, where available.
- All of these features can be included with new documents, or (if done carefully) with retro-born re-typesetting of articles, without any obvious effect on existing pagination.

Key words: math-extraction, PDF techniques, metadata attachments, jsMath, Unicode strings

1 Introduction

The nature of scientific publication is changing and it is becoming increasingly common for articles to be accessed and read on-line, without the need for printing. However, many researchers still prefer to print out an article, having first obtained it as a PDF file, say. Thus it is necessary to produce the PDFs in such a way as to cater for both online and printed formats. For online reading, one needs navigation aids that give quick and easy access to cross-references, citations, metadata, and such-like. However, with retro-born versions of books

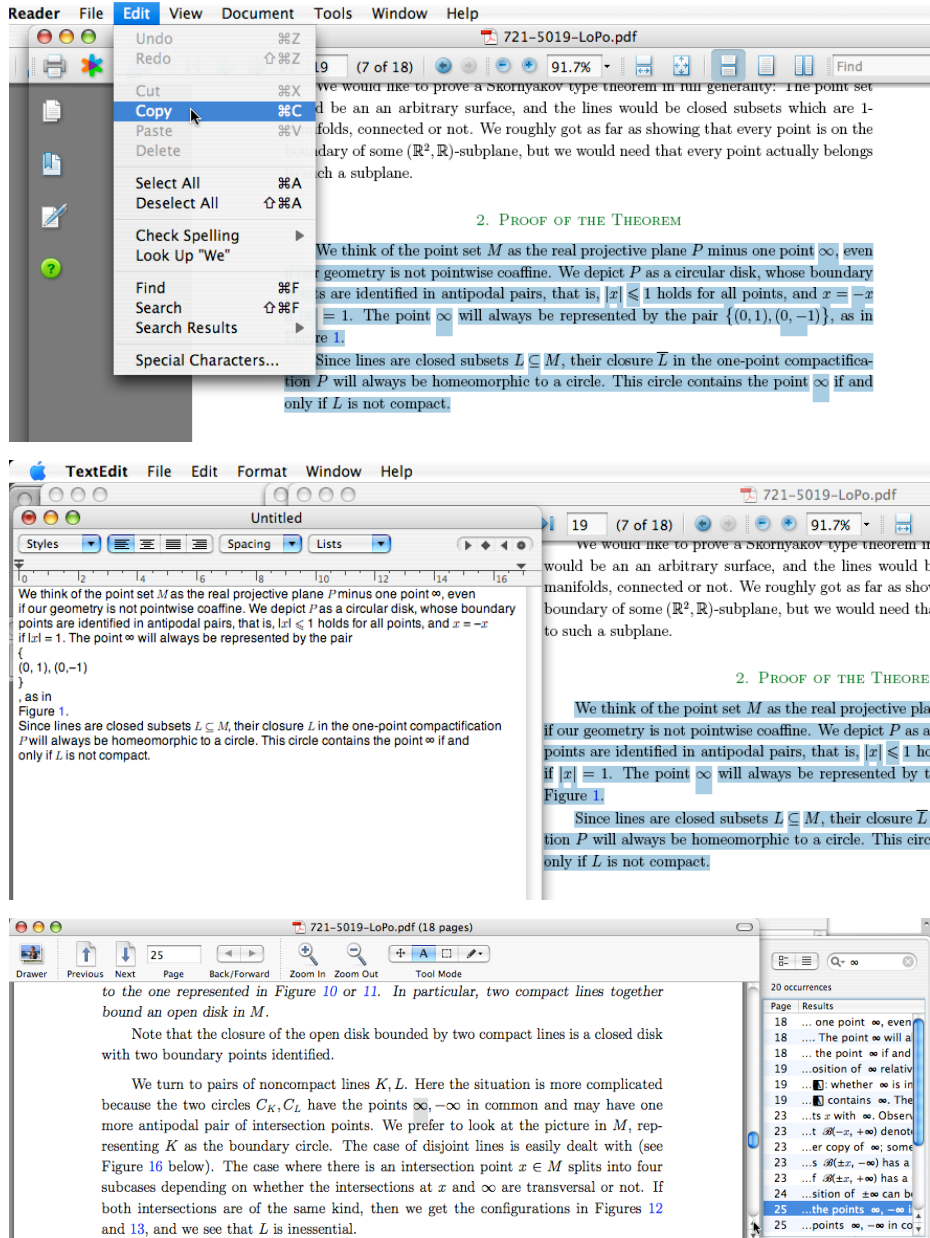


Fig. 1. Copy/Paste of mathematical content: the middle image shows the result of pasting the contents that have been selected and copied, as shown in the upper image. The actual result of a copy/paste action may depend on the particular software being used; that is, the PDF-browser used to view the article. The lower image shows how searching for mathematical symbols is done, with suitably enabled PDF viewers.

and journal articles, the addition of such aids must not have an effect on the original pagination. Furthermore, for long-term digital archiving, accurate metadata and links to access cited materials become especially important.

There are effects that are possible with current web-based and PDF technologies, but which hitherto have not been widely used with scientific articles. The effects were programmed for processing with PdfTeX, along with extra packages and coding to adjust the output produced by L^AT_EX internal macros. Almost no further adjustments were made to the body of files used to produce the original printed version of each paper, apart from the imposition of `\label` and `\ref` commands where there has had not formerly been used with cross-references.

In this paper such effects are presented mostly in a graphical way, as figures with extensive captions. A limited amount of extra text explains aspects that are not immediately apparent from seeing the images alone, along with a short outline of how these were implemented in L^AT_EX, in terms of which (internal) macros needed to be patched, and what extra resources were used.

The HTML examples use Davide Cervone's *jsMath* software,¹ which can be linked to any website to provide proper typesetting of mathematics, as well as high-quality printing. This solves many of the problems that accompany other methods of displaying mathematics within webpages.

2 PDF Files of Full Articles

The author has produced retro-born versions of all articles from a complete journal volume² as part of a feasibility trial to move from print-only to online access. Bearing in mind the considerations mentioned above, and with pdfTeX and Unicode support becoming more widely available, much care was taken to incorporate navigational aids that are available with PDF documents. Several new techniques were developed to make these documents as useful as possible to researchers. These are now described briefly.

a. *Copy and Paste of blocks of text which include mathematical symbols (see Figure 1).* This is achieved by adding a character map (CMap) resource³ to each of the fonts that T_EX uses, for typesetting the text and the mathematics. Such resources do not affect the appearance of the typeset material, but associate each (perhaps accented) letter and mathematical symbol to their Unicode code-point. In this way symbols are given a unique identity which can be used for copy/paste to other applications, and for searching within the PDF itself. Currently the actual result of a copy/paste action may depend on the particular software being used; that is, the PDF-browser used to view the article, and the text-editor, or other software, into which the content is pasted.

¹ *jsMath* homepage: <http://www.math.union.edu/~dpvc/jsMath/>.

² Bulletin of the Australian Mathematical Society, Volume 72 (2005); freely available online at <http://www.austms.org.au/Bulletin>.

³ see Adobe CMap and CID specifications at http://www.adobe.com/devnet/font/pdfs/5014.CIDFont_Spec.pdf.

The author has produced CMap resources for the following old-style \TeX font encodings: OT1, OML, OMS, OMX. Articles in the journal volumes¹ also used symbols from the AMS fonts MSAM and MSBM, and a few other characters, so CMap resources have been made for their ‘U’ (Unknown) encodings; namely files umsa.cmap, umsb.cmap, ulasy.cmap, and upzd.cmap (Zapf Dingbats) and upsy.cmap (Adobe Symbol). Resources have also been created for LY1 (Lucida) and LMR (Lucida Bright Math symbols) encodings.

In the case of OML encoding, as used with the cmmi math-italic font family, the ordinary letters A, B, . . . , a, b, . . . , z are associated with “math alphanumeric symbols” in Unicode Plane 1. Such symbols can be seen as the M, P, x and L within the **TextEdit** window in the middle image of Figure 1. However, bold symbols from cmmib fonts use the same OML encoding. Work is ongoing to construct CMap resources that are specific to the font face and style, rather than just to the encoding. Preliminary tests for omlmmi.cmap and omlmmib.cmap, supporting the cmmi and cmmib font families respectively. A little more work is needed to extend the method to also support sans-serif and typewriter math-alphanumeric styles. Similarly support for the Euler fonts is planned, and other symbol fonts also can be supported.

- b.** *Images of figures and tables which pop-up (see Figure 2) near the place in the text where the figure/table has been referenced.*

This feature allows figures and tables to be viewed without changing the PDF page that is displayed. It requires JavaScript⁴ (or ECMAScript) to be enabled within the PDF browser. If the toggled image pops-up in a place that is inconvenient for further reading, then it can be shifted to elsewhere on the page. With further developments of the PDF specifications and browser software, the means to move the image could be redesigned to become more intuitive. (Indeed it would be nice if browsers had a ‘cross-reference spy-glass’ feature, providing a small-sized view of a different part of the same PDF, in response to clicks on the cross-reference anchors.)

In case JavaScript is disabled in the PDF browser, so that the pop-up mechanism cannot work, the toggle button should not appear and the underlying cross-reference hyperlink should work as usual. Unfortunately, not all PDF browsers implement this properly.⁵ Indeed Xpdf and eVince, prior to the v0.8.3 release (on 4 June 2008) of the Poppler library, would not even load documents built with pdf \TeX containing form fields; more recent versions now work as intended.

Implementing this popup feature was done by altering the expansions of \LaTeX macros `\figure` and `\table`, changing the way that float contents are handled, and of `\ref` for building the toggle buttons. The first run of the \LaTeX job is largely unchanged, apart from recording the number of floats encountered as the expansion of a macro `\hasfloats` within the .aux file, followed by a macro

⁴ see <http://en.wikipedia.org/wiki/JavaScript>.

⁵ For example, Apple’s Preview browser, at least up to version 3.0.9, neither supports JavaScript, nor respects the button flags. Hence not only do pop-up images fail to work, but also the underlying hyperlink cannot be accessed.

Each of the two cases splits into subcases depending on the position of ∞ relative to \bar{L} : whether ∞ is in \bar{L} or not in \bar{L} , and in the latter case, which of the complementary components of \bar{L} contains ∞ . The resulting five possibilities are depicted in Figures 2 through 6.

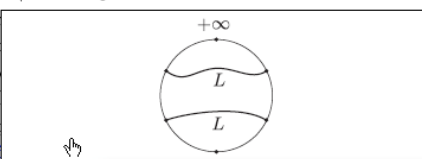
The figures also show the connected components of the complement of a line relative to M . Essential lines do not separate the point set M ; their complement is a punctured disk or an intact disk depending on whether the line is compact or not. An inessential

the pair formed by the real projective plane and one of its lines. In this case, the line L will be called *essential*. The complement $P \setminus \bar{L}$ is an open disk.

A similar procedure (needing extra corrections) may be applied in the case of two disjoint circles. In this case, the line L will be called *inessential*. The complement of \bar{L} in P is a disjoint union of an open disk and an open Moebius strip. In this case, essential lines do not occur (Proposition 5).

Each of the two cases splits into subcases depending on the position of ∞ relative to \bar{L} : whether ∞ is in \bar{L} or not in \bar{L} , and in the latter case, which of the complementary components of \bar{L} contains ∞ . The resulting five possibilities are depicted in Figures 2 through 6.

The figures also show the connected components of the complement of a line relative to M . Essential lines do not separate the point set M ; their complement is a punctured



Shift-click image then move mouse to shift image; click again (no Shift) anchors at destination. (requires Adobe Reader 7.0 or later).

Figure 6: A

map, or $\pi^{-1}(L)$ is a single circle $C = -C$, and π restricted to C is the unique two-fold

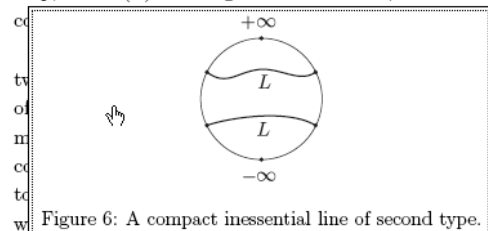


Figure 6: A compact inessential line of second type.

A similar procedure (needing extra corrections on the annulus defined by C and $-C$) may be applied in the case of two disjoint circles and results in a proof that (P, \bar{L}) then is homeomorphic to the pair formed by the real projective plane with a conic. In this case, the line L will be called *inessential*. The complement of \bar{L} in P is a disjoint union of an open disk and an open Moebius strip in this case. Our first aim is to show that inessential lines do not occur (Proposition 5).

Each of the two cases splits into subcases depending on the position of ∞ relative to \bar{L} : whether ∞ is in \bar{L} or not in \bar{L} , and in the latter case, which of the complementary components of \bar{L} contains ∞ . The resulting five possibilities are depicted in Figures 2 through 6.

The figures also show the connected components of the complement of a line relative

Fig. 2. Pop-up figures & tables: the images show how when the mouse hovers over a reference to a figure, it highlights a button to toggle showing and hiding a floating image. This image is moveable in case its natural position is inconvenient; e.g., obscuring where you wish to read.

`\testforfloats`. On the next \LaTeX run, as the `.aux` file is read `\testforfloats` is encountered, triggering code that reads the number of floats and causes extra packages to be loaded, such as `pdftricks.sty`, `insdljs.sty` for inserting JavaScript coding into a PDF, and `pdfpopup.sty` which has coding for building buttons to show/hide the pop-up images.

Now when a float is encountered, the full contents of the environment is written out into a file, named according to the figure or table number, as part of a new \LaTeX job which includes a preamble loading most of the same packages as in the main job. This subsidiary job is run, using `dvips/ps2eps/epstopdf` to get a single-page PDF with correct bounding box. The file is loaded back into the main job using `\pdfrefximage` within a `\setbox`, and its PDF object reference number is recorded for later use in placing the image on a JavaScript button field, when an appropriate `\ref` occurs. The object reference is also written into the `.aux` file as the expansion of a macro having name derived from the figure number. This allows the reference to be used on the next run, with any `\ref` command that occurs before the float has been encountered, and also allows checking to see whether the reference number has changed, in which case a message is written to the `.log` file warning that another \LaTeX run is required. Finally, the tokens for the float environment are recovered and processed normally.

On subsequent \LaTeX runs, the image files do not need to be rebuilt, but are loaded from the PDF images created on an earlier run. If editing of the main document source changes the order of floats, the toggle buttons may become associated with the wrong images. Simply delete those images from the current directory; the correct ones should be generated afresh on the next \LaTeX run. In case the content of a float may contain `\ref` and `\eqref` commands or citations, the subsidiary job that creates an image also loads a copy of the `.aux` file from the main job, which copy was made at the end of the previous run. It's possible that cross-references have not fully stabilised, so simply delete any affected image; after two more runs it will have been regenerated and included.

If browser software had a 'cross-reference spy-glass' feature (as suggested in the first paragraph of this item), then not only would there be no need for the PDF to contain JavaScript coding for the extra buttons, but also there would not be doubling-up of the information contained in figure and table floats. Furthermore, such a popup-like feature would 'just work' also for cross-references to section headings, numbered equations, etc., as well as to the floats, and perhaps also for citations and 'back-references' from the bibliography (see Figure 7, for example). This is surely the way that such a feature ought to be implemented; ideally it should not be necessary for a scientific document to include explicit programming which controls how its content be displayed, but just have declarations of which browser-supplied functions are to be used. The implementation presented here is mainly to demonstrate the usefulness and practicality of such a 'pop-up' feature for cross-referenced material, so that browser vendors might be encouraged to incorporate a similar feature within their own publicly-available software. However, there are certainly other,

simpler uses for pop-ups to show extra images that are not found elsewhere among the usual pages of a document.

c. *Extended use of bookmarks (see Figure 3), ...*

Use of bookmarks is quite common for the major sections of a document; this is automatic when using `\usepackage{hyperref}` with a \LaTeX document. This is here extended further to creating bookmarks for definitions, Theorems, Lemmas, etc. and also for figures, tables, and some equation displays. To avoid the bookmark window becoming too cluttered, only those equations that have actually been cross-referenced within the document are given their own bookmark.⁶

Having such bookmarks means that there are named destinations with the PDF at all the important places for the structure and content of the document. Furthermore these names are available in a separate file, so potentially this can be used to construct hyperlinks directly to these important places. This could be extremely useful in the context of a digital archive.

For figures and tables, a meaningful string to be the textual anchor in the 'Outline' window is obtained as the first sentence in the caption. This is obtained by reading the caption from the .lof or .lot file and parsing to locate the first full stop ('.'). With Theorems, Lemmas, Propositions, etc. the anchor-text uses the appropriate numbering, as seen in Figure 3. The limiting of bookmarks to only those referenced is achieved by patching the `\@setref` internal macro to implement a 'memory' of referenced labels. A line is written into the .aux file; this defines a macro, with name derived from the label. The coding for placing equation numbers is patched so that `\df@tag` now also places an anchor, and a bookmark for this anchor when the memory indicates that the equation has been referenced—which is known on the 2nd and subsequent \LaTeX runs.

d. *... including support for mathematical symbols and some super-/subscripts.*

The identifying strings displayed within the bookmarks window allow for the full range of Unicode characters (more correctly, code-points). This means that mathematical symbols can appear, which is most appropriate for figure captions, and mathematics is used with (sub-)section titles and theorem names, say. Even though the Unicode specification does not allow for complete alphabets of super-scripted and sub-scripted letters, an attempt is made to make good use of those that are available. Some further work is needed on this aspect of bookmarks.

e. *Inclusion of attachments (see Figure 4), containing metadata for the article, in various useful formats.*

The metadata for a scientific paper is important for various purposes, not least of which is for citation within future works. By distributing a document with its own metadata, common difficulties can be avoided, such as the incorrect spelling of an author's name, or wrong affiliation, etc., or just getting the page

⁶ It can be argued that if an equation is not referenced then it doesn't need an equation number. However, many articles have been written where the author has not followed this maxim. For creation of bookmarks, this maxim has been programmed-in.

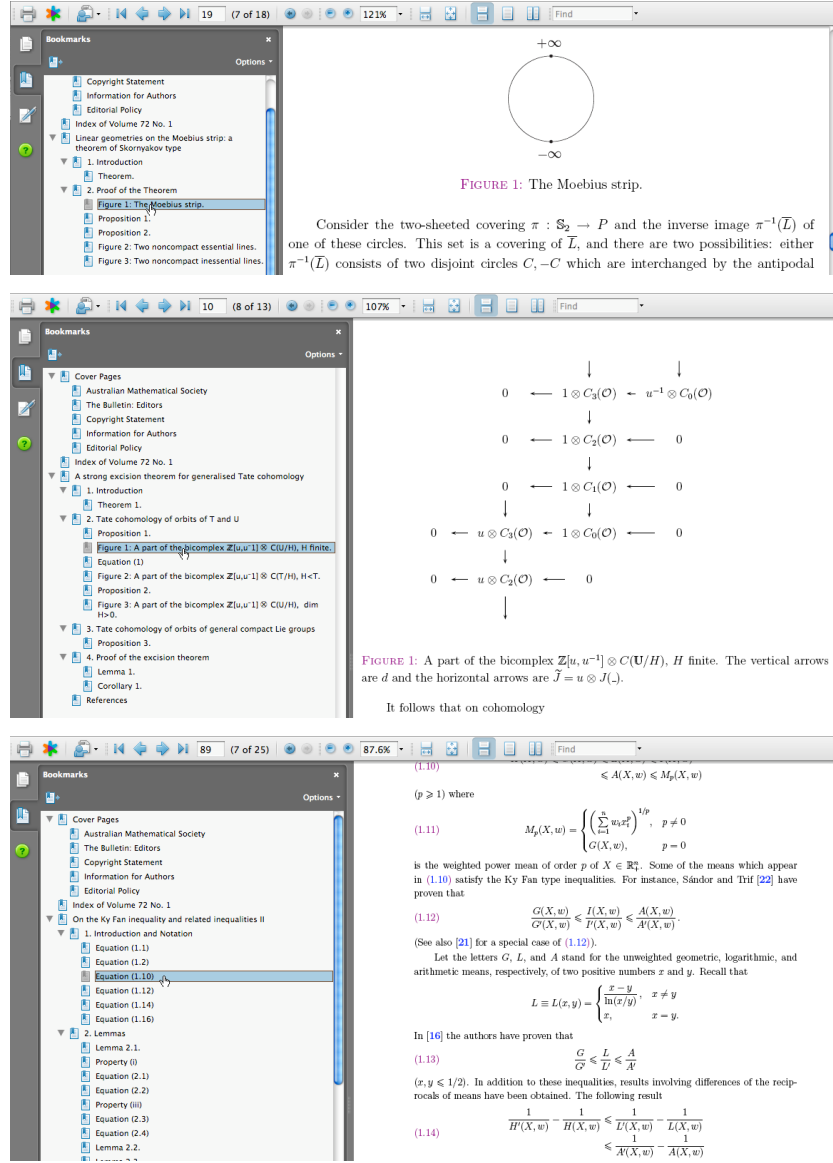


Fig. 3. Bookmarks: these three images, taken from different articles, illustrate various aspects of the automatically created ‘bookmarks’. These provide easy access to the important parts of the document, including front-matter as well as sections of the mathematical article itself; such as definitions, theorems, proofs, remarks and figures. The middle image shows that mathematical symbols can be used within bookmarks, while from the lower one it can be seen that a bookmark is not produced for every numbered equation, but only for those that are cross-referenced within the article itself.

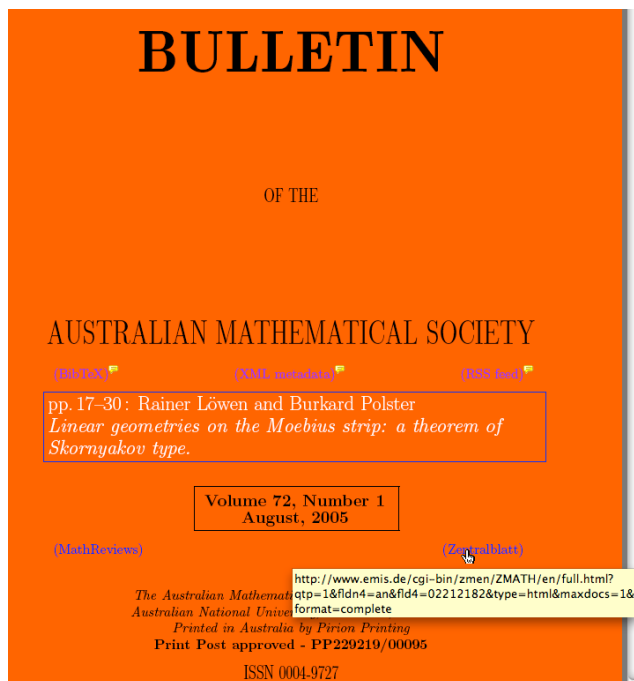


Fig. 4. Coverpage: this includes hyperlinks to recover the embedded metadata attachments, in various formats. Also there are links to the websites of the MR and *Zentralblatt MATH* reviewing agencies, where this article has been reviewed. The large button enclosing the title links directly to the start of the article proper.

Bulletin of the Australian Mathematical Society		
A note on the lattice of density preserving maps		
Sejal Shah and T.K. Das	1
A strong excision theorem for generalised Tate cohomology		
N. Mramor Kosta	7
Linear geometries on the Moebius strip: a theorem of Skorniyakov type		
Rainer Löwen and Burkard Polster	17
Div-curl type theorems on Lipschitz domains		
Zengjian Lou	21
A nonlinear map for midpoint locally uniformly rotund renorming		
S. Lajara and A.J. Pallarés	25
A remarkable continued fraction		
David Angell and Michael D. Hirschhorn	45
A new variational method for the $p(x)$ -Laplacian equation		
Marek Galewski	53

<http://www.austms.org.au/Publ/Bulletin/V72P1/index.html#31>

Fig. 5. Index page: containing links to the Australian Mathematical Society's website, for accessing each of the articles from the same journal issue.

numbers wrong. These PDFs come with text-file attachments containing various amounts of metadata in useful formats: (i) a Bib \TeX entry suitable for adding to a .bib file; (ii) a short HTML-formatted entry, suitable for an RSS feed; (iii) an extensive metadata file in XML format, which includes the complete bibliography of the article, as well as its own publishing details.

Each of these metadata files is generated “on-the-fly” by suitable \TeX macro coding, using the information provided for typesetting the PDF document itself. Thus, barring mistakes in the various format translations required for this, the metadata is guaranteed to be consistent with what appears in print.⁷ Access to these files is made easy by anchors on the cover-page (see Figure 4) located above the white text of the article title and authors, which itself anchors a hyperlink to the start of the article proper.

When available, database codes for reviews of the article are included with the Bib \TeX and XML metadata files. In this case, the cover-page has further anchors, for hyperlinks which give direct access to the review, at each of Math Reviews⁸, and *Zentralblatt MATH*⁹. The same typesetting run that produces the PDF also builds an HTML webpage for the article¹⁰. This presents all the metadata and has the same hyperlinks, and more. Some format translations are also required when building such webpages.

Part of the metadata for an article is the context in which it has been published; namely, the complete journal issue. For this, we have chosen to include the complete front-matter that would appear in the printed version of the journal. The images of Bookmarks in Figure 3 show the kind of material that is included: Editors, Copyright Statement, Information for Authors, etc. Of course the Index page (see Figure 5) lists all other articles appearing within the same issue. For each article there is an active hyperlink, using the page-number as the visible anchor, that directs a web-browser to the public page at the Australian Mathematical Society’s website where the article’s abstract can be read, and its metadata (including references) examined. Also, the name of the article itself is the anchor for another hyperlink to the start of the article proper.

f. Hyperlinking from the bibliography (see Figure 7) to the place within the text (back-referencing) where the citation occurred, and to reviews at MathSciNet.

Including back-references is not new, nor is having hyperlinks within the bibliography, when such are supplied by the article’s author. However, for articles where the original printed version did not have these features, there is the problem of how to include the extra information without upsetting the pagination. Figure 7 shows an elegant solution that places the hyperlink anchors discreetly into the left margin. The caret-accent character has been chosen to be

⁷ This is a useful feature for freshly published articles, but is not really appropriate for a digital library or preprint archive, which would presumably have its own database of metadata already prepared and checked for the documents that it serves.

⁸ A subscription to *MathSciNet* is required to make use of this hyperlink.

⁹ Unregistered users have reduced access to the features available at this site.

¹⁰ <http://www.austms.org.au/Publ/Bulletin/V72P1/721-5019-LoPo/index.shtml>

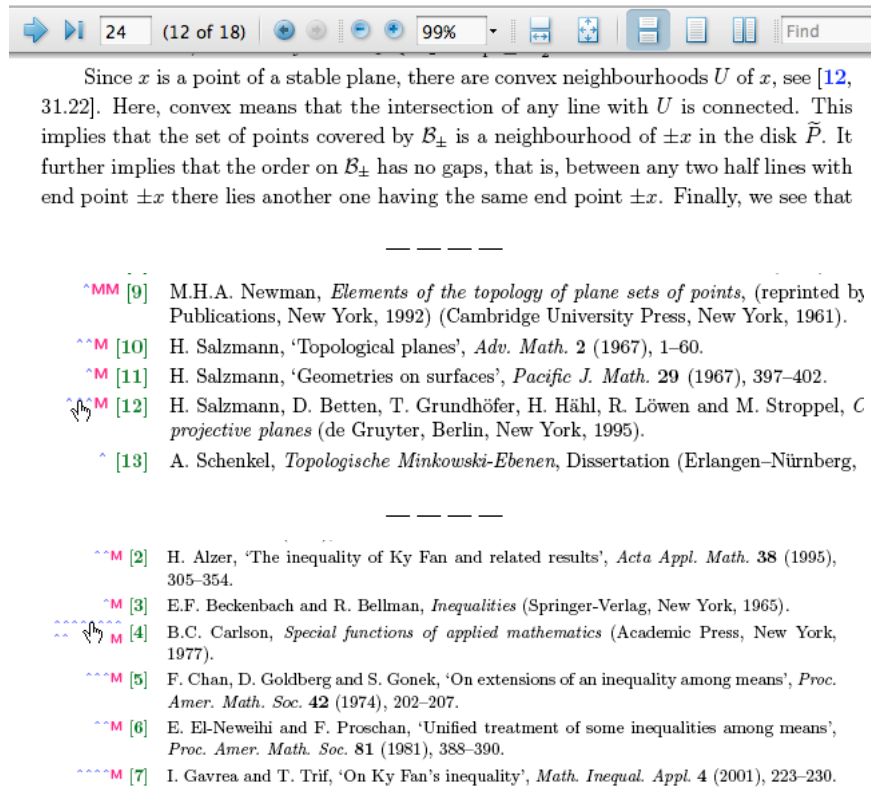


Fig. 6. Back-reference hyperlinks: the middle image indicates how the caret (^) in the left-margin is the anchor for an active hyperlink, which jumps to the location where the particular reference has been cited. The upper image shows the resulting change of focus. As there can be several citations of the same item, the carets are right-aligned, with up to eight in a row, as in the lower image.

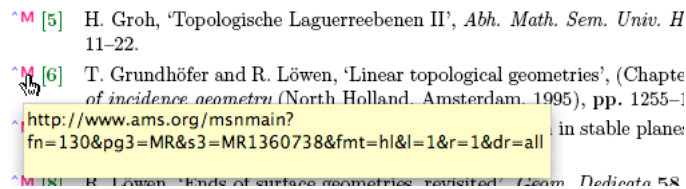


Fig. 7. Hyperlinks to *MathSciNet*: the small raised **M** is the anchor for a hyperlink that connects to the American Mathematical Society's website. It links to the review of this cited bibliography item. It is possible to have multiple such links, as can be seen in Figure 6. (See entry [9] in the middle image.)

suggestive of an upward link; that is, to the material preceding the bibliography, which is the main body content.

The second problem, which is perhaps the more difficult one, is that of how to automate the collection of data required to build hyperlinks to reviewing services, such as *MathSciNet* and *Zentralblatt MATH*, or other online archives of scientific material. For these PDFs the author made use of a small program called *bmref*, which is essentially a ‘batched’ version of *MathSciNet*’s *Mref* tool¹¹. Provided by Patrick Ion (Associate Editor of Mathematical Reviews, AMS), this short Perl program sends a carefully constructed XML file as a query to the *MathSciNet* database, as an HTTP ‘POST’ to <http://www.ams.org/batchmref>. The result returned is an XML file containing the same data, but with extra fields added for (i) the number of matches found, (ii) the reference numbers (MR-number) of found matches, and (iii) full bibliographic information, for each bibliographic item included in the original submission.

This allows the MR-numbers to be obtained for all items in the bibliography (well, all those that have been reviewed), with a single submission. Each MR-number is sufficient to build the desired hyperlink. Of course this is not 100% reliable, and some searching at *MathSciNet* can uncover MR-numbers for items that were not found in the automated search; but the bulk¹² of the job is done automatically. \TeX coding was developed to analyse the author-supplied bibliography and construct the XML file for use with *bmref*. Taking advantage of \TeX ’s `\write18` feature to run external commands and await the reply, the whole process can be fully automated and integrated with the typesetting run: craft the XML file, send to *MathSciNet*, analyse the result, extract the MR-numbers, then make these available for creation of hyperlinks. After a successful result of one such run, there is no need for the same tasks to be repeated on subsequent typesetting runs.

The bibliographic information returned can be requested to be in any of the usual formats: *bibtex*, *amsrefs*, *TeX*, *html* or as an *HTML* hyperlink. Thus this information could be used to check the bibliographic details provided by the author, or could even replace it altogether. This was not done with these tests, due to the desire to have the online content be the same as what was printed; however, if any factual errors were noticed (such as incorrect Volume or page numbers) then these were corrected.

The author also explored *Zentralblatt MATH* and other archives, for the availability of a program similar to *bmref*, but without any success. Searches for a single article could be automated, but with fuzzy-matching this would result in multiple hits, requiring significant extra processing to determine whether the sought-after article actually had been located at that site.

In the context of a digital library, there is presumably a large database of metadata readily available which can be easily searched instead. Indeed then it

¹¹ See <http://www.ams.org/mathscinet-mref>.

¹² Of 516 separate cited items from 51 papers, 67 were not found automatically; 19 of these were found with some manual searching. The remainder were to journals not covered at *MathSciNet*, or to unpublished theses, etc.

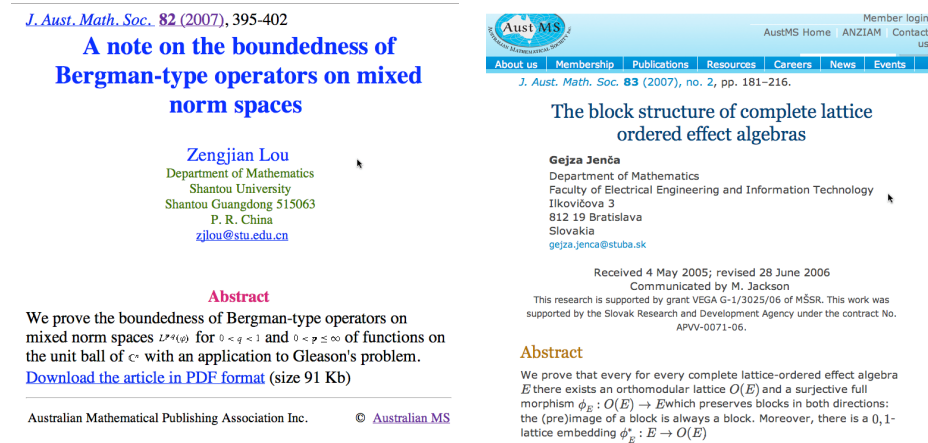


Fig. 8. *jsMath* in webpages: the image at right illustrates the high quality and proper scaling of mathematical expressions within a web-page that uses the *jsMath* applet software. This contrasts starkly with the image at left, displaying poorer quality and lack of scalability in the static images, used with older web technologies.

would be possible to include links not just to reviews of an article, but to the article itself. Some appropriate icon or symbol would then be used to indicate the kind of information to be found at the target of the hyperlink.

3 HTML Pages for Abstracts, etc.

At the end of 2007, the Australian Mathematical Society made a complete change in the hosting arrangements of its website¹³, as well as a change in the publication arrangements for its journals. There were still several journal issues¹⁴ that were not covered by the new arrangements, for which PDFs were still to be served from the Society's site. This necessitated the need for abstract pages which were publicly available, having hyperlinks to the PDFs which are accessible only with a subscription. The job of creating these pages fell to the current author, in the rôle of web-editor for the Society.

The same techniques that were used for the Bulletin PDFs were used for this task, only now there was no full article PDF being produced. Thus the principal outputs were the HTML pages and metadata files. MR-numbers were obtained in the same way as described above, only now the reference data was presented in a different way, as L^AT_EX-formatted .bb1 files. There were other differences in the metadata too, requiring some minor adaptations of coding used previously.

A significant addition however, was the choice to use *jsMath* for the mathematical expressions that would appear in titles and abstracts. Davide

¹³ Australian Mathematical Society's website: <http://www.austms.org.au/>.

¹⁴ J. Austral. Math. Soc., Vol. 83 (1)&(2): <http://www.austms.org.au/Publ/JAustMS/>, and ANZIAM Journal, Vol. 49 (1)&(2): <http://www.austms.org.au/Publ/ANZIAM/>.

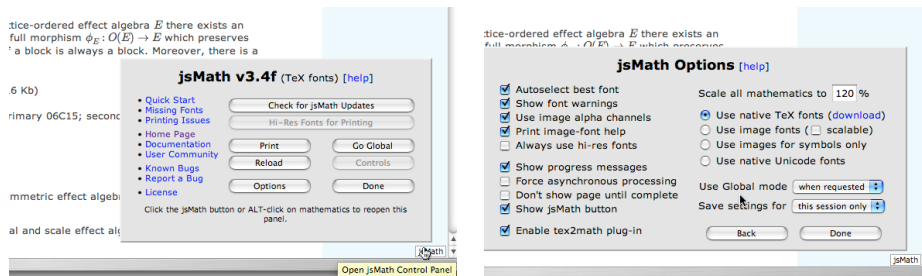


Fig. 9. Help, Options & fonts with *jsMath*: the image at right shows how to bring up the *jsMath* control panel, displaying the ready availability of options and help features. In the Options panel, shown at right, one sees the great flexibility in the way different fonts or images may be used to construct the mathematics. Best quality is obtained by downloading and installing the *jsMath* fonts, or using local Unicode fonts.

Cervone's *jsMath* software¹ is a JavaScript applet that is effectively a cut-down version of a \TeX compiler that does proper typesetting of stand-alone¹⁵ mathematical expressions. The author had used this before with the $\approx 5,000$ abstract submissions for the ICIAM '07¹⁶ Congress, and had made several suggestions for bug-fixes and other improvements that Cervone willingly implemented. As well as providing a better quality presentation of mathematics within webpages, *jsMath* also gives proper printing using fonts rather than images, and solves the problem of rescaling the mathematics to suit a web-surfer's choice of font size. Figure 8 displays this, by giving a comparison with a page produced using older methods.

All kinds of help is readily available, as Figure 9 shows. The *jsMath* 'Options' panel, see Figure 9, allows various choices of fonts to use for the mathematics. Specially prepared \TeX fonts can be downloaded and installed, which are then used with pages from any site that employs *jsMath*. This gives not only the best possible quality of image (since it leverages font-rendering machinery on the local operating system), but also speeds up processing since there is less information needing to be downloaded from that site. Alternatively, a local Unicode font could be used for similar speed gains¹⁷. Copy/paste and searching now refers to the Unicode code-points for mathematical symbols, whereas otherwise these operations would use the position in traditional \TeX encodings.

¹⁵ It doesn't do full pages, nor handle counters, cross-referencing, citations, etc.

¹⁶ Browse at the ICIAM '07 timetable: <http://www.iciam07.ethz.ch/timetable/>.

¹⁷ The layout of complicated math might not be tuned as with the *jsMath* \TeX fonts.