Math Font Encodings: A Workshop Summary

Abstract

The math font group is a joint venture of TEX Users Group and the LATEX3 project. Its aims are to investigate the requirements for mathematical typesetting using TEX, and to propose and implement new math font encodings which will satisfy these requirements. At the 1993 Aston TEX Users Group meeting, we held a workshop at which we discussed the needs for new math font encodings, and the work so far at meeting those needs. This paper is a summary of that workshop, for the benefit of those unable to attend. The panel consisted of Barbara Beeton, Alan Jeffrey, Frank Mittelbach, Chris Rowley and Justin Ziegler. There were many useful questions and suggestions from the audience.

Motivation

The current situation (as discussed by Berthold Horn in his stimulating paper *Where are the math fonts?*) is that there are over 14,000 text fonts available for use in TEX, but only six math fonts:

- Computer Modern
- Computer Concrete and Euler
- Lucida Math
- Lucida New Math
- MathTime
- Symbol

Each of these fonts uses different encodings, and each comes with its own selection of TEX macros. Although the Cork encoding is rapidly being established as the standard encoding for European Latin text, there is no similar encoding for mathematics. The result is:

- complex *ad hoc* macro packages for using each math font.
- it is difficult to set mathematics with Cork text, since the Cork encoding does not include the upper case Greek.
- installing PostScript math fonts such as Mathematical Pi is very difficult.

This is a bottleneck for uptake of the Cork fonts, and use of TEX for mathematical setting with anything other than the Computer Modern fonts.

The math font group (MFG, or Joint LATEX3 project / TEX Users Group Technical Working Group on Extended Math Font Encodings to give it its full title!) was formed in order to develop new encodings for setting mathematics.

These encodings should be fully upwardly compatible with `plain` TEX, LATEX, $\mathcal{AMS}$-TEX and $\mathcal{AMS}$-LATEX. The only effect most users should notice is that more symbols, and more math fonts will be available for use in TEX.

Overview

The MFG has developed an outline for a proposed math encoding, although the details of each encoding have yet to be worked out. There is still plenty of room for change!

The current math encoding proposal uses:

- T_1 'Cork' text encoding
- MC math core encoding
- MX math extension encoding
- MS_P math symbol primary encoding
- MS_1 math symbol additional 1 encoding
- MS_2 math symbol additional 2 encoding

In addition, we are proposing an TS_1 'text symbol' encoding, to hold the text glyphs such as '†' that are currently in math fonts.

Glyphs are being allocated to math encodings on the grounds of:

Glyph shape. All glyphs of a similar design should be in the same encoding. For example, all the Greek glyphs should live together, and the geometric symbols of similar appearance such as '⊕' and '⊗' should live together.

Kerning. Any glyphs which may need to have a kern pair should be in the same encoding. For example, one common request is for kerning between '*f*' and '(', and so these glyphs should live together. (The situation is somewhat more complex than this, since TEX will only kern or ligature when the first glyph is a math atom consisting only of a single-character mathord. See Rule 14 of Appendix G of *The TEXbook* for more details.)

Ligaturing. Any glyphs which may need to ligature should be in the same encoding.

Orthogonality. Each encoding should use as few different glyph styles as possible, to minimize the number of virtual fonts needed. For example, the Computer Modern Symbol encoding

includes roman glyphs, geometric symbols, calligraphic letters, and dingbats, and so a different VF is required for each combination of roman, geometric, calligraphic and dingbat font. A site with 100 text romans, four geometric symbol fonts, three calligraphic fonts, and three dingbat fonts might need $100 \times 4 \times 3 \times 3 = 3600$ VFs.

Slots. Some glyphs have preferred slots; for example it would be useful if the letter 'A' was always in slot 65.

None of the encodings will specify bold or sans glyphs, since these are expected to be kept in separate bold or sans math fonts, with the same encoding. The glyphs which are most commonly requested in bold will be placed in the MC encoding, so if many bold glyphs are used in a document, only one extra MC-encoded family containing bold glyphs needs to be used. If few bold glyphs are requested, these can be set using macros similar to `\boldsymbol` from $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX.

T_1 encoding

The T_1 (or Cork) encoding will be used for multi-letter identifiers such as 'log', 'sin' and 'lim'. Using the T_1 encoding allows arbitrary text fonts to be used for multi-letter identifiers. In many texts this will the same as the text roman, but this will not always be the case (for example Barandregt's *The Lambda Calculus*, North-Holland, 1984, has some multi-letter identifiers set in bold sans!).

This font will not normally be used for anything other than upper and lower case Latin letters. The symbol glyphs such as '+', '=' and '/' will be taken from the MS_P encoding.

Although the multi-letter identifier font will be T_1 encoded, it does not necessarily have to be a text font. In particular it may have the glyph width and italic correction adjusted to produce good subscript and superscript positioning, as long as this is not to the detriment of setting multi-letter identifiers.

Family 0 will contain a T_1 encoded font.

MC encoding

The MC encoding will contain:

- The default Latin letters (for example 'f').
- The default numerals (for example '1').
- The default punctuation (for example ',').
- The slanted and upright Greek in upper and lower case (for example 'α' and 'Γ').

Other glyphs (such as the math accents and Hebrew) will be included if there is space!

The font will also contain enough font dimensions to be used as `\fam2`, since the positioning of subscripts and superscripts depends much more on the math core font than the symbol fonts. It may also contain font dimensions for:

- Design size
- Suggested script and scriptscript design size
- Suggested values for `\mathsurround`, `\thickmuskip`, `\medmuskip` and `\thinmuskip`.

Family 2 will contain a MC encoded font.

MX encoding

The MX encoding will contain the extension glyphs from `cmex` and `ms*m`, plus frequently requested glyphs such as longer math accents, double brackets, and `\bigsqcap`.

Family 3 will contain an MX encoded font.

MS_i encodings

The MS_P, MS_1 and MS_2 encodings will contain the geometric glyphs from `cm*` and `ms*m`, plus frequently requested glyphs such as `\mapsfrom`. In addition:

- MS_P will contain calligraphic upper and lower case
- MS_1 will contain open (or 'inline' or 'outline' or 'blackboard bold') upper and lower case
- MS_2 will contain black letter (or 'fraktur') upper and lower case

There was quite a lively discussion about what to do with script upper and lower case! One possibility is to allow font implementors to replace the calligraphic letters by script letters in an MS_P font. Another is to ask that script letters be provided in T_1 encoded fonts. This point is still up for discussion.

All of the geometric glyphs used in `plain` TEX and LATEX will be kept in the MS_P encoding, so compatibility with `plain` TEX or LATEX can be achieved by loading four families, encoded as T_1, MS_P, MC and MX. Compatibility with $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX or $\mathcal{A}\mathcal{M}\mathcal{S}$-LATEX can be achieved by loading six families, encoded as T_1, MS_P, MC, MX, MS_1 and MS_2.

TS_1 encoding

There are a number of text glyphs that currently live in math fonts, such as '†' or '©'. These glyphs will be put into a 'text symbol' encoding, along with the Adobe standard and expert glyphs missing from the Cork encoding, such as 'f' (florin) and '½'.

The TS_1 encoding is not designed to be used in math mode.

Work to do

At the time of writing, there is still quite a lot of work to be done!

- Propose and document the math encodings.
- Implement the math encodings with METAFONT or virtual fonts.
- Provide user interfaces for `plain` TEX and LATEX.

If you would like to help with implementing or testing the new math font standards, please write to:

`math-font-request@cogs.susx.ac.uk`

We look forward to hearing from you!

◇ Barbara Beeton
American Mathematical Society
bnb@math.ams.org

◇ Alan Jeffrey
University of Sussex
alanje@cogs.susx.ac.uk

◇ Frank Mittelbach
LATEX3 project
mittelbach@mzdmza.zdv.uni-mainz.de

◇ Chris Rowley
Open University
c.a.rowley@open.ac.uk

◇ Justin Ziegler
Ecole des Mines de Saint Etienne
ziegler@educ.emse.fr