Implementation notes concerning the `mathtime` and `mathplus` math layouts

Ulrik Vieth

Abstract

This paper contains some implementation notes concerning the `mathtime` and `mathplus` math layouts in the new math setup.

1 Overview

The MathTime fonts are a set of professional-quality math typefaces in Adobe Type 1 format intended for use with text typeset in Times Roman. These fonts were designed by Michael Spivak's TEXplorators Corporation and are currently sold and distributed by Y&Y, Inc. They are available in two different sets:

- The basic "MathTime 1.1" distribution consists of just three fonts called `MTMI`, `MTSY`, and `MTEX`, covering most of the symbol complement of the old (LA)TEX math setup in the normal series.

- The supplementary "MathTime Plus" distribution consists of 13 fonts complementing the basic "MathTime 1.1" fonts. Most importantly, these include a revised version of the symbol font (`MTSYN`) and a bold and a heavy series of the three basic fonts. In addition, there is also a Script alphabet (`MTMS`, `MTMSB`) and an upright Greek alphabet (`MTGU`, `MTGUB`), both of which are available in a normal and a bold version.

The two sets together are sold at a reduced rate as "MathTime Complete".

The encodings used by the MathTime fonts are referred to as MY1, MY2, and MY3. They are derived from OML, OMS, and OMX, but they differ in some important aspects, as will be discussed in detail below.

In the new math setup, the MathTime math layout will come in two different variants, depending on whether the basic or the complete MathTime collection is available.

The `mathtime` layout will consist of just three fonts in the normal series encoded as MC, MSP, and MXP. These will be based only on glyphs taken from "MathTime 1.1" or Times Roman, which implies that they will unfortunately remain incomplete in certain aspects.

The `mathplus` layout will be available in the normal, bold, and heavy series, and each series will consist of three fonts encoded as MC, MSP, and MXP. They will be based on glyphs taken from "MathTime Complete", but even that will not be enough to cover all of the symbols in MSP and MXP.

2 Details

As mentioned in the introduction, the basic MathTime fonts `MTMI`, `MTSY(N)`, and `MTEX` are based on a set of non-standard encodings, referred to as MY1, MY2, and MY3, which are derived from OML, OMS, and OMX. In addition, the `MTMS` and `MTGU` fonts containing the Script and upright Greek alphabets from the "MathTime Plus" collection both use a font-specific encoding.

The MY1 encoding. The math italic font `MTMI` and its bold and heavy counterparts are encoded using the MY1 encoding. This encoding is mostly the same as OML, except that it leaves out the oldstyle digits, the small triangles, and some math accents to make room for the upright Greek capitals (normally taken from OT1), the lowercase Greek `\varkappa`, and the textstyle parentheses.

Since slot 127 in MY1 is allocated by a Greek letter, it cannot be used as the skewchar, which implies that another innocuous symbol has to be used for this purpose. In this case, the skewchar is the glyph 'arrowhookright' in slot 45.

It is important to note that the MY1 encoding makes it possible to have a number of additional kern pairs that are not normally found in OML-encoded fonts, such as between letters and opening parentheses or between upright Greek capitals and italic lowercase Greek or Latin letters. For a faithful representation of MathTime in the new math setup, it is important to ensure that all these kern pairs are preserved intact.

The MY2 encoding. The math symbol font `MTSY` is based on the MY2 encoding. This encoding is mostly the same as OMS, except that it leaves out the Script alphabet to make room for a variety of extra symbols.

Among these symbols we have the '+' and '=' signs (normally taken from OT1), a variant '=' for use as an arrow extension module, the math accents from OT1 and OML, the small triangles displaced from MY1, and a semicolon adjusted for math spacing. The revised symbol font `MTSYN` and its bold and heavy counterparts augment these symbols by a colon and exclamation mark, as well as the square brackets, all of which would otherwise be taken from a Times Roman text font in OT1.

Unlike the parentheses from MY1, the square brackets from MY2 and the remaining delimiters cannot be kerned with letters from the Greek or Latin alphabets, but taking them from a math font at least offers the advantage that it allows to have special versions with increased side-bearings.

The MY3 encoding. While the MY1 and MY2 exhibit a number of subtle differences from OML and OMS, the MY3 encoding is fully identical to OMX in the lower half. The only difference is that it also contains a copy of the textstyle radical (normally taken from OMS) in the upper half of the font table.

In the implementation of the MathTime math layout in the new math setup, this should allow to implement the MXP encoding based on MY3 without having to rely on the MY2-encoded symbol font. It would also allow to reencode the symbol font, so as to replace the lowered radical by a normal radical, better suited for use with non-TeXnical software.

The `MTMS` encoding. The Script alphabet in the "MathTime Plus" collection uses a straight-forward ASCII encoding for the default set of letters, but it also contains some variants which cannot easily be accounted for in the new math font encodings. A zero-width glyph named 'asterisknull' in slot 42 is used as the skewchar, while the default skewchar in the OMS encoding would be slot 48.

The `MTGU` encoding. The upright Greek alphabet in the "MathTime Plus" collection uses a font-specific encoding, which is aimed at facilitating the typesetting of short quotations of Greek text using a straight-forward transliteration scheme based on a tricky ligaturing program. In the new MC encoding, only the lowercase upright Greek alphabet will be taken from `MTGU` while the upright Greek capitals will come from `MTMI`. Since the Greek letters in `MTGU` were not designed for use in math mode, it remains to be seen if adjustments to the glyph positioning need to be applied for optimal results.

Virtual and real base fonts. On some platforms (e. g. Unix), the math italic font `MTMI` is actually implemented as a virtual font, combining glyphs from Times Italic and a raw font called `RMTMI`. On other platforms (e. g. Windows), `MTMI` is provided as a real font to avoid problems with ancient TeX systems that do not support virtual fonts.

In the implementation of the new math font encodings, it seems preferable to pretend that `MTMI` is a real font, on which the installation can be based. This way, it will be be possible to inherit the wisdom regarding optimal glyph placement from the base font without having to deal with the nitty-gritty internal details of the `MTMI` virtual font.

The same also applies for the bold and heavy counterparts `MTMIB` and `MTMIH` from the "MathTime Plus" collection, which are implemented in the same way as `MTMI`.

3 What can be implemented?

Before we go into the details of the implementation issues, it may be a good idea to clarify what can or cannot be implemented in the `mathtime` or `mathplus` math layouts.

In the `mathtime` math layout based exclusively on the basic "MathTime 1.1" fonts, only a subset of the MC, MSP, and MXP encoding can be implemented. There is no upright lowercase Greek alphabet in MC, nor is there a Script alphabet in MSP (unless Zapf Chancery is substituted as a Calligraphic alphabet for lack of a real Script alphabet).

In the `mathplus` math layout based on "MathTime Complete", both the upright lowercase Greek alphabet in MC and the Script alphabet in MSP can be filled in, but there will remain some missing symbols which cannot be provided in either case.

Apart from the upright lowercase Greek, the MC font table is reasonably complete, missing only a few Hebrew letters, some non-standard delimiters, and some inverted or reflected symbols. Most of the missing inverted or reflected symbols presumably could be realized with simple manipulations at the PostScript level, but it would be too complicated to do so in the virtual font implementation.

The situation is different, however, in the MSP font table, which happens to include quite a few extra symbols from the AMS or LaTeX symbol fonts that go beyond the symbol complement provided in the old (La)TeX math setup. Since these glyphs are not available in MathTime, one has to choose between faking the missing glyphs wherever possible or leaving them out. In the case of simple negated symbols, faking glyphs may be viable alternative, but it is clear that such methods will never produce the quality of ready-made glyphs.

Of course, the ideal long-term solution to this problem would be a revised version of the MathTime fonts providing a more extensive symbol complement, but that is very unlikely to happen before the proposed new math font encodings become widely accepted.

4 Implementation issues and problems

While it may seem straight-forward to implement the new math font encodings based on MY1, MY2, and MY3 rather than using OT1, OML, OMS and OMX as the set of base fonts, there are a number of details that need to addressed specifically.

The first thing to note is that there are some glyph naming conflicts that need to resolved. While the MC encoding follows Adobe Symbol in using 'Upsilon1' for the variant 'Y' of the Greek capital

'$\mathit{\Upsilon}$' in italic shape, the MathTime fonts use the glyph names 'Delta1' ... 'Omega1', and hence 'Upsilon1' for the set of upright Greek capitals.

The easiest way to resolve this problem seems to be to disregard the original glyphs names and use an `.etx` file to describe the encodings, so that 'Upsilon1' will be recognized as 'Upsilonupright'. An advantage of this approach is that it will make it unnecessary to reshuffle the kern pairs involving these glyphs in an `.mtx` file, which reduces the chances of accidently losing some of them.

A second important point to consider is the handling of kern pairs involving the skewchar. In this case, it will be unavoidable to do reshuffle the existing skewchar kern pairs along the lines of

```
\setkern{A}{skewchar}
  {\kerning{A}{arrowhookright}}
```

but a similar procedure seems to be needed in any case, regardless of which glyph serves as the skewchar in the underlying base font encoding. Of course, the same also applies to the Script alphabet in the MSP encoding, along the lines of

```
\setkern{Acal}{skewchar}
  {\kerning{Acal}{asterisknull}}
```

FIXME: Anything else???

◇ Ulrik Vieth
Heinrich-Heine-Universität
Institut für Theoretische Physik II
Universitätsstraße 1
D-40225 Düsseldorf, Germany
`vieth@thphy.uni-duesseldorf.de`