Implementation notes concerning the `mathptm` math layout

Ulrik Vieth

Abstract

This paper contains some implementation notes concerning the `mathptm` package and the corresponding math layout in the new math setup.

1 Overview

The `mathptm` package by Alan Jeffrey represents a cheap stop-gap solution for those who need some sort of math fonts to go with text typeset in Times Roman, but don't have access to professional-quality math font sets such as MathTime.

The `mathptm` package consists of four virtual fonts providing a replacement for the normal version of the standard (LA)TEX math families, encoded as OT1, OML, OMS, and OMX. There is no bold series of these fonts due to lack of appropriate base fonts. The virtual fonts are implemented using alphanumeric glyphs and Humanist symbols taken from standard PostScript fonts (Times, Adobe Symbol, Zapf Chancery) combined with geometric symbols taken from Computer Modern.

In the new math font setup, the `mathptm` math layout will consist of three fonts, encoded as MC, MSP, and MXP, while the the math operator font can be taken directly as a T1-encoded text font.

2 Details

The `mathptm` fonts are implemented as follows:

- The Latin alphabets in OT1 and OML are taken from Times Roman or Times Italic. For the italic letters, special provisions are needed to adjust the glyph positioning and the skewchar kerning (see below).
- The Greek alphabets in OT1 and OML are taken from Adobe Symbol (`psyr`), which means that all of them will be available in upright shape only. Although some TEX distributions include a slanted version of Adobe Symbol (`psyro`), the latter is not used, perhaps because it cannot be assumed to be available everywhere.
- The Calligraphic alphabet in OMS is taken from Zapf Chancery scaled by a factor of 1124/1000 to match the x-height of Times Roman. While some people find that Zapf Chancery is a bad choice for a mathematical Script alphabet, it seems to be the only choice available among the standard PostScript fonts. Besides, it also doesn't seem to be too different from the Script alphabet in Mathematica or the Calligraphic alphabet in the Lucida New Math font set.
- The so-called Humanist glyphs (i. e. "shapy" glyphs like $\aleph$, $\Re$, $\Im$, †, ‡, §, ¶) are taken from Times or Adobe Symbol.
- Most of the math accents are taken from Times Roman, except for the vector accent which does not exist in standard text fonts.
- All the remaining (mostly geometric) symbols, including punctuation, delimiters, binary operators and relations, are taken from Computer Modern. Although some of these glyphs, especially punctuation and some delimiters, could be taken from Times or Symbol just as well, the available symbol complement is insufficient for a complete implementation, and the outcome would become inconsistent anyhow.

3 Implementation issues and problems

Glyph positioning: When a faked math italic alphabet is set up using glyphs taken from a text italic font, special care has to be taken to ensure proper results.

- In some cases, it may be necessary to adjust the left side-bearing of the glyphs within their bounding boxes.
- The nominal glyph width is interpreted in math mode as the subscript position and must be adjusted for optimal placement of subscripts.
- The nominal italic correction it interpreted in math mode as the offset between the subscript and superscript position and must be adjusted for optimal placement of superscripts. The sum of the nominal width and the italic correction should correspond to the actual glyph width, including the left and right side-bearings.
- Finally, the kern pairs between letters and the skewchar are used to encode corrections to the placement of math accents and must be adjusted for optimal placement. In the new math setup, a similar scheme is used to adjust the placement of under-accents as well.

For the `mathptm` fonts using glyphs from Times Italic, one set of adjustment values has been derived by Alan Jeffrey, presumably by method of trial and error. Regardless of whether or not these values may have deficiencies, it might be wise to keep them unchanged to ensure metrics compatibility between the old and new implementation of the `mathptm` math layout. An alternative approach would be to steal the wisdom of professional-quality fonts such as MathTime, but that would break compatibility with the original `mathptm` package.

Greek letters: In the new math setup, the MC encoding will include two complete sets of Greek in upright and italic shape. This way, it will be possible to provide a slanted set of Greek taken from `psyro` for those who want it, while at the same time allowing to ensure compatibility with the original `mathptm` package (which included upright Greek from `psyr` only). Although the italic angle will not match exactly, it may be preferable to use the existing slanted version of Adobe Symbol rather than creating a new special-purpose version. On the other hand, relying on the existence of `psyro` might not work for all TeX distributions either.

Humanist and geometric symbols: In the new math setup, most of the Humanist symbols will find a place in the MC encoding, while MSP and MXP will consist mostly of geometric symbols. As a result of this arrangement, the MC font in the `mathptm` layout can be implemented almost exclusively using glyphs from Times and Symbol, while the MSP font in the `mathptm` layout will be almost the same as the Computer Modern version, with the exception of the Calligraphic alphabet being replaced by Zapf Chancery. In the MXP font, most glyphs will also be taken from Computer Modern, with the only exception of the $\sum$ and $\prod$ operators being taken from a scaled version of Adobe Symbol.

The only remaining question will be the choice which font set to use for the normalsize delimiters in the MC encoding. While the original `mathptm` implementation uses Computer Modern for these, there are situations in which this choice may lead to inconsistencies between delimiters in text mode and math mode set in different typefaces. For a faithful representation of the original `mathptm` package, using delimiters taken from Computer Modern will be the only way to go, even if that means extra complications in the implementation of the MC font encoding.

FIXME: Anything else???

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

 Task 3: Technical Studies