

New TX font package

Michael Sharpe

November 14, 2012

1 Introduction

The `txfonts` package by Young Ryu has been available for over ten years but has been corrected very little since its introduction. It is meant to be a complete text and math package with roman text font provided by Times or a Times clone, sans serif based on a Helvetica clone, typewriter faces, plus math symbol fonts whose math italic letters are from a Times Italic clone. The additional glyphs, whether taken from free sources or designed by him, are generally of high quality, but the accompanying metrics are not, making the math inconsistently spaced, but at times so cramped that neighboring symbols collide.

The goal of this new project is use his glyphs along with a few additions and with completely reworked metrics which are generally looser, but not as loose as Computer Modern math, to make a math package that works with not only Times, but also with fonts that are intermediate in weight between Computer Modern and Times. The free font Linux Libertine is one particular target—it is of nearly the same x-height as Computer Modern, but, not being a *modern* font, does not have a high contrast ratio, and so appears denser than Computer Modern but not as much so as Times. It is meant as a replacement for Times, but differs from it in many characteristics, more similar to MinionPro than Times, and provides a better range of variants than Times—three weights (regular, semi-bold and bold) rather than just two, and has expert features in all weights: old-style figures, more extensive and more interesting ligatures, and small caps. In my opinion, material typeset in Linux Libertine looks better than the corresponding material typeset in Times. This seems especially true on the screen. As of version 1.0, the package also offers support for MinionPro as a math font, but with limitations described in detail below.

This new package differs from `txfonts` in the following ways:

- the new package is split into separate text and math packages that do not need to be used in conjunction;
- both text and math packages offer options not present in the original package, described below, including the option to use libertine Latin and Greek letters to replace Times, as well as a similar option minion;
- wide accent glyphs have been corrected (they should have zero depth) so that they no longer collide with the underlying glyph;
- for those who do not like the integral in `txfonts`, an emboldened version of the Computer Modern integral is made available, matching the weight of the `txfonts` symbols;
- an upright partial derivative symbol has been added, named `\uppartial`;
- there is now an option to get braces more pleasing to older eyes;

- macros have been added to bring the calls to Greek symbols more into conformity with `PSNFSS` and `Mathtime Pro 2`;
- problems using AMS macro packages before `txfonts` are settled;
- `\coloneq` and `\eqcolon` now point to the correct glyphs;
- The problem with the ogonek accent and tabular environments (bad definition of `\k`) is fixed;
- The default encoding for `newtxtext` is now T1, but support is offered also for OT1 and LY1. As some add-on packages are available only in T1, that seems the best current choice.

2 Text mode options

The text mode environment invoked by

```
\usepackage{newtxtext}
```

has two options: you may write

```
\usepackage[scaled=.93]{newtxtext}
```

to load the roman and typewriter text fonts at 93% of normal size, and the sans serif (Helvetica clone) at scale $0.9 * 0.93$. This is not of much utility if the package is used with the math package `newtxmath` to which it is already matched, but may be with other math packages. The option `osf` instructs the text fonts to use old-style figures rather than the default lining figures 1234567890.

3 Math mode options

The package invoked by

```
\usepackage{newtxmath}
```

loads the math part of the `txfonts` (with revised metrics and additional glyphs) and should be loaded *after* the text font and its encoding have been specified, as it uses the text font settings to define how operators, numbers, math accents, `\mathrm`, `\mathbf` etc. are rendered. You should also load a Typewriter font so as not to generate mysterious error messages about metafont trying to generate `ectt10`. The package offers a number of options.

- `varg` causes the math italic letters `g`, `v`, `w`, `y` to be replaced by versions which are more distinctive—eg, useful for distinguishing math italic `v` from `\nu`;
- `libertine` loads different versions of math italic and bold math italic based on Libertine rather than Times—the `varg` option is disabled in this case, as the equivalent variant forms are made available by default;
- `minion` loads different versions of math italic and bold math italic based on MinionPro rather than Times—the `varg` option is disabled in this case, as the equivalent variant forms are made available by default—see the extended discussion below;
- the `libertine` option also replaces both slanted and upright Greek symbols by the corresponding Libertine glyphs, and similarly for `minion`;

- `cmintegrals` instructs `newtxmath` to load a thicker version of the Computer Modern integral in place of the `newtxmath` default—the `txfonts` integral (identical to the integral in the Wolfram fonts), which is not to everyone’s taste—a consequence is that none of the special forms of `txfonts` integrals are available;

- the combination

```
\usepackage{amsmath}% loads amstext, amsbsy, amsopn but not amssymb
\usepackage{newtxmath}
```

causes no error, unlike the same combination with `txfonts`, but does nothing significant. (Recall that `amsmath` is loaded automatically if you use an AMS document class such as `amstext` or `amsbook`, as is `amstex`.) The integrals are as defined in `txfonts`. On the other hand,

```
\usepackage{amsmath}
\usepackage[cmintegrals]{newtxmath}
```

allows you to use the forms `\iint`, `\iiint`, `\iiiiint` and `\idotsint` defined in `amsmath`, but using the pumped-up Computer Modern integral loaded by `newtxmath`.

- `uprightGreek` and `slantedGreek` determine the form of Greek alphabet loaded—the default is `uprightGreek`, which loads upright uppercase and slanted lowercase Greek symbols, as is customary in Anglo-American mathematical typesetting. With the option `slantedGreek`, which you might want to use if you cared about ISO standards, all Greek symbols are slanted. No matter which is set, `\Gammaup` (or `\upGamma`) gives you upright Γ , etc.
- The option `cmbraces` instructs `newtxmath` to ignore the brace collections from `txfonts`, substituting a collection based on thickened versions of the Computer Modern braces, which I find much easier to distinguish from other delimiters. This works quite well in regular weight but looks a bit clunky in bold.
- The `newtxmath` package contains three different Blackboard Bold alphabets, where original `txfonts` contained two. The default, triggered by `\mathbb{}`, takes its glyphs from the font which replaces `msbm` and has the same overall appearance of a hollowed-out text font, which I find neither bold nor blackboard-like. The second option, taken from `txfonts`, is triggered by `\varmathbb{}`, is more geometric and, in my opinion, preferable but not optimal. The option `varbb` makes `\mathbb{}` synonymous with `\varmathbb{}`. The third option is the double-struck glyphs from the STIX collection. See the expanded discussion below.
- `nosymbolsc` causes the package to not load the `symbolsc` fonts, saving a math family. (This font contains mostly exotic symbols, along with some very useful, commonly used symbols like `\coloneq` \coloneqq , `\eqcolon` \equiv , `\notin` \notin , `\notni` \ni , `\neq` \neq , `\subset` \subset and `\supset` \supset , but these have been moved (virtually) to `lettersA` so they may continue to be used even if you use the option `nosymbolsc`.)
- `amssymbols` (the default) and `noamssymbols` determine whether the `txfonts` versions of the AMS symbols (`msam`, `msbm`) are loaded—if so, they override previous settings in `amsmath`. If you use the option `noamssymbols`, then `\mathbb{}` is set to mean the same as `\varmathbb{}`. (One advantage of `noamssymbols` is that you save two of your precious math families for other purposes, such as setting a couple of external math alphabets by means of the `mathalfa` package.)
- `libaltw` has effect only if the `libertine` option is selected—in this case, it substitutes for math italic *v* and *w* hand-crafted versions based on the Libertine υ glyphs.

- `largelibfigs` has effect only if the `libertine` option is selected—with this option, full-sized figures are substituted in math mode for the default Libertine figures, which are about 8% below Libertine’s Capheight.

IMPORTANT: The Libertine text package is now `libertine-type1`, which requires arguments that are different from the original `libertine` package.

EXAMPLE 1:

```
\usepackage{newtxtext} % T1, lining figures so math uses lf
\usepackage{textcomp} % required for special glyphs
\usepackage{amsmath}
\usepackage[varg,cmintegrals,cmbraces]{newtxmath}
\usepackage{bm} % load after all math to give access to bold math
\renewcommand{\rmdefault}{ntxrj} %old-style figures in text, not in math
```

EXAMPLE 2:

```
\usepackage[lining,semibold]{libertine-type1} % a bit lighter than Times--no osf in math
\usepackage[T1]{fontenc} % best for Western European languages
\usepackage{textcomp} % required to get special symbols
\usepackage[scaled=.85]{beramono}% a typewriter font must be defined
\usepackage{amsmath,amsthm}
\usepackage[libertine,cmintegrals,cmbraces,vvmathbb]{newtxmath}
\usepackage[scr=rsfs]{mathalfa}
\usepackage{bm}% load after all math to give access to bold math
\renewcommand*{\rmdefault}{LinuxLibertine0-0sF} %old-style figures in text, not math
% use T0sF for tabular figures rather than proportional
```

Caution: In encodings T1 and LY1, the macro `\mathrm` does not produce what you might expect for Greek letters, due to the difference between those encodings and OT1. But what were you expecting with `\mathrm{\Gamma}` anyway? In T1 and LY1, you’ll get a math accent.

4 The minion option

This option allows the use of MinionPro as math letters (Latin and Greek) within the math font, but there are some caveats:

- you must use a recent version of MinionPro, such as the version that comes with recent versions of Adobe Reader. In addition, the fonts must be installed under special names so they can be recognized by the support files—this is spelled out in the documentation for the `minion2newtx` package that must be downloaded from CTAN.tug.org and installed separately.

5 The T_EX math font problem

Math font packages in T_EX are susceptible to the “Too many math alphabets” error, due to exceeding the limit of just sixteen math font families, or `mathgroups`, as they are called in T_EX. Put in oversimplified terms that do however correctly represent how this all works in `newtxmath`, the following math fonts

are always loaded and permanently (if you didn't prevent loading of some features) occupying slots immediately following `\begin{document}`, and others that may be called for in typesetting a mathematical expression can add to the list as the document grows.

ALWAYS LOADED:

```
0: operators
1: letters
2: symbols
3: largesymbols
4: AMSa
5: AMSb
6: lettersA
7: symbolsC
8: largesymbolsA
9: boldoperators
10: boldletters
11: boldsymbols
12: boldlargesymbols
```

NOTES:

- The `operators` font is essentially the Roman text font, used for names of operators and as the target for `\mathrm`—its bold version is used as the target for `\mathbf`;
- if you typeset an expression that, say, calls for a single bold glyph from `symbolsC`, that costs you an entire new slot, leaving only two remaining;
- same with, eg, `\mathit`;
- same with an external Fraktur, Blackboard Bold or Script glyphs;
- if you run out of math alphabets, look first to dropping `AMSa` and `AMSB`, as well as `symbolsC`, which can save you three slots;
- if space is tight, do not call bold versions of the fonts listed above where the bold version is not already loaded, to avoid loading a new mathgroup;
- if you absolutely need a letter (not a math symbol) from some math font that would normally cost you another mathgroup, you might consider using as if it were text, with something like

```
\mbox{{\usefont{U}{ntxmia}{b}{n} X}}
```

which allows you to use letter X from `boldlettersA` but without any math features;

- there is a macro `\ShowMathFonts` in `newtxmath` which may be called at any point in your document, which will provide you a list of the mathgroups currently in use. This can be helpful in figuring out where problems occur. The output lines take the form

```
(<fam number>: <internal font id> = <tfm name> [newtx name])
```

6 Bold math fonts

Every math font in the `txfonts` package, and in the `newtx` package, is accompanied by a bold version. Some usage examples are given below. Generally, one may use either `\boldmath` to change an entire formula to bold, or `\boldsymbol` to change one symbol, but the spacing generally works better after loading the `bm` package and using the `\bm` macro.

The text glyphs `\i` and `\j` are rarely needed in actual text—in many fonts, roman `\i` is very similar to the numeral 1. They are however sometimes needed to build special math glyphs. The following table shows how to generate the mathematical forms of `\i` and `\j`. I illustrate with only `\i`—`\j` is entirely analogous.

Type	Weight	Command	Result
Math Italic	Regular	<code>\hat{\imath}</code>	\hat{i}
Math Italic	Bold	<code>\bm{\hat{\imath}}</code>	$\hat{\boldsymbol{i}}$
Roman	Regular	<code>\hat{\textrm{i}}</code>	\hat{i}
Roman	Bold	<code>\bm{\hat{\textbf{i}}}</code>	$\hat{\boldsymbol{i}}$

In math, bold roman characters are often used to indicate vector quantities, and for this one uses constructions like

- `\mathbf{x}` produces \mathbf{x} ;
- `\bm{\mathrm{x}}` produces \mathbf{x} (same as previous but may offer improved spacing);
- `\bm{x}` produces \mathbf{x} (not roman);
- `\bm{\hat{\mathbf{x}}}` produces $\hat{\mathbf{x}}$ (with a bold accent);
- `\bm{\dot{x}}` produces $\dot{\mathbf{x}}$ (bold but not roman);
- the macro definition

```
\newcommand{\xyvec}[2]{%
\ensuremath{\#1\mkern1.5mu\bm{\textbf{\i}}\#2\mkern1.5mu\bm{\textbf{\j}}}}
```

allows you to write `\xyvec{x}{+y}` to output $x\mathbf{i} + y\mathbf{j}$;
- `\bm{\Gamma}` produces Γ (bold Gamma);
- `\bm{\hat{\Gamma}}` produces $\hat{\Gamma}$ (bold Gamma with bold accent).

(The last two assume that you have effectively set the option `uprightGreek`—the default.)

7 Blackboard bold

As mentioned briefly above, the package now has a new blackboard bold alphabet built-in, and some new macros to call the non-default versions. To summarize, the problems are:

- the default, called by `\mathbb` is quite ugly and indistinct, but does cooperate with the bolding macros `\bm` and `\boldsymbol`;
- the original variant form called with the macro `\varmathbb` is better, but the macro conflicts with `\bm`. (In fact, the bold version is identical to the regular version, but it is still not right that it conflicts

with them.) The problem stems from the part of the definition of `\varmathbb` which allows you to insert an argument with more than one character, like `\varmathbb{ABC}`. The solution is to make a new macro that takes a single argument, like `\vmathbb{A}`, which cooperates with `\bm`.

- the third, new, alphabet is borrowed from the STIX fonts—it is sharp and quite clear, geometric in design. The only slight problem is that it lacks a full bold version, with only the letters C, D, H, N, P, Q, R, Z, d, e, i and j available. The macro `\vmathbb` provided to access this alphabet allows only a single character for its argument. The new option `vvarbb` effectively makes `\mathbb` mean the same as `\vmathbb`. (The reason for including these in the `newtx` package rather than calling them from the `mathalfa` package is that `newtx` leaves very little space for new math symbol fonts and math alphabets, and this way requires no additional such resources.

One interesting feature of the new alphabet is that it contains blackboard bold numbers, of which 0 and 1 will likely be the most useful, perhaps as operator names. I find `\mathbb{1}` useful in specifying an indicator, AKA characteristic function. Here are some examples:

	Regular	Bold	Remarks
Default	<code>\mathbb{R}</code> \mathbb{R}	<code>\bm{\mathbb{R}}</code> \mathbb{R}	
Variant 1	<code>\varmathbb{R}</code> \mathbb{R}		Bold is same as regular
... or	<code>\vmathbb{R}</code> \mathbb{R}		Single char. argument
Variant 2	<code>\vmathbb{R}</code> \mathbb{R}	<code>\boldmath \vmathbb{R}</code> \mathbb{R}	Make a macro for this!

8 Samples from free Times and Libertine packages

TXFONTS:

`\usepackage{txfonts}`

A formula from the *L^AT_EX Companion*, 2nd Edition, p.390:

$$t[u_1, \dots, u_n] = \sum_{k=1}^n \binom{n-1}{k-1} (1-t)^{n-k} t^{k-1} u_k.$$

The ISO would prefer that a formula like

$$\Phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^u e^{-t^2/2} dt$$

be typeset instead as

$$\Phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^u e^{-t^2/2} dt,$$

with upright π , e and d. I dislike the look of dt when the slope of t is too great.

- Complete match between text and math size and weight;
- first formula much too cramped;
- upper limit of integral much too close to integral sign;
- square on t in integrand comes very close to colliding with it;

- square root in denominator aligned too far right.

NEWTXFONTS:

```
\usepackage{newtxtext}
\usepackage{newtxmath}
```

A formula from the \LaTeX Companion, 2nd Edition, p.390:

$$t[u_1, \dots, u_n] = \sum_{k=1}^n \binom{n-1}{k-1} (1-t)^{n-k} t^{k-1} u_k.$$

The ISO would prefer that a formula like

$$\Phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^u e^{-t^2/2} dt$$

be typeset instead as

$$\Phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^u e^{-t^2/2} dt,$$

with upright π , e and d. I dislike the look of dt when the slope of t is too great.

- Complete match between text and math size and weight;
- first formula much less cramped;
- upper limit of integral not too close to integral sign;
- square not too close to t in exponent;
- better alignment of square root in denominator.

MATHTIMEPRO2:

```
\usepackage{newtxtext}
\usepackage[lite]{mtpro2}
```

A formula from the \LaTeX Companion, 2nd Edition, p.390:

$$t[u_1, \dots, u_n] = \sum_{k=1}^n \binom{n-1}{k-1} (1-t)^{n-k} t^{k-1} u_k.$$

The ISO would prefer that a formula like

$$\Phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^u e^{-t^2/2} dt$$

be typeset instead as

$$\Phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^u e^{-t^2/2} dt,$$

with upright π , e and d. I dislike the look of dt when the slope of t is too great.

- Complete match between text and math size and weight;
- first formula quite spread out;
- upper limit of integral not too close to integral sign;
- plenty of space between square and t in exponent.

LIBERTINE AND MATHTIMEPRO2:

```
\usepackage{libertine}
\usepackage[T1]{fontenc}
\usepackage[lite]{mtpro2}
```

A formula from the \LaTeX Companion, 2nd Edition, p.390:

$$t[u_1, \dots, u_n] = \sum_{k=1}^n \binom{n-1}{k-1} (1-t)^{n-k} t^{k-1} u_k.$$

The ISO would prefer that a formula like

$$\Phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^u e^{-t^2/2} dt$$

be typeset instead as

$$\Phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^u e^{-t^2/2} dt,$$

with upright π , e and d . I dislike the look of dt when the slope of t is too great.

- Mismatch of weight between text and math;
- first formula quite spread out;
- upper limit of integral not too close to integral sign;
- plenty of space between square and t in exponent.

LIBERTINE AND NEWTXMATH:

```
\usepackage{libertine}
\usepackage[T1]{fontenc}
\usepackage[libertine]{newtxmath}
```

A formula from the \LaTeX Companion, 2nd Edition, p.390:

$$t[u_1, \dots, u_n] = \sum_{k=1}^n \binom{n-1}{k-1} (1-t)^{n-k} t^{k-1} u_k.$$

The ISO would prefer that a formula like

$$\Phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^u e^{-t^2/2} dt$$

be typeset instead as

$$\Phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^u e^{-t^2/2} dt,$$

with upright π , e and d. I dislike the look of dt when the slope of t is too great.

- Very good match between text and math in size and weight;
- first formula not cramped;
- upper limit of integral not too close to integral sign;
- space between square and t in exponent;
- better alignment of square root in denominator.

`MATHPTMX:`

`\usepackage{mathptmx}`

A formula from the \LaTeX Companion, 2nd Edition, p.390:

$$t[u_1, \dots, u_n] = \sum_{k=1}^n \binom{n-1}{k-1} (1-t)^{n-k} t^{k-1} u_k.$$

The ISO would prefer that a formula like

$$\Phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^u e^{-t^2/2} dt$$

be typeset instead as

$$\Phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^u e^{-t^2/2} dt,$$

with upright π , e and d. I dislike the look of dt when the slope of t is too great.

- Good match between text and math size and weight, though the summation symbol (from the system `symbol` font) is too small and too dark;
- first formula well spread;
- upper limit of integral not too close to integral sign;
- space between square and t in exponent;
- there are no upright Greek lowercase letters in this package;

- good alignment of square root in denominator;
- infinity symbol not sufficiently large?
- the package lacks a number of amenities that are present in other packages.

9 Items installed

As well as a collection of PostScript fonts, virtual fonts, font definition files and the central `newtxtext.sty` and `newtxmath.sty` files, the package contains one map file `newtx.map` that must be enabled for the package to function correctly. Its name was changed from `ntx.map` to mirror the package name.) The file `implementation.pdf` in this distribution provides a manifest of all files installed together with a brief indication of the sources.

The font files `ntxexmods.pfb` and `ntxbexmods.pfb` were derived from `cmex10.pfb` by FontForgery, thickening the Computer Modern braces to match the weight of the `txfonts` braces. The pair `ntxexb.pfb` and `ntxbexb.pfb` were similarly derived from `cmsy7.pfb` and `cmex10.pfb` to produce more braces and matching integral signs based on Computer Modern. The `.tfm` files `rtx[b]mio.tfm` are simply unslanted versions of `rtxmi`, from which we construct upright partial derivative symbols. The last two entries provide us with a way to access custom-encoded versions of `fxlri.pfb` and `fxlbi.map` in order to access some of the unencoded alternate characters—eg, Greek letters, `J.alt` and `v.alt`. The font file `LibertineTheta-Regular.pfb` was created from the Theta symbol in `fxlri.pfb`, which requires some FontForge help to look correct.

This version contains optical versions of the math italic and symbol fonts at 7pt and 5pt, allowing better rendering in `\scriptstyle` and `\scriptscriptstyle`.