XCharter-Math

Daniel Flipo
daniel.flipo@free.fr

4th May 2022

1 What is XCharter-Math?

XCharter-Math is an Opentype mathematical font based on Bitstream Charter meant to be used with XCharter text fonts.

Latin and Greek letters and many Math symbols are borrowed or derived from Mickael Sharpe's XCharter fonts. Other sources for Math glyphs were found in MathDesign (by Paul Pichaureau) and Fourier-GUT*enberg* (by Michel Bovani).

It requires LuaTeX or XeTeX as engine and the `unicode-math` package[1].

Please note that the current version (0.32) is *experimental, do expect metrics and glyphs to change* until version 1.0 is reached. Comments, suggestions and bug reports are welcome!

2 Usage

2.1 Calling `\setmathfont`

A basic call for XCharter-Math would be:

```
\usepackage{unicode-math}
\setmathfont{XCharter-Math.otf} % Call by file name or
\setmathfont{XCharter Math}     % Call by font name
```

this loads XCharter-Math as math font with the default options, see subsections 3.1 on page 3, 3.2 on page 4 and 3.3 on page 4 for customisation.

Please note that the three sets of text fonts have to be chosen separately, f.i.:

```
\setmainfont{XCharter}
\setsansfont{Cabin}[Scale=MatchLowercase]          % sf
\setmonofont{Inconsolatazi4}[Scale=MatchLowercase] % tt
```

otherwise you would get Latin Modern for text fonts.

[1] Please read the documentation `unicode-math.pdf`.

2.2 Calling `xcharter-otf.sty` (recommended)

As an alternative to load XCharter-Math you can type:

`\usepackage[` *options* [2] `]{xcharter-otf}`

it loads `unicode-math` with the default options, sets XCharter-Math as Math font and XCharter Text fonts as Roman fonts (families *sf* and *tt* left unchanged) but does a bit more:

1. it loads `realscripts` for better superscripts;
2. it checks at `\begin{document}` if packages `amssymb` or `latexsym` are loaded and issues warnings in case they are;
3. it provides aliases for glyphs named differently in Unicode, so that `latexsym` or AMS names are also available;
4. it defines specific Math characters like `\varemptyset` (∅), `\parallelslant` (∥), `\shortparallelslant` (⫽), etc.;
5. it reduces spacing in math mode: `\thinmuskip`, `\medmuskip` and `\thickmuskip` unless the `loose` option is activated.

Apart from the `loose` option mentionned above, `xcharter-otf.sty` provides two options `no-text` and `Scale=`<*decimal*> meant to be used to load the XCharter-Math font together with roman text fonts other than XCharter, while keeping the advantages 1. to 5. pointed in the preceeding list, f.i.

```
\usepackage[no-text,Scale=0.98]{xcharter-otf}
```

Option `no-text` can also be useful if XCharter is to be loaded with specific options, f.i.

```
\usepackage[no-text]{xcharter-otf}
\setmainfont{XCharter}[RawFeature=+onum;+ss01]
```

3 What is provided?

XCharter-Math provides all glyphs supplied by Fourier-GUT*enberg* plus all glyphs available in the `amssymb` and `latexsym` packages and many more. Therefore, these two packages *should not* be loaded as they might override XCharter-Math glyphs.

Sans-serif, typewriter and fraktur styles are borrowed from Latin Modern fonts. See in section 3.6 on page 8 how to choose from other Math fonts for these styles.

A full list of available glyphs is shown in file `unimath-xcharter.pdf`.

[2] Possible *options* are `loose`, `no-text`, `Scale=` or any of the options described in sections 3.1, 3.2 and 3.3.

3.1 Upright or slanted?

Package `unicode-math` follows TEX conventions for Latin and Greek letters: in math mode, the default option (`math-style=TeX`) prints Latin letters $a\ldots z$ $A\ldots Z$ and lowercase greek letters $\alpha\ldots\omega$ slanted (italic) while uppercase greek letters $\mathrm{AB}\Gamma\ldots\Omega$ are printed upright. This can be changed by option `math-style` as shown in table 1.

Table 1: Effects of the `math-style` package option.

Package option	Latin	Greek
`math-style=ISO`	(a,z,B,X)	$(\alpha,\beta,\Gamma,\Xi)$
`math-style=TeX`	(a,z,B,X)	$(\alpha,\beta,\Gamma,\Xi)$
`math-style=french`	$(a,z,\mathrm{B},\mathrm{X})$	$(\alpha,\beta,\Gamma,\Xi)$
`math-style=upright`	$(\mathrm{a},\mathrm{z},\mathrm{B},\mathrm{X})$	$(\alpha,\beta,\Gamma,\Xi)$

Bold letters are printed upright except lowercase Greek letters which are slanted (the default option is `bold-style=TeX`). This can be changed by option `bold-style` as shown in table 2.

Table 2: Effects of the `bold-style` package option.

Package option	Latin	Greek
`bold-style=ISO`	$(\boldsymbol{a},\boldsymbol{z},\boldsymbol{B},\boldsymbol{X})$	$(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\Gamma},\boldsymbol{\Xi})$
`bold-style=TeX`	$(\mathbf{a},\mathbf{z},\mathbf{B},\mathbf{X})$	$(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\Gamma},\boldsymbol{\Xi})$
`bold-style=upright`	$(\mathbf{a},\mathbf{z},\mathbf{B},\mathbf{X})$	$(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\Gamma},\boldsymbol{\Xi})$

Other possible customisation: ∇ is printed upright and ∂ is printed slanted by default, but `nabla=italic` and `partial=upright` can change this.

All these options are offered by the `unicode-math` package but they can be added to the `\setmathfont` call[3], for example:

`\setmathfont{XCharter-Math.otf}[math-style=french,partial=upright]`

will print for the code

```
\[ \frac{\partial f}{\partial x} = \alpha \symbf{V} + a\nabla\Gamma
                                 + \symbf{\beta}\symbf{M} \]
```

$$\frac{\partial f}{\partial x} = \alpha\mathbf{V} + a\nabla\Gamma + \boldsymbol{\beta}\mathbf{M}$$

while the default settings would print

$$\frac{\partial f}{\partial x} = \alpha\mathbf{V} + a\nabla\Gamma + \boldsymbol{\beta}\mathbf{M}$$

Both shapes remain available anytime: `$\uppi,\itpi$` prints π,π.

If your text editor is able to handle greek letters or math symbols, they can be entered in the code instead control sequences (i.e. α, β, Γ,… for `\alpha`, `\beta`, `\Gamma`,…).

[3] IMHO it is easier to add *all options* to the `\setmathfont` command.

3.2 Character variants

XCharter-Math provides fourteen "Character Variants" options, listed on table 3, to choose between different glyphs for Greek characters and some others. Alternative calligraphic capitals have been added for E, Q and T in version 0.50.

Table 3: Character variants.

	Default	Variant	Name
cv00	0	𝟎	0
cv01	ℏ	ℏ	\hslash
cv02	∅	∅	\emptyset
cv03	ϵ	ε	\epsilon
cv04	κ	$\varkappa$	\kappa
cv05	π	ϖ	\pi
cv06	ϕ	φ	\phi
cv07	ρ	ϱ	\rho
cv08	σ	ς	\sigma
cv09	θ	ϑ	\theta
cv10	Θ	ϴ	\Theta
cv20	$\mathcal{E}$	$\mathscr{E}$	\symcal{E}
cv21	$\mathcal{Q}$	$\mathscr{Q}$	\symcal{Q}
cv22	$\mathcal{T}$	$\mathscr{T}$	\symcal{T}

For instance, to get \epsilon and \phi typeset as ε and φ instead of ϵ and ϕ, you can add option CharacterVariant={3,6} to the \setmathfont call:

```
\setmathfont{XCharter-Math.otf}[CharacterVariant={3,6}]
```

This works for all shapes and weights of these characters: f.i. $\symbf{\epsilon}$, $\symbf{\phi}$ are output as $\boldsymbol{\varepsilon}$, $\boldsymbol{\varphi}$ instead of $\boldsymbol{\epsilon}$, $\boldsymbol{\phi}$.

Similarly with math-style=french, \epsilon and \phi are output as ε and φ (upright).

Please note that curly braces are mandatory whenever more than one "Character Variant" is selected.

Note about \hbar (v 0.43): unicode-math defines \hbar as \hslash (U+210F) while amsmath provides two different glyphs (italic h with horizontal or diagonal stroke). XCharter Math now follows unicode-math; the italic h with horizontal stroke can be printed using \hslash or \hbar together with character variant cv01 or with \mithbar (replacement for AMS' command \hbar).

3.3 Stylistic sets

XCharter-Math provides five "Stylistic Sets" options to choose between different glyphs for families of mathematical symbols.

StylisticSet=4, alias[4] Style=leqslant, converts large inequalites into their slanted variants, see table 5a.

StylisticSet=5, alias Style=smaller, converts some symbols into their smaller variants, see table 5b.

Table 4: Stylistic Sets 4 and 5

(a) Style=leqslant (+ss04)

Command	Default	Variant
\leq	$\leq$	$\leqslant$
\geq	$\geq$	$\geqslant$
\nleq	$\not\leq$	$\not\leqslant$
\ngeq	$\not\geq$	$\not\geqslant$
\leqq	$\leqq$	$\leqslant$
\geqq	$\geqq$	$\geqslant$
\eqless	$\eqless$	$\leqslant$
\eqgtr	$\eqgtr$	$\geqslant$
\lesseqgtr	$\lesseqgtr$	$\lesseqgtr$
\gtreqless	$\gtreqless$	$\gtreqless$
\lesseqqgtr	$\lesseqqgtr$	$\lesseqqgtr$
\gtreqqless	$\gtreqqless$	$\gtreqqless$

(b) Style=smaller (+ss05)

Command	Default	Variant
\in	$\in$	$\in$
\ni	$\ni$	$\ni$
\mid	$\mid$	$\mid$
\nmid	$\nmid$	$\nmid$
\parallel	$\parallel$	$\parallel$
\nparallel	$\nparallel$	$\nparallel$
\parallelslant	$\parallelslant$	$\parallelslant$
\nparallelslant	$\nparallelslant$	$\nparallelslant$

StylisticSet=6, alias Style=subsetneq, converts some inclusion symbols, as shown in table 6a.

StylisticSet=7, alias Style=parallelslant, converts "parallel" symbols into their slanted variants, see table 6b.

Table 5: Stylistic Sets 6 and 7

(a) Style=subsetneq (+ss06)

Command	Default	Variant
\subsetneq	$\subsetneq$	$\subsetneq$
\supsetneq	$\supsetneq$	$\supsetneq$
\subsetneqq	$\subsetneqq$	$\subsetneqq$
\supsetneqq	$\supsetneqq$	$\supsetneqq$

(b) Style=parallelslant (+ss07)

Command	Default	Variant
\parallel	$\parallel$	$\parallelslant$
\nparallel	$\nparallel$	$\nparallelslant$
\shortparallel	$\shortparallel$	$\shortparallelslant$
\nshortparallel	$\nshortparallel$	$\nshortparallelslant$

To enable Stylistic Sets 4, 6 and 7 for XCharter-Math, you should enter

```
\setmathfont{XCharter-Math.otf}[StylisticSet={4,6,7}]  or
\usepackage[Style={leqslant,subsetneq,parallelslant}]{xcharter-otf}
```

[4]These Style aliases are provided by xcharter-otf.sty.

then, `\[x\leq y \quad A \subsetneq B \quad D \parallel D'\]` will print as

$$x \leqslant y \quad A \subsetneq B \quad D \parallel D'$$

instead of

$$x \leq y \quad A \subsetneq B \quad D \parallel D'$$

`StylisticSet=3`, alias[5] `Style=upint`, converts integrals signs into their upright variants, see table 6.

Table 6: Style=upint (+ss03)

Command	`\int`	`\iint`	`\iiint`	`\iiiint`	`\oint`	`\oiint`	`\oiiint`	
Default	$\int$	$\iint$	$\iiint$	$\iiiint$	$\oint$	$\oiint$	$\oiiint$	
Upright	$\int$	$\iint$	$\iiint$	$\iiiint$	$\oint$	$\oiint$	$\oiiint$	

Command	`\intclockwise`	`\awint`	`\varointclockwise`	`\ointctrclockwise`
Default	$\intclockwise$	$\awint$	$\varointclockwise$	$\ointctrclockwise$
Upright	$\intclockwise$	$\awint$	$\varointclockwise$	$\ointctrclockwise$

3.4 Other font features

To get oldstyle numbers in Maths, the feature `+onum` is available:

```
\setmathfont{XCharter-Math.otf}[Numbers=OldStyle]  or
\usepackage[Style={fulloldstyle]{xcharter-otf}
```

0123456789, **0123456789**

3.5 Standard LaTeX math commands

All standard LaTeX math commands, all `amssymb` commands and all `latexsym` commands are supported by XCharter-Math, for some of them loading `xcharter-otf.sty` is required.

Various wide acccents are also supported:

☞ `\widehat` and `\widetilde`

$$\hat{x}\ \widehat{xx}\ \widehat{xxx}\ \widehat{xxxx}\ \widehat{xxxxx}\ \widehat{xxxxxx}\ \widehat{xxxxxxx}\ \tilde{x}\ \widetilde{xx}\ \widetilde{xxx}\ \widetilde{xxxx}\ \widetilde{xxxxx}\ \widetilde{xxxxxx}\ \widetilde{xxxxxxx}$$

[5]These `Style` aliases are provided by `xcharter-otf.sty`.

☞ \overline and \underline

$$\overline{x} \quad \overline{xy} \quad \overline{xyz} \quad \overline{A\cup B} \quad \overline{A\cup(B\cap C)\cup D} \quad \underline{m+n+p}$$

☞ \wideoverbar, \widecheck and \widebreve

$$\overline{x} \quad \overline{xy} \quad \overline{xyz} \quad \breve{x} \quad \breve{xxxx} \quad \breve{xxxxxx} \quad \breve{x} \quad \breve{xxxx} \quad \breve{xxxxxx}$$

☞ \overparen and \underparen

$$\overparen{x} \quad \overparen{xy} \quad \overparen{xyz} \quad \overparen{\mathring{A\cup B}} \quad \overparen{A\cup(B\cap C)\cup D} \quad \overparen{x+y}^{2} \quad \overparen{a+b+\ldots+z}^{26}$$

$$\underparen{x} \quad \underparen{xz} \quad \underparen{xyz} \quad \underparen{x+z}_{2} \quad \underparen{a+b+\ldots+z}_{26}$$

☞ \overbrace and \underbrace

$$\overbrace{a} \quad \overbrace{ab} \quad \overbrace{abc} \quad \overbrace{abcd} \quad \overbrace{abcde} \quad \overbrace{a+b+c}^{3} \quad \overbrace{a+b+\ldots+z}^{26}$$

$$\underbrace{a} \quad \underbrace{ab} \quad \underbrace{abc} \quad \underbrace{abcd} \quad \underbrace{abcde} \quad \underbrace{a+b+c}_{3} \quad \underbrace{a+b+\ldots+z}_{26}$$

☞ \overbracket and \underbracket

$$\overbracket{a} \quad \overbracket{ab} \quad \overbracket{abc} \quad \overbracket{abcd} \quad \overbracket{abcde} \quad \overbracket{a+b+c}^{3} \quad \overbracket{a+b+\ldots+z}^{26}$$

$$\underbracket{a} \quad \underbracket{ab} \quad \underbracket{abc} \quad \underbracket{abcd} \quad \underbracket{abcde} \quad \underbracket{a+b+c}_{3} \quad \underbracket{a+b+\ldots+z}_{26}$$

☞ \overrightarrow and \overleftarrow

$$\overrightarrow{v} \quad \overrightarrow{M} \quad \overrightarrow{vv} \quad \overrightarrow{AB} \quad \overrightarrow{ABC} \quad \overrightarrow{ABCD} \quad \overrightarrow{ABCDEFGH}.$$

$$\overleftarrow{v} \quad \overleftarrow{M} \quad \overleftarrow{vv} \quad \overleftarrow{AB} \quad \overleftarrow{ABC} \quad \overleftarrow{ABCD} \quad \overleftarrow{ABCDEFGH}$$

☞ Finally \widearc and \overrightarc (loading xcharter-otf.sty is required)

$$\overparen{AMB} \quad \overrightarc{AMB}$$

3.6 Mathematical alphabets

- ☞ All Latin and Greek characters are available in italic, upright, bold and bold italic via the `\symit{}`, `\symup{}`, `\symbf{}` and `\symbfit{}` commands.
- ☞ Calligraphic alphabet (`\symscr` or `\symcal` or `\mathcal` command), uppercase:
  $\mathcal{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$
- ☞ Blackboard-bold alphabet (`\symbb` or `\mathbb` command) uppercase only, plus `\Bbbone` and `\Bbbk`:
  $\mathbb{ABCDEFGHIJKLMNOPQRSTUVWXYZ}, \Bbbone, \Bbbk$
- ☞ Fraktur alphabet is borrowed from Latin Modern,
  $\mathfrak{ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz}$
  but this can overwritten, i.e.

```
\setmathfont{Asana-Math.otf}[range=frak,Scale=MatchUppercase]
$\symfrak{ABCDEFGHIJKL...XYZ abcdefghijkl...xyz}$
```

  $\mathfrak{ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz}$
- ☞ Sans serif alphabet is borrowed from Latin Modern,
  ABCDEFGHIJKLMabcdefghijk $\mathit{NOPQRSTUVWXYZmnopqrstuvwxyz}$
  but it can be borrowed from another Math font, i.e.

```
\setmathfont{STIXTwoMath-Regular.otf}[range={sfup,sfit},
                                      Scale=MatchUppercase]
$\symsfup{ABCD...klm}\quad\symsfit{NOPQ...xyz}$
```

  ABCDEFGHIJKLMabcdefghijklm $\mathit{NOPQRSTUVWXYZnopqrstuvwxyz}$
- ☞ Typewriter alphabet is borrowed from Latin Modern,
  `ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz`
  but it can be borrowed from another Math font, i.e.

```
\setmathfont{STIXTwoMath-Regular.otf}[range=tt,
                                      Scale=MatchUppercase]
$\symtt{ABCDE...XYZ abcde...xyz}$
```

  `ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz`

Like Latin Modern, XCharter Math provides only four lowercase latin letters in script (or calligraphic) shape: e, g, ℓ, o (`\mscre`, `\mscrg`, `\ell`, `\mscro`).

All others (range `"1D4B6` to `"1D4CF`) have to be borrowed from another Math font if needed, i.e.

```
\setmathfont{LibertinusMath-Regular.otf}%
           [range="1D4B6-"1D4CF, Scale=MatchLowercase]
```

3.7 Missing symbols

XCharter-Math does not aim at being as complete as `STIXTwoMath-Regular` or `Cambria`, the current glyph coverage compares with TeXGyre Math fonts. In case some symbols do not show up in the output file, you will see warnings in the `.log` file, for instance:

`Missing character: There is no ⇒ (U+2964) in font XCharterMath`

Borrowing them from a more complete font, say `Asana-Math`, is a possible workaround:

```
\setmathfont{Asana-Math.otf}[range={"2964},Scale=1.02]
```

scaling is possible, multiple character ranges are separated with commas:

```
\setmathfont{Asana-Math.otf}[range={"294A-"2951,"2964,"2ABB-"2ABE}]
```

Let's mention `albatross`, a useful tool to find out the list of fonts providing a given glyph: f.i. type in a terminal "`albatross U+2964`", see the manpage or `albatross-manual.pdf`.

4 Acknowledgements

Many thanks to Michael Sharpe for providing the XCharter text fonts in OpenType format.

I am grateful to George Williams and his co-workers for providing and maintaining FontForge and to Ulrik Vieth for his illuminating paper published in TUGboat 2009 Volume 30 about Open Type Math.