Two features of \texttt{mathastext}:
extended scope of the math alphabets and added italic corrections

The package makes ! , ? , : ; + – = ( ) [ ] < > { }, the asterisk *, and ./\# $ % & obey the math alphabet commands (this is the maximal list, some characters may have been excluded by the corresponding package options). For the characters listed first the mechanism involves a ‘mathematical activation’.

As this process may create incompatibilities, it will be put into action for ! , ? , : ; + – = ( ) [ ] < > only if the user makes use of the package command \MTnonlettersobeymathxx (and the braces ne-
cessitate \MTexplicitbracesobeymathxx).

It could be that one such character has been made ‘active’ in the entire document by some other package, typically a language definition file for the babel system. Here for example we have used babel with the french option, which makes the high punctuation characters!?: active throughout the document (extra spacing is put in front of the character when used in text; no change in math but perhaps for other languages and characters this could happen, it is up to the language definition file to decide).

When \texttt{mathastext} detects that a character it wants to ‘mathematically activate’ is already ‘ac-
tive’, it does not go further except if it seems that the activation was done by Babel. If the activation was done by Babel, then \texttt{mathastext} replaces the expansion of the active character in math mode by what is necessary to achieve its goal. It does not additionally mathematically activate the character; rather it makes sure that the character is \texttt{not} mathematically active. In the present document the colon was made mathematically active by \texttt{mathtools} but this was already canceled in the preamble by \texttt{mathastext} as it was loaded later. And it is better so, because the combination babel (with option frenchb)+mathtools (with centercolon) makes $:$ create an infinite loop!

But even if someone had mathematically activated the colon after the preamble, or after the loading of \texttt{mathastext}, this would be canceled again automatically for each inline or displayed mathematical formula (if the user does \MTnonlettersobeymathxx).

The conclusion with \MTnonlettersobeymathxx is: if some package has tried to make the character mathematically active, this will be overruled by \texttt{mathastext}; if some package has made the character globally active, then the package wins except if it is Babel, as \texttt{mathastext} may in the latter case safely modify the action in math mode (paying attention to the fact that the character should be usable in \texttt{label} and \texttt{ref} in and outside of math mode).

The displayed equations next illustrate the extended scope of the math alphabets which now apply to =, –, , , , (, ) but not to the large delimiters of course. Furthermore, for testing purposes the equations were labeled using such characters, for example the last one has label \texttt{eq=7}, to check that the mathematical activation of = does not cause problems with \texttt{\label/\ref}.

\begin{equation}
\left\{ \left( \left[ \sin(a) + \cos(b) - \log(c) = \sec(d) \right] \right) \right\}
\end{equation}

\texttt{mathnormalbold : \mathbb{1}}

\begin{equation}
\left\{ \left( \left[ \sin(a) + \cos(b) - \log(c) = \sec(d) \right] \right) \right\}
\end{equation}

\texttt{mathrm : \mathbb{1}}

\begin{equation}
\left\{ \left( \left[ \sin(a) + \cos(b) - \log(c) = \sec(d) \right] \right) \right\}
\end{equation}

1
\[
\mathbf{???}
\]

\[
\mathit{???}
\]

\[
\mathtt{???}
\]

\[
\mathsf{???}
\]

Equations above are numbered 1, 2, 3, 4, and 5, 6, and 7.

\[
\begin{align*}
a \cdot b & \quad a \cdot b \\
a \cdot b & \quad a \cdot b \\
a \cdot b & \quad a \cdot b \\
a \cdot b & \quad a \cdot b \\
a \cdot b & \quad a \cdot b \\
a \cdot b & \quad a \cdot b \\
\end{align*}
\]

\[
\begin{align*}
< x, y > & \quad < x, y > \\
< x, y > & \quad < x, y > \\
< x, y > & \quad < x, y > \\
< x, y > & \quad < x, y > \\
\end{align*}
\]

\[
\begin{align*}
\langle x, y \rangle & \quad \langle x, y \rangle \\
\langle x, y \rangle & \quad \langle x, y \rangle \\
\langle x, y \rangle & \quad \langle x, y \rangle \\
\langle x, y \rangle & \quad \langle x, y \rangle \\
\end{align*}
\]

\[
\begin{align*}
\{ a, b \} & \quad \{ a, b \} \\
\{ a, b \} & \quad \{ a, b \} \\
\{ a, b \} & \quad \{ a, b \} \\
\{ a, b \} & \quad \{ a, b \} \\
\end{align*}
\]

The question mark has been made active by \texttt{babel+frenchb.mathastext} has imposed in math mode its ways (now \$\texttt{mathbf{???}}\$ gives ???). As the extra spacing is added by \texttt{frenchb} only in text, we had to use the math alphabet to check that indeed \texttt{mathastext} overruled Babel.

To double-check we will now make \texttt{?} mathematically active : \texttt{\mathcode'="8000}. This is a sure cause for disaster normally with Babel (don’t do this at home without \texttt{mathastext}!). But here with \$??\$ no bad surprise (infinite loop !) awaits us : just ?.

Let’s take some other character, for example the opening parenthesis, and make it catcode active : \texttt{\catcode'='\active \def \{X}. Let’s try the input ( and \$\{$. This gives X and X. We
see that mathastext does not attempt to modify the definition of the active character, as this activation was not done via \texttt{babel} services. We now revert the parenthesis to catcode other (but maintain the definition of its active version), and then make it mathematically active using the command \texttt{\textbackslash def \{mathcode'}(=\texttt{8000}). If we try $((($ we see that the parenthesis is not converted into an $X$ character. The mathematically active character was overruled by \texttt{mathastext}.

Issuing \texttt{\textbackslash MTnonlettersdonotobeymathxx} we do get the $X$'s from the input $((($: XXX

This shows that \texttt{mathastext} now does not modify in math mode the non-letter $(X).

We defined in the preamble of the document a \texttt{mathastext-enhanced math} version (named \texttt{upright}) having the Latin letters upright in math mode. Let’s switch to it:

\texttt{\textbackslash MTversion\{upright\}}

With a font which is neither italic nor slanted, mathastext automatically inserts italic corrections for better positioning of the subscript: $f_i^i$ gives $f_i^i$. After \texttt{\textbackslash MTnoicinmath} which turns off this feature, the same input gives $f_i^i$, which is different.\footnote{last time I tried, this only worked with PDFLaTeX, not with LuaLaTeX or XeLaTeX.}

Again with italic corrections on (\texttt{\textbackslash MTicinmath}) $f_{abc}^{def}$ gives $f_{abc}^{def}$, and here is another one: $f_{abc}^{def}$. Without italic corrections: $f_{abc}^{def}$, and respectively $f_{abc}^{def}$.

\texttt{mathastext} does not add these italic corrections inside arguments of math alphabets, as this would prevent the formation of ligatures: ff, ff, ff, ff, ff, ff (no ligature in teletype) and ff.\footnote{Prior to 1.3i, italic corrections were added to the \texttt{\textbackslash mathnormal} arguments.}