

The MATHDOTS package*

Dan Luecking

2014/06/11

Abstract

This package provides vertical dots and diagonal dots in math, slanting in either direction. It should work equally well in L^AT_EX or plain T_EX. It improves on the default definitions of plain T_EX and L^AT_EX. Similar improvements are provided for the triple and quadruple dot accents of AMST_EX/AMSL^AT_EX.

1 Introduction

Running L^AT_EX or T_EX on `mathdots.ins` generates `mathdots.sty`, a L^AT_EX package, and `mathdots.tex` a plain T_EX input file. If one puts `\input mathdots.sty` in a plain T_EX file, it will cause `mathdots.tex` to be input. This is for backward compatibility.

Unlike the commands `\ddots` and `\vdots` provided by L^AT_EX, the versions defined by MATHDOTS change size with L^AT_EX size changing commands, as well as in `{sub[sub] | super[super]}scripts`. After MATHDOTS is loaded, `{\large\ddots}` will be larger and `{\small\ddots}` will be smaller than `\ddots` at normal size. Moreover, `$2^{2^{\ddots}}$` will also produce a smaller symbol. This last example is very unlikely to be needed, but the version with the opposite slant is very common (and is provided by MATHDOTS).

The AMSMATH accents `\dddots` and `\ddddots` have similar problems that MATHDOTS attempts to correct when AMSMATH is loaded.

See Table 1 for examples of the MATHDOTS commands at different sizes and in exponents. See Table 2 for comparison, showing the results without MATHDOTS.

`\ddots` The default L^AT_EX (and plain T_EX) definition changes the size of the dots, and the horizontal dimensions, but not the vertical dimensions, giving them different slants at different sizes. Also, using the default version in `'scripts` produces symbols identical to those produced elsewhere. Finally, the default takes the dots from the current text font, whereas the MATHDOTS package takes them from the current math symbol font, so they should match other dots in math mode. This is only really a concern in 17pt sizes or larger where `cmr17` is scaled to get text dots, but `cmsy12` is scaled to get math dots.

`\iddots` MATHDOTS supplies also `\iddots`, with behavior similar to `\ddots`, except it produces the other diagonal. This command is not provided in basic T_EX or L^AT_EX.

`\vdots` `\vdots` produces vertical dots (as in standard T_EX or L^AT_EX), but again, the

*This file has version number v0.9, last revised 2014/06/11.

With MATHDOTS:

Command	Large	normal	scriptsize	in exponents	
<code>\ddots</code>	$\overset{\cdot}{\cdot}$	\ddots	$\scriptstyle\ddots$	2^{\ddots}	$2^{2^{\ddots}}$
<code>\vdots</code>	$\overset{\cdot}{\cdot}$	\vdots	$\scriptstyle\vdots$	2^{\vdots}	$2^{2^{\vdots}}$
<code>\iddots</code>	$\overset{\cdot}{\cdot}$	$\overset{\cdot}{\cdot}$	$\overset{\cdot}{\cdot}$	$2^{\overset{\cdot}{\cdot}}$	$2^{2^{\overset{\cdot}{\cdot}}}$
<code>\dddot{X}</code>	$\overset{\cdot\cdot\cdot}{X}$	$\overset{\cdot\cdot}{X}$	$\overset{\cdot}{X}$	$2^{\overset{\cdot\cdot}{X}}$	$2^{2^{\overset{\cdot\cdot}{X}}}$
<code>\ddddot{X}</code>	$\overset{\cdot\cdot\cdot\cdot}{X}$	$\overset{\cdot\cdot}{X}$	$\overset{\cdot}{X}$	$2^{\overset{\cdot\cdot}{X}}$	$2^{2^{\overset{\cdot\cdot}{X}}}$

Table 1: Dots at different sizes and in exponents.

Without MATHDOTS:

Command	Large	normal	scriptsize	in exponents	
<code>\ddots</code>	$\overset{\cdot}{\cdot}$	\ddots	$\scriptstyle\ddots$	$2^{\overset{\cdot}{\cdot}}$	$2^{2^{\overset{\cdot}{\cdot}}}$
<code>\vdots</code>	$\overset{\cdot}{\cdot}$	\vdots	$\scriptstyle\vdots$	$2^{\overset{\cdot}{\cdot}}$	$2^{2^{\overset{\cdot}{\cdot}}}$
<code>\dddot{X}</code>	$\overset{\cdot\cdot\cdot}{X}$	$\overset{\cdot\cdot}{X}$	$\overset{\cdot}{X}$	$2^{\overset{\cdot\cdot}{X}}$	$2^{2^{\overset{\cdot\cdot}{X}}}$
<code>\ddddot{X}</code>	$\overset{\cdot\cdot\cdot\cdot}{X}$	$\overset{\cdot\cdot}{X}$	$\overset{\cdot}{X}$	$2^{\overset{\cdot\cdot}{X}}$	$2^{2^{\overset{\cdot\cdot}{X}}}$

Table 2: Dots without MATHDOTS.

MATHDOTS version will change size with size changing commands.

`\fixeddots` The old behavior is stored in `\fixeddots` and `\fixedvdots`. If you wish
`\fixedvdots` `\iddots` to behave similarly then use `\fixediddots`.
`\fixediddots` The `\fixed...` versions try to maintain the (awful) default behavior, which
is to let the horizontal dimensions change size, as well as the dots themselves, but
hard code unchanging vertical dimensions. The ‘fixed’ part of the name does not
mean ‘corrected’, but rather ‘unchanging’ or ‘frozen’.

`\originaldots` On the chance that some other package changes the behavior before MATHDOTS
`\originalvdots` is loaded, use `\originaldots`, `\originalvdots`, and `\originaliddots` to obtain
`\originaliddots` the versions in force before MATHDOTS is loaded.

`\dddot` If the AMSMATH package is detected in L^AT_EX, the commands `\dddot` and
`\ddddot` `\ddddot` are modified to get the same size changing behavior in ‘scripts’. They
also correct a positioning bug in the AMSMATH code. Compare the old and new
versions and the 2-dot accent (from basic L^AT_EX): \ddot{X} , $\ddot{\ddot{X}}$, $\ddot{\ddot{\ddot{X}}}$. I have also taken the
liberty of changing the spacing between dots to match that in the 2-dot accent.

`\originaldddot` Some of the changes to `\dddot` and `\ddddot` require the command `\text`,
`\originalddddot` and so is only applied if AMSMATH has been loaded *before* MATHDOTS. The old
amsmath commands are saved in `\originaldddot` and `\originalddddot`. Similar
comments apply to plain T_EX with regard to `amstex.tex`.

`\MDoprekern` Different math fonts have different metrics for the `\dot` and `\ddot` accents. If
`\MDodotkern` you use a math font family other than the CM fonts, you can adjust the positioning
and spacing of the the dots in `\dddot` and `\ddddot` to better match the spacing
`\MDopostkern` in `\ddot` with the following length commands: `\MDoprekern`, `\MDodotkern` and
`\MDopostkern`. These adjust the space before the first dot, between the dots and
after the last dot, respectively. They are in units of `\mu`, which is 1/3 the size of the
space made by `\,`. Reducing the first and/or increasing the last one cause the dots
to be shifted left. Reducing the middle one causes the dots to be closer together.
As there is usually a bit of white on either side of the dot character, these usually
have to be given negative values (so “reduce” means “make more negative”). The
defaults for the CM fonts are accomplished by the commands

```

\MDoprekern=0mu
\MDodotkern=-1.3mu
\MDopostkern=-1mu

```

I have found that the MATHPTMX package (Times fonts in text and math) benefits
from changing these via the following commands

```

\MDoprekern=-1mu
\MDodotkern=-2mu
\MDopostkern=0mu

```

`\MDoprekern`, `\MDodotkern`, and `\MDopostkern` were made user-accessible com-
mands in version 0.9.

`\MDoddots` In L^AT_EX, all these commands have been made robust as of version 0.6. On
`\MDovdots` the chance that some later loaded package changes things, the MATHDOTS ver-
`\MDoiddots` sions can be obtained with `\MDoddots`, `\MDovdots`, `\MDoiddots`, `\MDodddot` and
`\MDodddot` `\MDoddddot` (as of version 0.9).
`\MDoddddot`

2 The implementation

The following are commands for plain T_EX that prevent double loading, announce the package name, and change/restore the category of @.

```

1 <*tex>
2 \expandafter
3 \ifx\csname MathDots\mathdotsfileversion\endcsname\relax
4 \else
5 \immediate\write16{Mathdots already loaded.}\expandafter\endinput
6 \fi
7 \immediate\write16{Package mathdots, \mathdotsfiledate\space version
8 \mathdotsfileversion.}%
9 \expandafter\edef\csname MathDots\mathdotsfileversion\endcsname{%
10 \catcode'\noexpand\@=\the\catcode'@}%
11 \catcode'\@=11
12 \def\@nameuse#1{\csname#1\endcsname}%
13 </tex>

```

Now various shorthands. For example ‘\MDo@us\MDo@t’ becomes ‘\textstyle’ and ‘\MDo@uf\MDo@ss’ produces ‘\scriptscriptfont0’.

```

14 \def\MDo@us#1{\@nameuse{#1style}}%
15 \def\MDo@uf#1{\@nameuse{#1font}0}%
16 \def\MDo@t{text}\def\MDo@s{script}\def\MDo@ss{scriptscript}%
17 \newdimen\MDo@unit \MDo@unit\p@

```

The following are the main utility macros to implement changes of style. Since most of our symbols are built with boxes, they have to exit math mode and need `\mathchoice` so we know the current style (i.e., size) when we re-enter math mode.

We detect differences in sizes (for example, after `\large`) by examining the appropriate `\fontdimen` of family 0. Instead of the hard coded 1pt, we use 0.1em from this family.

We pass along the current style via `\everymath`.

In `\MDo@palette`, #1 is a command which we feed the current style name. For example, `\MDo@palette\MDo@ddots` will be our definition of `\ddots`. Then `\MDo@ddots` will see the current style as its first argument.

```

18 \def\MDo@changestyle#1{\relax\MDo@unit0.1\fontdimen6\MDo@uf{#1}}%
19 \everymath\expandafter{\the\everymath\MDo@us{#1}}%
20 \def\MDo@dot{\$m@th\ldotp$}%
21 \def\MDo@palette#1{\mathchoice{#1\MDo@t}{#1\MDo@t}{#1\MDo@s}{#1\MDo@ss}}%

```

In the following commands, #1 is the current style (supplied via `\MDo@palette`). Except for the change in the unit used and the dot used, the first two definitions are essentially the same as plain’s `\ddots` and `\vdots`, and the third is the same as the first with the boxes reversed.

```

22 \def\MDo@ddots#1{\MDo@changestyle{#1}}%
23 \mkern1mu\raise7\MDo@unit\vbox{\kern7\MDo@unit\hbox{\MDo@dot}}%
24 \mkern2mu\raise4\MDo@unit\hbox{\MDo@dot}%
25 \mkern2mu\raise \MDo@unit\hbox{\MDo@dot}\mkern1mu}}%
26 \def\MDo@vdots#1{\vbox{\MDo@changestyle{#1}}%

```

```

27 \baselineskip4\MDo@unit\lineskiplimit\z@
28 \kern6\MDo@unit\hbox{\MDo@dot}\hbox{\MDo@dot}\hbox{\MDo@dot}}}%
29 \def\MDo@iddots#1{\MDo@changestyle{#1}%
30 \mkern1mu\raise \MDo@unit\hbox{\MDo@dot}%
31 \mkern2mu\raise4\MDo@unit\hbox{\MDo@dot}%
32 \mkern2mu\raise7\MDo@unit\vbox{\kern7\MDo@unit\hbox{\MDo@dot}}}}}%

\fixedddots      We include current (as of 2014/06/11) LATEX/plainTEX definitions so a user
\fixedvdots      can choose them. LATEX and plainTEX don't have the other diagonal, so we provide
fixediddots      a \fixediddots which is just \fixedddots with the boxes reversed.
\originalvdots   We save the versions at load time so user can use them in case he loads MATH-
\originalddots   DOTS for its other features. Chances are the original \iddots is undefined.
\originaliddots

33 <sty>\DeclareRobustCommand\fixedvdots{%
34 <tex>\def\fixedvdots{%
35 \vbox{\baselineskip4\p@ \lineskiplimit\z@
36 \kern6\p@\hbox{.}\hbox{.}\hbox{.}}}%
37 <sty>\DeclareRobustCommand\fixedddots{%
38 <tex>\def\fixedddots{%
39 \mathinner{\mkern1mu
40 \raise7\p@\vbox{\kern7\p@\hbox{.}}\mkern2mu
41 \raise4\p@\hbox{.}\mkern2mu
42 \raise\p@\hbox{.}\mkern1mu}}}%
43 <sty>\DeclareRobustCommand\fixediddots{%
44 <tex>\def\fixediddots{%
45 \mathinner{\mkern1mu
46 \raise\p@\hbox{.}\mkern2mu
47 \raise4\p@\hbox{.}\mkern2mu
48 \raise7\p@\vbox{\kern7\p@\hbox{.}}\mkern1mu}}}%
49 \let\originalddots=\ddots
50 \let\originalvdots=\vdots
51 \let\originaliddots=\iddots

\ddots           Here are the actual (re)definitions of these three commands. The plainTEX ver-
\vdots           sion emits messages similar to that produced by LATEX's \DeclareRobustCommand.
\iddots          Since \ddots et al. are defined in LATEX with \DeclareRobustCommand,
                 saving the original definition with \let\originalddots\ddots will not work
                 unless the internal command “\ddots␣” is unchanged. Thus we can't use
                 \DeclareRobustCommand\ddots directly because it changes that internal. Instead
                 we define \MDoddots and then do \let\ddots\MDoddots

52 <sty>\DeclareRobustCommand\MDoddots{%
53 <tex>\def\MDoddots{%
54 \mathinner{\MDo@palette\MDo@ddots}}}%
55 <sty>\DeclareRobustCommand\MDoiddots{%
56 <tex>\def\MDoiddots{%
57 \mathinner{\MDo@palette\MDo@iddots}}}%
58 <sty>\DeclareRobustCommand\MDovdots{%
59 <tex>\def\MDovdots{%
60 \mathinner{\MDo@palette\MDo@vdots}}}%
61 <tex>\wlog{Mathdots Info: Redefining \string\ddots.}%
62 <tex>\wlog{Mathdots Info: Redefining \string\vdots.}%

```

```

63 \let\ddots \MDoddots
64 \let\iddots\MDoiddots
65 \let\vdots \MDovdots

```

Now we try to get AMS `\ddot` and `\ddddot` accents to behave nicely.

```

\ddot
\ddddot

MDoprekern
MDodotkern
MDopostkern

\originalddot
\originalddddot

```

We add a `\kern\z@` to prevent T_EX from vertically centering #1. I don't know if the `\text` command is necessary, but it doesn't seem to hurt. I included it in the original version because that one used `text dots` and needed the `text` to change size. Now I use a math symbol (the dot accent) and `\mathpalette` to implement the change. Still, I don't really understand how `\ex@` works in the AMS packages, so I leave it in. For possible fine tuning, I have used two macros that represent the math spacing placed before and after the sequence of dots (`\MDoprekern` and `\MDopostkern`). Also, the actual dot is in the definition of `\MDo@D` where another spacing command follows the dot (`\MDodotkern`). In fact, by redefining `\MDo@D` and changing the spacing, one can define other multiple "dot" accent commands.

We also save the original AMSMATH versions.

```

66 \newbox\MDo@dotsbox
67 \newmuskip\MDoprekern
68 \newmuskip\MDopostkern
69 \newmuskip\MDodotkern
70 \MDoprekern 0mu
71 \MDopostkern -1mu
72 \MDodotkern -1.3 mu
73 \def\MDo@dotsaccent#1{\mathpalette{\MDo@@dotsaccent{#1}}}%
74 \let\originalddot \ddot
75 \let\originalddddot\ddddot
76 \langle*sty\rangle
77 \def\MDo@@dotsaccent#1#2#3{%
78   {\setbox\MDo@dotsbox\hbox{${#2}\mkern\MDoprekern#1\mkern\MDopostkern$}}%
79   \mathop{#3\kern\z@}\limits^{#1}%
80   \text{\vbox to-1.4\ex@{\kern-1.8\ex@\copy\MDo@dotsbox\vss}}}%
81 }%
82 \langleifpackageloaded{amsmath}\rangle%
83 \def\MDo@D{\mathchar"5F\mkern\MDodotkern}%
84 \DeclareRobustCommand\MDoddot {%
85   \MDo@dotsaccent{\MDo@D\MDo@D\MDo@D}}%
86 \DeclareRobustCommand\MDoddidot{%
87   \MDo@dotsaccent{\MDo@D\MDo@D\MDo@D\MDo@D}}%
88 \let\ddot \MDoddot
89 \let\ddddot\MDoddidot
90 \rangle%
91 \langle/sty\rangle

```

For plain `tex + amstex`: `\ex@` is set in `amstex.tex` to `.2326ex`, so its value depends on the value of `ex` when `AMSTEX` is loaded. Unlike `AMSMATH`, it doesn't seem to be recalculated with size changes. Thus, in plain TeX we add a `reset`.

```

92 \langle*tex\rangle
93 \def\MDo@@dotsaccent#1#2#3{%
94 {%

```

```

95 \setbox\MDo@dotsbox\hbox{${#2}\mkern\MDoprekern#1\mkern\MDopostkern$}%
96 \mathop{#3\kern\z@\}\limits^{\text{\ex@.2326ex
97 \vbox to-1.4\ex@{\kern-1.8\ex@\copy\MDo@dotsbox\vss}}}%
98 }}%
99 \ifx\amstexloaded@\relax % true if amstex has been loaded
100 % A backup test in case \amstexloaded@ became \relax by accident:
101 \ifx\ddddot\UndEfInEd
102 \else
103 \def\MDo@D{\mathchar"5F\mkern\MDodotkern}%
104 \wlog{Mathdots Info: Redefining \string\ddddot.}%
105 \wlog{Mathdots Info: Redefining \string\dddot.}%
106 \def\MDodddot {\MDo@dotsaccent{\MDo@D\MDo@D\MDo@D}}%
107 \def\MDoddddots{\MDo@dotsaccent{\MDo@D\MDo@D\MDo@D\MDo@D}}%
108 \let\dddot \MDodddot
109 \let\ddddot\MDoddddots
110 \fi
111 \fi
112 \csname MathDots\mathdotsfileversion\endcsname
113 </tex>
114 </sty | tex>

```

Index

Numbers refer to the page(s) where the corresponding entry is described.

	D		I		\MDovdots 3
\ddddot	3, 6	\iddots	1, 5		
\dddot	3, 6			O	
\ddots	1, 5	\MDodddot	3	\originalddddot ..	3, 6
		\MDoddddots	3	\originaldddot ...	3, 6
		\MDoddots	3	\originalddots ...	3, 5
	F	\MDodotkern	3, 6	\originaliddots ..	3, 5
\fixedddots	3, 5	\MDoiddots	3	\originalvdots ...	3, 5
\fixediddots	3, 5	\MDopostkern	3, 6		
\fixedvdots	3, 5	\MDoprekern	3, 6	V	
				\vdots	1, 5