mma2ltx

Version 1.23

Giuseppe Ghibò
⟨ghibo@galileo.polito.it⟩

June 21, 1995

Mma2ltx is a program which allows a Mathematica user to include the graphics produced by Mathematica in a \LaTeX{} document using own \LaTeX{} fonts and symbols for labels.
1 Introduction

Mathematica has perhaps the best data plotting tool currently available, but its control over mathematical symbols and formulae inside graphics is very poor\(^1\): it is limited to the characters available in the font ‘Symbol’.

\TeX\ instead is very powerful in this subject but has no data plotting capability: it can only include external graphics.

mma2ltx is the “bridge” across the two worlds: it allows to use any \LaTeX\ symbol and font as labels in graphics created by Mathematica.

2 What does mma2ltx do?

Mma2ltx reads a PostScript graphic file generated by the Mathematica command ‘Display’\(^2\) and writes two output files; the first one is a \LaTeX\ file and contains every string of text of the original graphics file in \LaTeX\ form; the second is an EPSF file: it substantially contains the same things of the original PostScript file, except it has been stripped of any string of text.

3 Requirements

In order to include Mathematica graphics processed by mma2ltx into \LaTeX\ documents you need \LaTeX\ (obvious) and the Rokicki’s dvips\(^3\) \dvips\ processor.

Files processed by mma2ltx were tested under \LaTeX\ v2.09 (25 March 1992) and dvips v5.55 (and newer). The graphics files used were created with Mathematica\(^4\) v2.2.

4 Distribution/Disclaimer

mma2ltx is shareware. If you find it useful (or continue using it longer than a week) please consider paying the fee (the easiest way is simply to send the cash in an envelope) of US $15 (US Dollars), or 20 DM (German Marks) to the author (see §14 for the author’s address).

mma2ltx is Copyright © 1994 by Giuseppe Ghibò.

This software is provided “as is” with no explicit or implicit warranty of any kind. You are using it at your own risk.

The author disclaims any liability for damages, including any direct, indirect, incidental, special, exemplary, or consequential damages arising in any way out of the use of this software, even if advised of the possibility of such damage.

---

\(^1\)\textit{Mathematica} supports output in \TeX\ form, but this feature is only for formulae and doesn’t regard the graphics.

\(^2\)The ‘\texttt{Display}’ command saves a raw PostScript representation of a graphic in a file.

\(^3\)\texttt{dvips} is a converter from \dvips to PostScript files. It was developed by T. Rokicki and is available via anonymous FTP from labrea.stanford.edu or in any CTAN site.

\(^4\)\textit{Mathematica} is Copyright ©1988, 1995 by Wolfram Research, Inc.
This software may be freely distributed and copied as long as the following conditions are acknowledged:

- All parts of the program and the documentation must be left intact in any ways.
- The distribution of single parts is not allowed. The repacking of this distribution with other packers/archivers is, however, allowed.

5 Using \texttt{mma2ltx}

To use \texttt{mma2ltx}, just type

\begin{verbatim}
C:\> mma2ltx mypic.ps
\end{verbatim}

where \texttt{mypic.ps} is the output of the \textit{Mathematica}'s primitive 'Display'. \texttt{mma2ltx} supports many options, as described in the next section.

6 Command Line Options

Options are specified on the command line using a dash ‘\texttt{-}’ followed by a letter. No spaces are allowed between the ‘\texttt{-}’ and the letter. The letter may be followed by an argument. Spaces between the letter and the argument are allowed instead.

Here follows a description of the options supported by \texttt{mma2ltx}. Facultative arguments are indicated in the ‘\texttt{Template}’ enclosed between square brackets \([\ldots]\).

Note that every string in the command line which is not an option argument is taken as an input file.

6.1 Option \texttt{-?}

Option \texttt{-?} (\texttt{-"?"} on Unix) shows the following help messages:

\texttt{mma2ltx v1.23 - Copyright (C) 1994, 1995 by Giuseppe Ghibi'}

Usage: \texttt{mma2ltx [<options>] <filename(s)> [<options>]}  
Where \texttt{<options>} is one or more of:

\begin{itemize}
  \item \texttt{-?} Show these messages
  \item \texttt{-d} Don't keep the aspect ratio
  \item \texttt{-n} Deactivate automatic \$...\$ enclosing
  \item \texttt{-b} Enclose every string into a white box (default = transpar. box)
  \item \texttt{-p[<str>]} Include the \textit{Mathematica} PostScript prologue in the \texttt{.EPS} file
  \item \texttt{-h[<dimen>]} Set picture height to \texttt{<dimen>} (default = 100bp)
  \item \texttt{-w[<dimen>]} Set picture width to \texttt{<dimen>} (default = 161bp)
  \item \texttt{-f[<dimen>]} Add an \texttt{box} to the picture (\texttt{\fboxsep=}<dimen>)
  \item \texttt{-u<unit>} Set all dimensions in the unit \texttt{<unit>}
\end{itemize}
-s<cmd>  Set the font size with the TeX command <cmd>
-o<str>  Output filename
-e<str>  Change only MMA labels which begin with <str>
-c(sx, sy)=(newsx, newsy)<(dimen>,<dimen>) (change alignment)
-a[num]:<num>:<num>] Draw arrows on x and y axes

<dimen> = a number followed by one of TeX’s unit (e.g. 10.3cm)
<num> = a number (e.g. 0.0125)
<cmd> = a TeX’s unit (e.g. cm)
<str> = a TeX command without the backslash ‘\’

Example:
mma2ltx -sfootnotesize -w5in pic1.ps pic2.ps
processes the files ‘pic1.ps’ and ‘pic2.ps’. The width of the pictures
will be 5 inch and \footnotesize will be used as LaTeX command to
set the font size.

6.2 Options -w and -h

Template: -w<dimen>  
-h<dimen>

Options -w and -h must be used to specify respectively the width and the height of the
picture. The argument <dimen> is a number followed by one of TeX’s unit (i.e. one of mm, cm, pt, bp, pc, in, dd, cc, sp). For example, ‘-w10.3cm’ specifies a 10.3cm wide picture.
Note that ‘-w 10.3cm’, ‘-w=10.3cm’ and ‘-w:10.3cm’ are accepted too, but ‘-w10.3 cm’ isn’t accepted (note the space after the number 10). Note also that we may specify only one of ‘-w’ or ‘-h’: the other dimension is calculated to keep the Mathematica aspect ratio. If either the
width and the height are specified, the picture will have (approximately) those dimensions,
but the inside graphic will have dimensions such to fit one of height or width, according to the
aspect ratio. For instance, specifying on the command line ‘-w10cm -h10cm’ and the aspect ratio\footnote{The aspect ratio is height/width in scaled coordinates (i.e., from 0 to 1).} is 0.62 then the picture will be 10cm × 10cm large (this is the dimension “visible”
to \LaTeX), but the inside graphic will be 10cm wide and 6.2cm high. If we have instead
‘-w10cm -h3cm’, the picture will be 10cm × 3cm large but the inside graphic will be just
4.84cm wide and 3cm high. If we don’t want to keep the Mathematica aspect ratio we must
use the -d option. Default width is 161 bp; default height is 100 bp.

6.3 Option -d

Suppress the aspect ratio keeping.
6.4 Option \texttt{-n}

By default \texttt{mma2ltx} encloses every string grabbed from the \textit{Mathematica} PostScript file into a \$...\$ pair. Specifying the \texttt{-n} option on the command line, this behaviour will be disabled.

6.5 Option \texttt{-b}

By default every string is placed on the graphic as if it was enclosed in a transparent box. Using this option every string will be no longer “transparent”, but rather enclosed in a white box having the same size (see the string “some text” shown in Fig. 4 for the behaviour of this option).

6.6 Option \texttt{-o}

\textbf{Template:} \texttt{-o\{filename\}}

Specify the output filename. By default \texttt{mma2ltx} uses as output names the names of the input files stripped of the extension to which append the proper file extension (i.e., \texttt{.tex} for the \LaTeX file and \texttt{.eps} for the EPSF file). The \texttt{-o} option allows you to specify a different name for the EPSF PostScript output file. In this case the name of the \LaTeX file will be \texttt{\{filename\}.tex} anyway.

6.7 Option \texttt{-f}

\textbf{Template:} \texttt{-f [\{dimen\}]} \texttt{

The \texttt{-f} option tells \texttt{mma2ltx} to enclose the whole picture into an \texttt{\fbox}. The optional argument is the amount of \texttt{\fboxsep}; by default \texttt{mma2ltx} assumes \texttt{\fboxsep=0pt}. For instance the command

\texttt{mma2ltx -f5pt -w8cm mypic.ps}

produces a picture 8 cm wide, enclosed into \texttt{\fbox}; from each edge of the box and its contents there are 5 pt.

6.8 Option \texttt{-s}

\textbf{Template:} \texttt{-s\{control sequence\}}

This option specifies a \LaTeX font-size control sequence to change the size of the picture labels. Note that \texttt{mma2ltx} doesn’t check if the \texttt{\{control sequence\}} is a valid \LaTeX command. So be careful.

Generally a \LaTeX font-size command may be one of \texttt{tiny}, \texttt{scriptsize}, \texttt{footnotesize}, \texttt{normalsize}, \texttt{large}, \texttt{Large}, \texttt{LARGE}, \texttt{huge}, \texttt{Huge}. No leading backslash is needed (you must use \texttt{-sfootnotesize} instead of \texttt{-s\footnotesize}).

By default the picture uses the \LaTeX current font size. Note that this command will affect size the of all the strings contained in the picture.
6.9  Option \texttt{-u}

**Template:** \texttt{-u (T\mbox{E}X's unit)}

The \texttt{-u} option specifies the unit of measure of quantities contained in the .tex file generated by \texttt{mma2ltx}. Also the messages shown during \LaTeX{} and \texttt{mma2ltx} processing will use that unit.

6.10  Option \texttt{-p}

**Template:** \texttt{-p \langle prologue file \rangle}

By default the EPSF file produced by \texttt{mma2ltx} doesn't contain the \texttt{Mathematica} Post-Script prologue (i.e. it cannot be printed as is). In fact this prologue is included only once in the final PostScript file produced by \texttt{dvips}.

The \texttt{\langle prologue file \rangle} is an optional argument and allows to specify an alternate \texttt{Mathematica} prologue file (e.g. a newer prologue file). To obtain a prologue file you can use the program \texttt{extpro}. See §10 for further details.

Using this option the EPSF file produced by \texttt{mma2ltx} will contain the \texttt{Mathematica} prologue file. This may be very useful for some \texttt{dvips} viewer with capability to show Post-Script specials.

6.11  Option \texttt{-c}

**Template:** \texttt{-c \langle (s_x, s_y) = (s'_x, s'_y)(dimen_x, dimen_y) \rangle}

The \texttt{-c} option can be used to override a peculiar behaviour of \texttt{Mathematica}'s primitive \texttt{Text}. To place text, \texttt{mma2ltx} normally uses the same conventions of the \texttt{Mathematica}'s primitive \texttt{Text}': reference point \((x, y)\) is realized as follows:

- The text string is placed into a box having the same size.
- An offset \((s_x, s_y)\) in the bounding box coordinates system (see the Fig. 1) determines where the reference point goes.

For instance the offset \((-1, 1)\) means that the box containing the string is placed with the point \((-1, 1)\) at the reference point \((x, y)\), i.e. left and top aligned. If the offset is \((0, 0)\) then the box is centered on the reference point \((x, y)\). Note that we may also have offsets greater than 1. For instance the labelling of the \(x\)-axis is realized (by \texttt{Mathematica}) using a reference point lying on the \(x\)-axis and a bounding box offset of \((0, 2)\). With such offset, different height \(x\)-labels would be placed at different distance from the \(x\)-axis. The \texttt{-c} option by-passes this behaviour. It replaces any label having \((s_x, s_y)\) bb-offset with labels having \((s'_x, s'_y)\) bb-offset further shifted by \((\text{dimen}_x, \text{dimen}_y)\) from the current position (\text{dimen}_x and \text{dimen}_y must be numbers followed by one of T\mbox{E}X's unit).

![Figure 1: Bounding box coordinates.](image)
The following example could make this clear. Consider a graphic having the following labels

\[-\frac{3}{2}, -1, -\frac{1}{2}, 1, \frac{3}{2}\]

under the x-axis. Since labels \(-\frac{3}{2}, -\frac{1}{2}, \frac{1}{2}\) and \(\frac{3}{2}\) are higher than label \(-1\) and \(1\), they are placed lower than the labels \(-1\). Using the option \(-c(0, 2)=(0, 1)(0pt, -5pt)\) every labels will be placed with the top edge of the box that bounds them, at 5 pt from the x-axis, as shown in Fig. 2.

Note that it is possible to specify multiple \(-c\) options on the same command line.

![Without and With \(-c\) correction](image)

Figure 2: Behaviour of the \(-c\) option.

### 6.12 Option \(-e\)

**Template:** \(-e(escape\ sequence)\)

This option tells mma2ltx to convert into \(\LaTeX\) only the labels which begin with the sequence \(escape\ sequence\). For instance with

```
mma2ltx -elatex: mypic.ps
```

only labels which begin with the string “latex:” will be converted. Other labels are left as in the original Mathematica graphic.

### 6.13 Option \(-a\)

**Template:** \(-a[length]:[width]:[inset]\)

Using this option, mma2ltx will add two arrows on the x and y axes of a 2D graphic. \(length\), \(width\) and \(inset\) are optional parameters to specify the arrow size, as shown in figure 3.

For instance

```
mma2ltx -a mypic.ps
```

or

```
mma2ltx -a[length]:[width]:[inset] mypic.ps
```
Figure 3: Examples with the ‘-a’ option.

\mma2ltx -a 0.02:0.01:0.005 mypic.ps

The three parameters \(\langle\text{length}\rangle, \langle\text{width}\rangle, \langle\text{inset}\rangle\) must be specified as a fraction of the picture size (scaled coordinates). Default values are

\[
\begin{align*}
\text{length} &= 0.025 \\
\text{width} &= 0.012 \\
\text{inset} &= 0.006
\end{align*}
\]

7 Including \mma2ltx figures

To include a picture processed by \mma2ltx into a \LaTeX{} document, first you should move the file ‘\texttt{mmatext.sty}’ in your \LaTeX{} input directory and the files ‘\texttt{texmma22.pro}’, ‘\texttt{mmawhite.eps}’ in your \TeX{} PostScript directory\(^6\). Then include the style ‘\texttt{mmatext}’ at the top of your document:

\[
\documentstyle[....,mmatext,...]{...}
\]

and invoke the following macro at the point where you wish to include the picture:

\[
\texttt{\input{mypic}}
\]

where ‘\texttt{mypic}’ is the name of the file processed by \mma2ltx. Note that the command ‘\texttt{\input{mypic}}’ may be invoked within any \LaTeX{} environment, for instance the commands:

\[
\begin{verbatim}
\begin{figure}
  \centering
  \tabcolsep=1cm
  \begin{tabular}{cc}
    \input{mypic1} & \input{mypic2}  \\[2cm]
    \input{mypic3} & \input{mypic4}
  \end{tabular}
\end{tabular}
\end{figure}
\end{verbatim}
\]

\(^6\)This is the directory where you keep the dvips prologue files.
will produce a figure containing four Mathematica pictures. Sometimes, during the \LaTeX\ processing of a \LaTeX\ file containing one or more \mma2ltx\ pictures a message as

Mathematica picture: 'mypic.eps' deltax=0.48502 cm

or a message as

Mathematica picture: 'mypic.eps' deltax=0.48502 cm

or both, could appear. If this happens it means that the picture is wider or higher (by the amount shown) than the picture whose dimensions were established with \mma2ltx.

8 Generating Mathematica pictures

\mma2ltx\ needs a PostScript file. This file must be created from within Mathematica using the primitive 'Display' (see the Mathematica manual for a detailed description of this primitive).

Since \mma2ltx\ just executes a plain translation of every string contained in the Mathematica PostScript file, we may specify a \LaTeX\ control sequence directly from within Mathematica. For instance, to mark ticks with the \LaTeX\ greek letter \('\pi\)', we may use

\begin{verbatim}
Show[g, Ticks -> {{0, {Pi/2, "\\pi/over2"}, {Pi, "\pi"},
{3Pi/2, "3\{\Pi/over2\}"}, {2Pi, "2\\pi"}, Automatic}]
\end{verbatim}

Note, to obtain the backslash '\(' from within Mathematica it must be doubled. So every \LaTeX\ control sequence specified into a Mathematica string must be preceded by a '\"'.

For example to place the formula

\[
f(x) = \sin \frac{1}{x}
\]

at the point \((0.5,0.5)\) of a graphic, left and bottom aligned, we may use the Mathematica command

\begin{verbatim}
Text["f(x)=\sin\frac{1}{x}", {0.5,0.5}, {-1,-1}]
\end{verbatim}

9 Manual adjustment of labels

Sometimes may happens to have two or more labels too much closed each other. In this case a manual adjustment is needed. To do this, edit the file generated by \mma2ltx\ having the extension '.tex'. For instance, let's analyze the file 'mypic.tex':

\begin{verbatim}
% Picture: mypic.eps
% Created by mma2ltx v1.2 - Copyright (C) 1994 Giuseppe Ghib\'o
% Command line: mma2ltx -ucm -w10cm -sfootnotesize mypic.ps
% Creation date: Sat Jul 16 10:40:43 1994
\mmaheaderprotrue
{%
\end{verbatim}
We may observe that every label appears twice in the \texttt{.tex} file: within the command \texttt{\matextfits} and in the command \texttt{\maputtext} (and sometimes in the command \texttt{\maputtext*}). These commands have the following syntax:

\begin{verbatim}
\matextfits(\textit{x,y})(\textit{s_x, s_y})[\textit{(off_x, off_y)}]\{\textit{object}\}
\maputtext(\textit{x,y})(\textit{s_x, s_y})[\textit{(off_x, off_y)}]\{\textit{object}\}
\maputtext*(\textit{x,y})(\textit{s_x, s_y})[\textit{(off_x, off_y)}]\{\textit{object}\}
\end{verbatim}

where \textit{(x,y)} are the coordinates of the reference point. \textit{(s_x, s_y)} are the bounding box coordinates as explained at §6.11. \textit{(off_x, off_y)} is an optional argument and represents the offset in the \textit{x} and \textit{y} direction from the reference point \textit{(x,y)}. \{\textit{object}\} may be any \LaTeX{} object (even another \texttt{mma2\LaTeX} picture). The unit of measure is the one which appears in \texttt{\setpic} command (in this case \texttt{cm}). The command \texttt{\maputtext*} has the same effect of \texttt{\maputtext}, except it encloses the \textit{object} in a white box (see the label “some text” in the 3D graphic shown in Fig. 4).

For instance suppose we want to move right the label ‘0.2’ by 0.5cm, then we must replace the line

\begin{verbatim}
\matextfits(2.152,3.090)(0,2)\{$0.2\$}
\end{verbatim}

with the line

\begin{verbatim}
\matextfits(2.652,3.090)(0,2)\{$0.2\$}
\end{verbatim}
and the line

\texttt{\v{m}matextfits(2.152,3.090)(0,2)(0.5,0){$0.2\$}}

with the line

\texttt{\v{m}maputtext(2.152,3.090)(0,2){$0.2\$}}

Note that this approach is similar to the one explained at §6.11, with the exception that we may control every label, rather than a group of labels.

Now suppose we want to replace the label ‘0.8’ placed under the $x$-axis with the label ‘$x_1$’ to place over the $x$-axis with the low edge of the (invisible) box that bounds this label at 0.2 cm from the $x$-axis. In this case we must replace the lines

\texttt{\v{m}matextfits(7.848,3.090)(0,2){$0.8\$}}

with the lines

\texttt{\v{m}matextfits(7.848,3.090)(0,-1)(0,0.2){$x_1$}}

The result is shown in Fig. 4.
10 The program extpro

The program extpro extracts a prologue file from a *Mathematica* PostScript saved picture. Simply save a graphics within *Mathematica* is PS format, or pass the output through *psfix*. Then use

```
    extpro (mma file) (prologue file)
```

where *(mma file)* is the name of the graphics saved in PS format, and *(prologue file)* is the name of the prologue file to save. E.g.,

```
    extpro mygraph.ps texmma23.pro
```

The obtained prologue file can be used as optional argument for the option `-p`. 
11 Distribution Files

This archive contains the following files:

- `msdos/mma2ltx.exe`: Binary executable for MS/DOS
- `msdos/extpro.exe`: Binary executable for MS/DOS
- `amiga/mma2ltx`: Binary executable for the Amiga
- `amiga/extpro`: Binary executable for the Amiga
- `mma2ltx.c`: C source of `mma2ltx`
- `extpro.c`: C source of `extpro`
- `mmatext.sty`: LaTeX macro file
- `texmma22.lpro`: Mathematica v2.2 PostScript prologue file
- `texmma22.pro`: Squeezed version of `texmma22.lpro`
- `mmawhite.eps`: A PostScript file needed to `mmatext.sty`
- `Makefile`: A Unix Makefile
- `makefile.ami`: Makefile for the Amiga
- `makefile.msc`: Makefile for MS/DOS
- `doc/mma2ltx.dvi`: Documentation of `mma2ltx` (dvi format)
- `doc/mma2ltx.ps`: Documentation of `mma2ltx` (PostScript format at 300 dpi)
- `doc/mma2ltx6.ps`: Documentation of `mma2ltx` (PostScript format at 600 dpi)
- `doc/6mag.eps`: `mma2ltx` EPSF file (needed to `mma2ltx.dvi`)
- `doc/12mag.eps`: `mma2ltx` EPSF file (needed to `mma2ltx.dvi`)
- `doc/12mag_3d.eps`: `mma2ltx` EPSF file (needed to `mma2ltx.dvi`)
- `doc/optc.eps`: `mma2ltx` EPSF file (needed to `mma2ltx.dvi`)
- `doc/arrsamp.eps`: `mma2ltx` EPSF file (needed to `mma2ltx.dvi`)
- `doc/arrparm.l`: `mma2ltx` EPSF file (needed to `mma2ltx.dvi`)
- `mysample.tex`: A sample file
- `mypic.ps`: A sample picture created by Mathematica
- `mypic.tex`: The file `mypic.ps` as processed by `mma2ltx`
- `mypic.eps`: The file `mypic.ps` as processed by `mma2ltx`

12 Limits of `mma2ltx`

Currently aren’t (yet) supported:

- rotated labels.
- multiple graphics (the ones produced with `GraphicsArray`).
- the ‘...->FontForm’ Mathematica parameter.

13 To do list

Here follow future enhancements which are on my list:
• Add support for \LaTeXe.

• Add support for rotated labels.

• Add support for others \verb!dvi! to PostScript processors.

14 Author info

If you have some questions, suggestions, comments, bug report or enhancement requests, please feel free to contact me at one of the following addresses:

• ordinary mail:

  Giuseppe Ghibò  
  via Sestriere, 133  
  I-10090 Cascine Vica – Rivoli (Torino)  
  ITALY

• internet: ghibo@galileo.polito.it

15 Acknowledgements

The author wishes to thanks:

• P. Lepora for his significative suggestions and collaboration.

• P. Boieri for his suggestions and for having intensely tested \verb!mma2ltx!.

16 History

version 1.23

• Added option \verb!-a! to draw arrows on axes of a 2D graphic.

• Fixed a bug in the style \verb!mmatext.sty!.

version 1.22

• Fixed a small bug which caused 'segmentation fault' under Linux.

• Use of perror() instead of strerror() (suggested by Peter Whaite).

version 1.21

• Possibility to use newer prologue files from Mathematica.

• Added support for multiple \verb!-c! options.
• Added option -e (suggested by Holger Danielsson).
• Fixed a bug in the function strtolwr() (reported by Klaus Burkhard).

version 1.2

• Added support to obtain non-transparent objects. Now objects (strings, pictures
  and so on), can be placed to overlap the background graphic, i.e. as if they were
  non-transparent.
• Added PostScript documentation for 600 dpi printers.
• Added binary executable for the Amiga.

version 1.1  First public release.