mma2ltx

Version 1.21

Giuseppe Ghibò
⟨*ghibo@galileo.polito.it*⟩

May 17, 1995

Mma2ltx is a program which allows to a *Mathematica* user to include the graphics produced by *Mathematica* in a LaTeX document using own LaTeX fonts and symbols.

1 Introduction

Mathematica has perhaps the best data plotting tool currently available, but its control over mathematical symbols and formulæ 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] `dvi` processor.

Files processed by *mma2ltx* were tested under LATEX v2.09 (25 March 1992) and `dvips` v5.55 (27 March 1994). 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 (simply cash in an envelope) of or US $15 (US Dollars), or 20 DM (German Marks) to the author (see §14 for the author's address).

[1] *Mathematica* supports output in TEX form, but this feature is only for formulæ and doesn't regard the graphics.

[2] The '`Display`' command saves a raw PostScript representation of a graphic in a file.

[3] `dvips` is a converter from `dvi` 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] *Mathematica* is Copyright ©1988, 1994 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 *mma2ltx*

To use *mma2ltx*, just type

```
C:\> mma2ltx mypic.ps
```

where `mypic.ps` is the output of the *Mathematica*'s primitive '`Display`'. *mma2ltx* supports many options. They are described in the next section.

6 Command Line Options

Options are specified on the command line using a dash '`-`' followed by a letter. No spaces are allowed between the '`-`' 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 *mma2ltx*. Facultative arguments are indicated in the '**Template**' enclosed between square brackets [...].

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

6.1 Option `-?`

Option `-?` (`-"?"` on Unix) shows the following help messages:

```
mma2ltx v1.21 - Copyright (C) 1994, 1995 by Giuseppe Ghibo`

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

  -?  Show these messages
  -d  Don't keep the aspect ratio
  -n  Deactivate automatic $...$ enclosing
  -p  Include the Mathematica PostScript prologue in the .EPS file
  -b  Enclose every string into a white box (default = transpar. box)
  -h<dimen> Set picture height to <dimen> (default = 100bp)
  -w<dimen> Set picture width  to <dimen> (default = 161bp)
  -f<dimen> Add an \fbox to the picture (\fboxsep=<dimen>)
  -u<unit>  Set all dimensions in the unit <unit>
```

```
  -s<cmd>   Set the font size with the TeX command <cmd>
  -o<filename> Output filename
  -e<str> = changes only MMA labels which begin with <str>
  -c(sx,sy)=(newsx,newsy)(<dimen>,<dimen>) (change alignment)

  <dimen> = a number followed by one of TeX's unit (e.g. 10.3cm)
  <unit>  = a TeX's unit (e.g. cm)
  <cmd>   = a TeX command without the backslash '\'
  <str>   = a text string

Example:
 mma2ltx -sfootnotesize -w5in pic1.ps pic2.ps
process 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 10.3 cm 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[5] is 0.62 then the picture will be 10 cm × 10 cm large (this is the dimension "visible" to LATEX), but the inside graphic will be 10 cm wide and 6.2 cm high. If we have instead '`-w10cm -h3cm`', the picture will be 10 cm × 3 cm large but the inside graphic will be just 4.84 cm wide and 3 cm 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 `-n`

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

[5] The aspect ratio is height/width in scaled coordinates (i.e., from 0 to 1).

6.5 Option `-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. 3 for the behaviour of this option).

6.6 Option `-o`

Template: `-o`⟨*filename*⟩

Specify the output filename. By default *mma2ltx* uses as output names the names of the input files stripped of the extension to which append the proper file extension (i.e., '`.tex`' for the LaTeX file and '`.eps`' for the EPSF file). The `-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 ⟨*filename*⟩`.tex` anyway.

6.7 Option `-f`

Template: `-f` [⟨*dimen*⟩]

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

```
mma2ltx -f5pt -w8cm mypic.ps
```

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

6.8 Option `-s`

Template: `-s`⟨*control sequence*⟩

This option specifies a LaTeX font-size control sequence to change the size of the picture labels. Note that *mma2ltx* doesn't check if the ⟨*control sequence*⟩ is a valid LaTeX command. So be careful.

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

Template: `-u`⟨*TeX's unit*⟩

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

6.10 Option -p

Template: `-p`[⟨*prologue file*⟩]

By default the EPSF file produced by *mma2ltx* doesn't contain the *Mathematica* PostScript prologue (i.e. it cannot be printed as is). In fact this prologue is included only once in the final PostScript file produced by `dvips`.

The ⟨*prologue file*⟩ is an optional argument and allows to specify an alternate *Mathematica* prologue file (e.g. a newer prologue file). To obatin a prologue file you can use the program `extpro`. See §10 for further details.

Using this option the EPSF file produced by *mma2ltx* will contains the Mathematica prologue file. This may be very useful for some `dvi` previewer with capability to show PostScript specials.

6.11 Option -c

Template: `-c`$\langle (s_x, s_y) = (s'_x, s'_y)(dimen_x, dimen_y)\rangle$

The `-c` option can be used to override a peculiar behaviour of *Mathematica*'s primitive `Text`. To place text, *mma2ltx* normally uses the same conventions of the *Mathematica*'s primitive '`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).

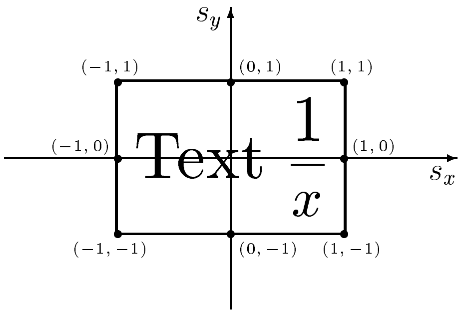


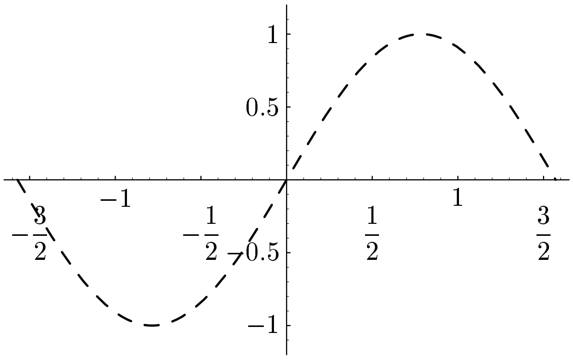
Figure 1: Bounding box coordinates.

Note that we may also have offsets greater than 1. For instance the labelling of the x-axis is realized (by *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 '`-c`' option bypasses this behaviour. It replaces any labels having (s_x, s_y) bb-offset with labels having (s'_x, s'_y) bb-offset further shifted by $(dimen_x, dimen_y)$ from the current position ($dimen_x$ and $dimen_y$ must be numbers followed by one of TeX's unit). The following example could make this clear. Consider a graphic having the following labels

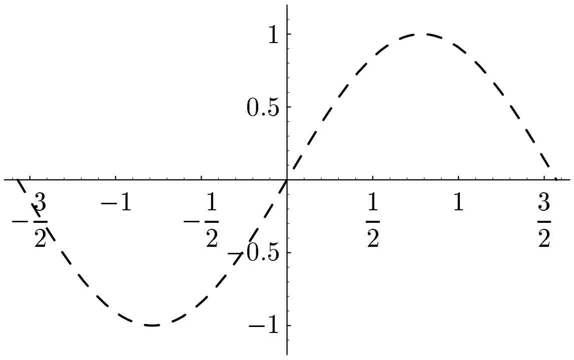
$$-\frac{3}{2} \qquad -1 \qquad -\frac{1}{2} \qquad \frac{1}{2} \qquad 1 \qquad \frac{3}{2}$$

under the x-axis. Since labels '$-\frac{3}{2}$', '$-\frac{1}{2}$', '$\frac{1}{2}$' and 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 ‘`-c`’ correction



With ‘`-c`’ correction

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.

7 Including *mma2ltx* figures

To include a picture processed by *mma2ltx* into a LaTeX documents, first you should move the file ‘`mmatext.sty`’ in your LaTeX input directory and the files ‘`texmma22.pro`’, ‘`mmawhite.eps`’ in your TeX PostScript directory[6]. Then include the style ‘`mmatext`’ at the top of your document:

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

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

```
\input{mypic}
```

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

[6]This is the directory where you keep the `dvips` prologue files.

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

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' deltay=1.28744 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 'π', we may use

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

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

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

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`':

```
% 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
{%
\footnotesize%
\mmasetpic(10.0000,6.1803)[cm]{mypic.eps}
\mmatextfits(2.152,3.090)(0,2){$0.2$}
\mmatextfits(4.051,3.090)(0,2){$0.4$}
\mmatextfits(5.949,3.090)(0,2){$0.6$}
\mmatextfits(7.848,3.090)(0,2){$0.8$}
\mmatextfits(9.747,3.090)(0,2){$1$}
\mmatextfits(0.129,0.157)(1,0){$-1$}
\mmatextfits(0.129,1.623)(1,0){$-0.5$}
\mmatextfits(0.129,4.557)(1,0){$0.5$}
\mmatextfits(0.129,6.024)(1,0){$1$}
\mmatextfits(5.000,4.557)(-1,-1){$f(x)= \sin \frac{1}{x}$}
\begin{mmapicture}
\mmaputtext(2.152,3.090)(0,2){$0.2$}
\mmaputtext(4.051,3.090)(0,2){$0.4$}
\mmaputtext(5.949,3.090)(0,2){$0.6$}
\mmaputtext(7.848,3.090)(0,2){$0.8$}
\mmaputtext(9.747,3.090)(0,2){$1$}
\mmaputtext(0.129,0.157)(1,0){$-1$}
\mmaputtext(0.129,1.623)(1,0){$-0.5$}
\mmaputtext(0.129,4.557)(1,0){$0.5$}
\mmaputtext(0.129,6.024)(1,0){$1$}
\mmaputtext(5.000,4.557)(-1,-1){$f(x)= \sin \frac{1}{x}$}
\end{mmapicture}%
}%
```

We may observe that every label appears twice in the `.tex` file: within the command `\mmatextfits` and in the command `\mmaputtext` (and sometimes in the command `\mmaputtext*`). These commands have the following syntax:

`\mmatextfits`$(x, y)\,(s_x, s_y)\,[(\mathit{off}_x, \mathit{off}_y)]\{\mathit{object}\}$
`\mmaputtext` $(x, y)\,(s_x, s_y)\,[(\mathit{off}_x, \mathit{off}_y)]\{\mathit{object}\}$
`\mmaputtext*`$(x, y)\,(s_x, s_y)\,[(\mathit{off}_x, \mathit{off}_y)]\{\mathit{object}\}$

where (x, y) are the coordinates of the reference point. (s_x, s_y) are the bounding box coordinates as explained at §6.11. $(\mathit{off}_x, \mathit{off}_y)$ is an optional argument and represents the offset in the x and y direction from the reference point (x, y). $\{\mathit{object}\}$ may be any LaTeX object (even another *mma2ltx* picture). The unit of measure is the one which appears in '`\mmasetpic`' command (in this case `cm`). The command '`\mmaputtext*`' has the same effect of '`\mmaputtext`', except it encloses the *object* in a white box (see the text "some text" in the 3D graphic shown in Fig. 3).

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

```
\mmatextfits(2.152,3.090)(0,2){$0.2$}
```

with the line

```
\mmatextfits(2.152,3.090)(0,2)(0.5,0){$0.2$}
```

and the line

```
\mmaputtext(2.152,3.090)(0,2){$0.2$}
```

with the line

```
\mmaputtext(2.152,3.090)(0,2)(0.5,0){$0.2$}
```

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

```
\mmatextfits(7.848,3.090)(0,2){$0.8$}
.
.
.
\mmaputtext(7.848,3.090)(0,2){$0.8$}
```

with the lines

```
\mmatextfits(7.848,3.090)(0,-1)(0,0.2){$x_1$}
.
.
.
\mmaputtext(7.848,3.090)(0,-1)(0,0.2){$x_1$}
```

The result is shown in Fig. 3.

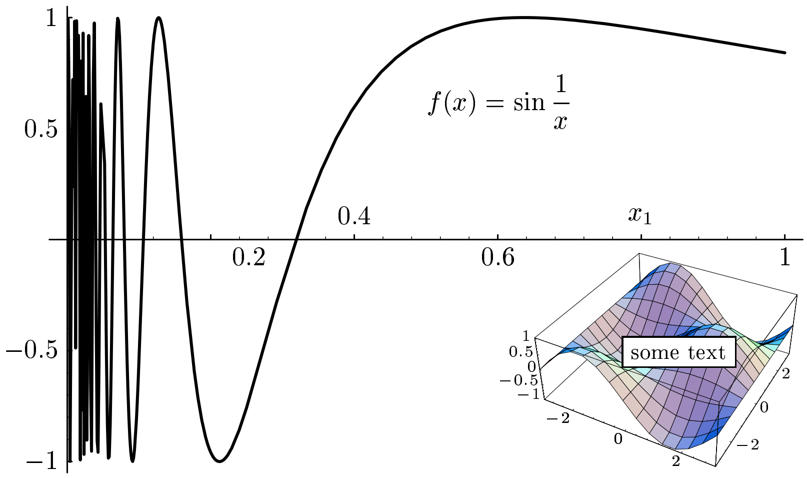


Figure 3: A sample figure.

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

$$\texttt{extpro}\ \langle \textit{mma file} \rangle\ \langle \textit{prologue file} \rangle$$

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` form)
`doc/mma2ltx.ps`	Documentation of *mma2ltx* (PostScript form at 300 dpi)
`doc/mma2ltx6.ps`	Documentation of *mma2ltx* (PostScript form 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`')
`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*
`README`	A short description of *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 LaTeX 2_ε.
- Add support for rotated labels.

- Add support for others `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:**

  Ghibò Giuseppe
  via Sestriere, 133
  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 *mma2ltx*.

16 History

version 1.21

- Possibility to use newer prologue files from *Mathematica*.
- Added support for multiple `-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.