egplot:
Encapsulated gnuplot for LaTeX*

Axel Probst†

c/o
Federal Institute for Materials Research and Testing
Unter den Eichen 87
D-12205 Berlin
Germany

June 25, 1998

Abstract

The egplot package allows to encapsulate gnuplot commands in LaTeX sources. This is very useful for keeping illustrations in sync with the text. It also frees the user from inventing descriptive names for PostScript files. Additionally the package provides commands that enable the user to let gnuplot do calculations and insert the result values into the generated output.

Copying

egplot is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2, or (at your option) any later version.
egplot is distributed in the hope that it will be useful, but *without any warranty*; without even the implied warranty of *merchantability* or *fitness for a particular purpose*. See the GNU General Public License for more details.
You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

*This is `egplot.sty`, version v1.01, date 1998/06/20.

†e-mail: `Axel.Probst@bam.de`

1 Introduction

When adding illustrations to documents, one faces two bookkeeping problems:

1. How to encourage oneself to keep the illustrations in sync with the text, when the document is updated?

2. How to make sure that the illustrations appear on the right spot?

For both problems, the best solution is to encapsulate the figures in the LaTeX source:

1. It is much easier to remember to update an illustration if one doesn't have to switch files in the editor.

2. One does not have to invent illustrative filenames, if the computer keeps track of them.

This concept of integrating the image generating commands into the LaTeX source was implemented for METAFONT by Thorsten Ohl[1] in the EM -package. The egplot package now allows the encapsulation of gnuplot [5] into LaTeX [1, 2, 3]. Although gnuplot provides several output formats that are suitable for the inclusion into LaTeX the egplot package is only intended for use with the Postscript terminal of gnuplot so far.
In addition to the image inclusion commands egplot provides the user with commands to let gnuplot do calculations and include the results into the document. Unfortunately these features are implemented with the UN*X text utils and so they are only usable if these are installed on the system. Since the names for the PostScript and the result values files are built by appending the figure/calculation number and a three letter extension (`.eps` or `.val`) to `\jobname` the user has to choose a `\jobname` that is short enough so that the generated filenames fit into the conventions of certain operating systems.

2 Usage

2.1 Options

`Options` Besides the options of the `graphicx` package the option `german` is recognized by the egplot package.
If `german` is specified the calculated values and the tic labels of the diagrams are changed to show a ',' as decimal point character. The default is a '.'. This feature is also implemented with UN*X text utils and is only available if they are installed on the system.

2.2 Commands and Environments

2.2.1 Miscellaneous

`egpfile` All descriptions that should go into one gnuplot file are placed inside a `egpfile` environment which takes the name of the gnuplot file as an optional argument:

[1] e-mail:`Thorsten.Ohl@Physik.TU-Darmstadt.de`

```
\begin{egpfile}[⟨gnuplot-file⟩]
    ...
\end{egpfile}
```

The default gnuplot-filename is `\jobname.gp`.

`egpcmds` `\egpwrite` Write gnuplot commands to the current file outside of a figure. The `\egpwrite` command is intended for short one line commands.

```
\begin{egpcmds}
    ⟨gnuplot-commands⟩
\end{egpcmds}
```

`\egpprelude` `\egpaddtoprelude` Define and add to the set of commands that are prepended to the top of every gnuplot file. It is intended for the global definition of variables or functions. The default is empty.

2.2.2 Figures

`egp` `egpx` `egpdef` The egp as the egpx environment contains the description of a single figure that will be placed at the location of the environment. The egpdef environment only defines a figure but does not include it into the document. This is useful, because these environments use the verbatim package and can therefore *not* be used as an argument to other macros. The ⟨*name*⟩ that is assigned to the figure is used for later inclusion with the `\egpuse{`⟨*name*⟩`}` command. For the egp and egpx environment the assignment of the ⟨*name*⟩ is optional. The required argument of the egpx environment accepts any set of keys accepted by the `\includegraphics` command of the graphicx package.

```
\begin{egp}[⟨name⟩]
    ⟨gnuplot-commands⟩
\end{egp}

\begin{egpx}[⟨name⟩]{⟨key val list⟩}
    ⟨gnuplot-commands⟩
\end{egpx}

\begin{egpdef}{⟨name⟩}
    ⟨gnuplot-commands⟩
\end{egpdef}
```

`\egpuse` Reuse a previously defined figure. The optional argument of the `\egpuse` command accepts any set of the keys that is accepted by the `\includegraphics` command of the graphicx package.

```
\egpuse[⟨key val list⟩]{⟨name⟩}
```

`\egpfigprelude` `\egpaddtofigprelude` Define and add to a gnuplot prelude that is prepended to the output of every egp, egpx or egpdef environment. The default is:

```
set terminal postscript eps monochrome dashed "Helvetica" 17
```

In fact this is the command where the terminal for the gnuplot-plot command is set. So the user has to take care that (Encapsulated) PostScript output is generated.

\egpfigepilog
\egpaddtofigepilog

Define and add to a gnuplot epilog that is appended to the output of every egp, egpx or egpdef environment. This command can be used for e.g. replotting the figure to the screen or reseting to the defaults after every figure.
The default is:

```
reset
```

2.2.3 Calculating

In addition to the commands and environments to generate and include gnuplot figures the egplot-package provides commands to use gnuplot for the calculation of arbitrary arithmetic expressions. Since the gnuplot-plot command is used for this feature every expression that is accepted by this command is possible. But this may also lead to unexpected results if the expression contains the variable x which is used as the independent variable of the gnuplot-plot command. As stated above (cf. p. 2) the UN*X text utils are used for the implementation and so the calculation commands can only be used on systems where these are installed.

\egpcalc

Let gnuplot calculate the value of a ⟨*gnuplot-expression*⟩. The result is written to a file. The optional argument assigns a name to be used with \egpuseval{⟨*name*⟩}.

```
\egpcalc[⟨name⟩]{⟨gnuplot-expression⟩}
```

\egpuseval

Insert a previously defined calculation result.

\egpshowval

Does the same as the \egpcalc-command but additionally the result is placed in the output at the position of the \egpshowval-command.

\egpassign

The first argument is the name of a ⟨*gnuplot-variable*⟩ or ⟨*gnuplot-user function*⟩ which is assigned the second argument which is a ⟨*gnuplot-expression*⟩. The result is placed in the output as for the \egpshowval command.

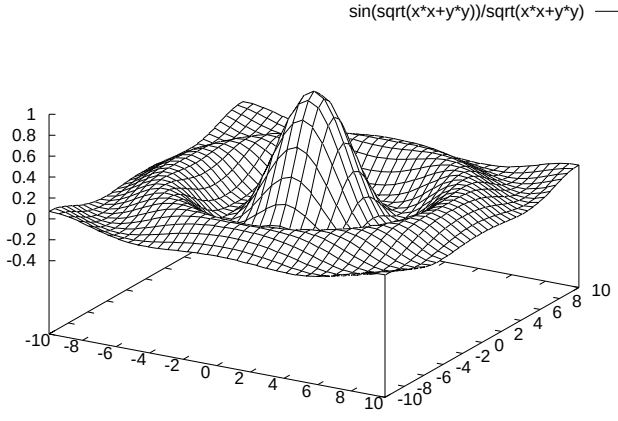
2.3 Procedure

After LaTeX has done it's job for the first time you have to invoke gnuplot on the generated file (default: \jobname.gp). Then another LaTeX run is necessary to include the figures and the results into the output.

2.4 Examples

For a simple example, let's draw the function $f(x) = \sin(\sqrt{x^2+y^2})/\sqrt{x^2+y^2}$.

```
⟨*sample⟩
\begin{egpfile}
\begin{center}
\begin{egpx}[sombrero]{width=0.8\linewidth}
    set hidden3d
    set nogrid
    set samples 1000
    set isosamples 35
    splot [-10:10] [-10:10] sin(sqrt(x*x+y*y))/sqrt(x*x+y*y)
\end{egpx}
\end{center}
```



Additionally we define a figure that will not be shown here but at the place of the appropriate `\egpuse` command.

```
\begin{egpdef}{kleinbottle}
    set hidden3d
    set parametric
    set nokey
    set nogrid
    set noborder
    set noxtics
    set noytics
    set noztics
    set xrange [-10:10]
    set yrange [-10:10]
    set zrange [-3:3]
    set urange [0:2*pi]
    set vrange [0:2*pi]
    set isosamples 39,60
    set view 60,120
    set title "Klein bottle"
    splot  (2*sin(u)*cos(v/2)-sin(2*u)*sin(v/2)+8)*cos(v), \
           (2*sin(u)*cos(v/2)-sin(2*u)*sin(v/2)+8)*sin(v), \
           2*sin(u)*sin(v/2)+sin(2*u)*cos(v/2)
\end{egpdef}
```

Since we have given a name to each diagram, we can now use them with

```
\begin{figure}
  \begin{center}
     \fbox{\egpuse[scale=0.4]{sombrero}}
     \fbox{\egpuse[scale=0.4]{kleinbottle}}
  \caption{Two examples taken from the \GP{} demo}\label{fig:demo}
  \end{center}
\end{figure}
```

and the result is shown in figure 1.
To calculate the value of $f(\pi/4)$ we issue the command
$f(\pi/4) = 0.707107$

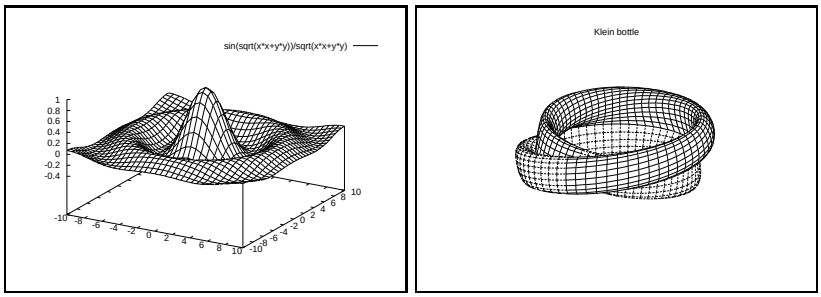


Figure 1: Two examples taken from the gnuplot demo

```
40 $f(\pi/4)=\egpshowval[sin_quarter_pi]{sin(pi/4)}$
```

and get $\frac{\sqrt{2}}{2} = \boxed{0.707107}$.

```
41 and get $\frac{\sqrt{2}}{2}=\fbox{\egpuseval{sin_quarter_pi}}$.
42 \end{egpfile}
```

43 ⟨/sample⟩

3 Acknowledgements

I would like to thank Thorsten Ohl for submitting the EM package to CTAN. By using it as a template I managed it to adapt the idea of integrating the image generating commands into LaTeX for gnuplot. A lot of code of the EM package was reused with only marginal changes. This is also caused by the fact that I am far away from understanding all of the code of EM .

4 Todo

In additon to optimising egplot it would be nice if the features that are provided through the use of UN*X text utils were implemented in TeX/LaTeX. Another interesting feature to implement in following versions of egplot is the possibility to use other output formats provided by gnuplot, especially the `pslatex` and `pstricks` terminals but also the `png` terminal for inclusion into PDF could be useful.

References

[1] Michel Goossens, Sebastian Rahtz, and Frank Mittelbach, *The LaTeX Graphics Companion*, Addison-Wesley, Reading MA, 1997.

[2] Leslie Lamport, *LaTeX — A Documentation Preparation System*, Addison-Wesley, Reading MA, 1985.

[3] Michel Goossens, Frank Mittelbach, and Alexander Samarin, *The LaTeX Companion*, Addison-Wesley, Reading MA, 1994.

[4] Thorsten Ohl, `emp`, available from CTAN (cf. p. 7), in the `macros/latex/contrib/supported/emp` directory.

[5] Thomas Williams and Colin Kelley, *gnuplot*, available from `ftp.dartmouth.edu` in the `/pub/gnuplot` directory.

Distribution

egplot is available by anonymous internet ftp from any of the Comprehensive TEX Archive Network (CTAN) hosts

`ftp.tex.ac.uk`, `ftp.dante.de`

in the directory

`macros/latex/contrib/supported/egplot`

Index

Numbers written in italic refer to the page where the corresponding entry is described, the ones underlined to the code line of the definition, the rest to the code lines where the entry is used.

E

`egp` (environment) . . . *3*
`\egpaddtofigepilog` . *4*
`\egpaddtofigprelude` *3*
`\egpaddtoprelude` . . *3*
`\egpassign` *4*
`\egpcalc` *4*
`egpcmds` (environment) *3*
`egpdef` (environment) *3*
`\egpfigepilog` *4*
`\egpfigprelude` *3*
`egpfile` (environment) *2*
`\egpprelude` *3*
`\egpshowval` *4*, 40
`\egpuse` *3*, 35, 36
`\egpuseval` *4*, 41
`\egpwrite` *3*
`egpx` (environment) . . *3*

environments:
`egpcmds` *3*
`egpdef` *3*
`egpfile` *2*
`egpx` *3*
`egp` *3*

O

`Options` *2*

Change History

v1.00

General: First released version . . 1

v1.01

General: Fixed bug with `german` option and negative tic mark values 1