

TeX-friendly gnuplot terminals

Ethan A Merritt - January 2022

This demo shows various options for combining gnuplot graphics with L^AT_EX processing of text, usually to produce a figure that is to be included in a larger L^AT_EX document. You have a choice among several TeX-friendly gnuplot terminals. Some of these (*pict2e*, *tikz*) hand off both the graphics and the text to be processed by latex or pdflatex. The *cairolatex* terminal uses the cairo graphics library to produce a file containing the graphics in *.png, *.eps or *.pdf format and a second parallel file *.tex containing the accompanying TeX text elements. The *pslatex* and *epslatex* terminals similarly create a pair of files but the graphics file is limited to PostScript (*.ps or *.eps).

Although all of these choices use a TeX or L^AT_EX variant to process the text component of the figure, the details vary. For example, some terminals optionally also use TeX to draw individual point symbols rather than treating them as graphics elements. Some offer a similar option for treating arrows as TeX elements rather than as graphics.

Some terminals, notably *cairolatex* and *tikz*, offer a "standalone" option to produce a self-contained L^AT_EX document rather than a fragment for inclusion in a larger document.

All of the figures shown here were produced with the same gnuplot code, changing only the terminal type. Here are the gnuplot commands

```
# set title "It would be nice if this were converted into a caption!"
set xtics 0.5 nomirror
set tics format "%.1f"
set margins -1,0,7,0
set xrange [-1:1]
set yrange [1:3]
set key notitle invert under reverse Left left spacing 2 samplen 0.7
set arrow 1 filled from graph 0.4, 0.7 to graph 0.6, 0.7
set label 1 at graph 0.5, 0.75 "$k$" center
Title_E = 'EllipticE$(k)=\int_0^{\pi/2} {\sqrt{1-k^2\sin^2\theta}}~d\theta$'
Title_K = 'EllipticK$(k)=\int_0^{\pi/2} {\sqrt{1-k^2\sin^2\theta}}^{-1}~d\theta$'

plot EllipticE(x) lw 3 title Title_E, EllipticK(x) lw 3 title Title_K
```

Things to note

- All text including plot titles is provided in TeX syntax, not in gnuplot's own markup syntax. However formats as in the `set tics` command are applied before output, so TeX never sees the % character.
- Even though we request the same font size and the same overall size for all terminals, it doesn't come out exactly that way.
- All of the latex terminals, except maybe tikz, do a poor job of choosing the margins automatically. It is a good idea to set them explicitly.
- There is no way I know of to automatically transform the text from gnuplot `set title "text"` into a corresponding latex `\caption{text}`. The captions you see here were added explicitly in the outer latex_demo.tex document.

pict2e terminal

This version of the figure is produced by the *pict2e* gnuplot terminal. This terminal was created by Bastian Märkisch based on the original gnuplot *latex* terminal written by David Kotz and its successor terminals *emtex*, *eepic*, and *tpic*. The *pict2e* terminal requires standard latex support packages *pict2e* *xcolor* *graphicx* *amssymb*. To support transparent fill areas when processed through pdflatex it also requires package *transparent*. Both the graphics and the text are processed by latex, or in the case of this demo by pdflatex or xelatex.

```
set terminal pict2e color texarrows font "cmr,10" size 3.5in,2.4in
```

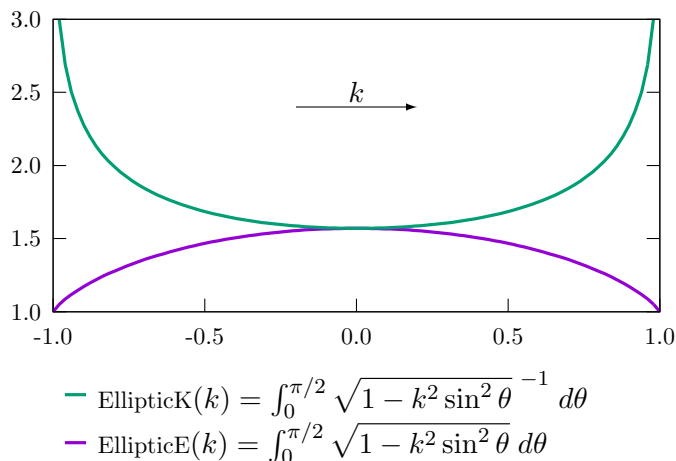


Figure 1: Complete elliptic integrals of the first and second kinds

cairolatex terminal

The *cairolatex* terminal uses the cairo graphics library to produce the graphical parts of the figure as a *.eps *.pdf or *.png file (your choice), and it produces a parallel *.tex file so that latex (or pdflatex, xelatex, ...) can combine the graphics and text into a single figure for later inclusion in a latex document. Generally you would use the `eps` option only when working in a plain latex/dvips environment that is limited to PostScript file inclusion, and use the `png` option only when the figure includes pixel-based images.

```
set terminal cairolatex color pdf font "cmr,10" fontscale 0.7 size 3.5in,2.4in
```

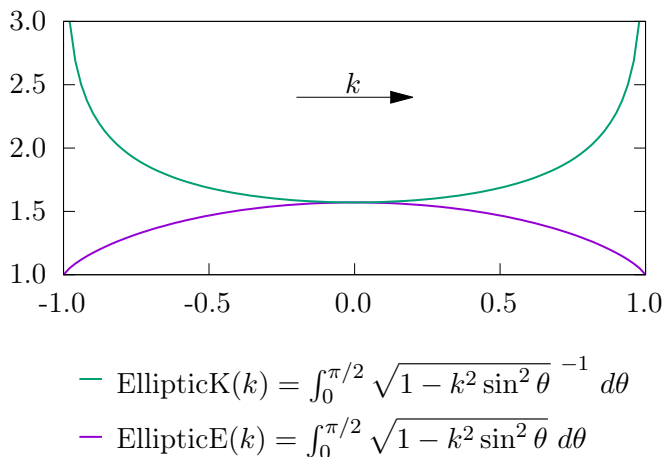


Figure 2: Complete elliptic integrals of the first and second kinds

tikz terminal

The *tikz* terminal is the option I usually recommend. It is simple to use although it is complex underneath, as the gnuplot terminal driver interfaces with an external lua script to produce a *.tex file containing both text and graphics commands for the PGF and TikZ T_EX packages.

```
set terminal tikz size 3.5in,2.4in
```

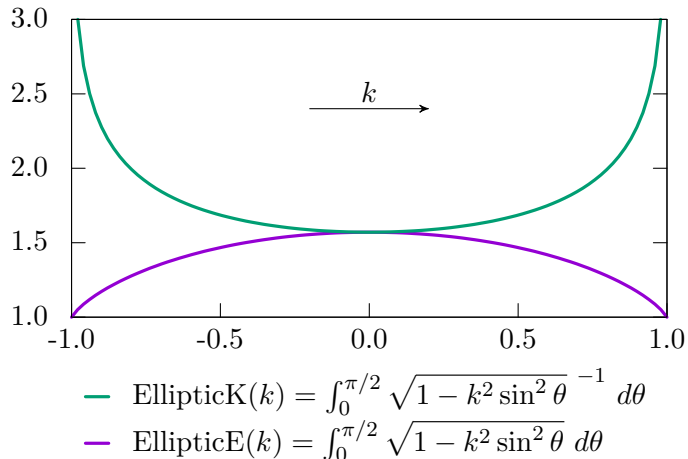


Figure 3: Complete elliptic integrals of the first and second kinds

By default it uses T_EX arrows, but uses gnuplot point symbols. To see the many terminal options available, tell gnuplot `set term tikz help`. The options `tex`, `latex` and `context` tailor the output for a specific T_EX environment. `latex` is the default. Load the corresponding style file at the beginning of your document:

```
\input gnuplot-lua-tikz.tex    % (for plain TeX)
\usepackage{gnuplot-lua-tikz} % (for LaTeX)
\usemodule[gnuplot-lua-tikz]  % (for ConTeXt)
```

context terminal

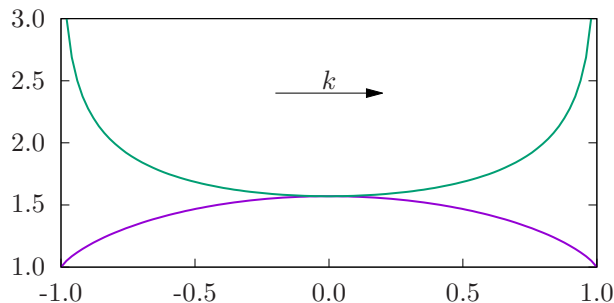
The `context` terminal produces Metafun code for use by the ConTeXt macro package. Typically this terminal is selected as part of a stream of gnuplot commands generated inside a ConTeXt session; the resulting output is then fed back into that session. For more information, see ctan.org/pkg/context-gnuplot.

Other terminals: pstex epslatex pslatex

The `pstex`, `epslatex`, and `pslatex` terminals are perhaps best viewed as serving a slightly different purpose. Rather than aiding inclusion of gnuplot graphs in a \LaTeX document, they are primarily intended to enable \TeX -set annotation of gnuplot graphs included in a PostScript document. The toolchain in this case is typically `gnuplot` \rightarrow `latex` \rightarrow `dvips` \rightarrow `document.ps`.

Nevertheless it is possible to use these terminals to create a standalone `*.eps` file containing both graphics and already-rendered \TeX content, and then include that `*.eps` file in a \LaTeX document using a `\includegraphics{}` statement as shown here.

```
set terminal epslatex color 10 size 3.5in,2.4in standalone
```



$$\begin{aligned} \text{---} & \text{EllipticK}(k) = \int_0^{\pi/2} \sqrt{1 - k^2 \sin^2 \theta}^{-1} d\theta \\ \text{---} & \text{EllipticE}(k) = \int_0^{\pi/2} \sqrt{1 - k^2 \sin^2 \theta} d\theta \end{aligned}$$

Legacy terminals

Two older gnuplot terminals `pstricks` and `texdraw` remain in the default build configuration. They offer limited graphics capability compared to the terminals described above.

Other older gnuplot terminals `latex`, `eepic`, `emtex`, and `tpic` have been deprecated and are no longer included by default when gnuplot is built. If necessary, the file `src/term.h` can be edited to uncomment `#include` statements for the corresponding terminal driver files prior to building gnuplot.

The `metapost` and `metafont` terminals have also been deprecated. They can be including when building gnuplot by adding configuration options `-with-metapost` `-with-metafont`.