Gnuplot FAQ

This document deals with **gnuplot** version 4.0 which is the latest official release as of April 2004.
Its version is `$Revision: 1.15 $`, dated `$Date: 2004/04/14 13:18:59 $`.

Contents

0 Meta – Questions

0.1 Where do I get this document?

The newest version of this document is on the web at `www.gnuplot.info/faq/`.

This document was/is posted sometimes to the newsgroups `comp.graphics.apps.gnuplot`.

0.2 Where do I send comments about this document?

Send comments, suggestions etc via email to the developer mailing list `gnuplot-beta@lists.sourceforge.net`. Please contribute your suggestions with respect to the file `faq.tex` available from `cvs.sourceforge.net/viewcvs.py/gnuplot/faq/`.

1 General Information

1.1 What is gnuplot?

gnuplot is a command-driven interactive function plotting program. It can be used to plot functions and data points in both two- and three-dimensional plots in many different formats. It is designed primarily for the visual display of scientific data. **gnuplot** is copyrighted, but freely distributable; you don't have to pay for it.

1.2 How did it come about and why is it called gnuplot?

The authors of **gnuplot** are: Thomas Williams, Colin Kelley, Russell Lang, Dave Kotz, John Campbell, Gershon Elber, Alexander Woo and many others.

The following quote comes from Thomas Williams:

> I was taking a differential equation class and Colin was taking Electromagnetics, we both thought it'd be helpful to visualize the mathematics behind them. We were both working as sys admin for an EE VLSI lab, so we had the graphics terminals and the time to do some coding. The posting was better received than we expected, and prompted us to add some, albeit lame, support for file data.
>
> Any reference to GNUplot is incorrect. The real name of the program is "**gnuplot**". You see people use "**Gnuplot**" quite a bit because many of us have an aversion to starting a sentence with a lower case letter, even in the case of proper nouns and titles. **gnuplot** is not related to the GNU project or the FSF in any but the most peripheral sense. Our software was designed completely independently and the name "**gnuplot**" was actually a compromise. I wanted to call it "llamaplot" and Colin wanted to call it "nplot." We agreed that "newplot" was acceptable but, we then discovered that there was an absolutely ghastly pascal program of that name that the Computer Science Dept. occasionally used. I decided that "**gnuplot**" would make a nice pun and after a fashion Colin agreed.

1.3 What does gnuplot offer?

- Plotting two-dimensional functions and data points in many different styles (points, lines, error bars)
- Plotting three-dimensional data points and surfaces in many different styles (contour plot, mesh)
- Algebraic computation in integer, float and complex arithmetic
- User-defined functions and hot-keys

- Support for a large number of operating systems, graphics file formats and output devices
- Extensive on-line help
- $\TeX$-like text formatting for labels, titles, axes, data points
- Interactive command line editing and history (most platforms)

1.4 Is gnuplot suitable for scripting?

Yes. Gnuplot can read in files containing additional commands during an interactive session, or it can be run in batch mode by piping a pre-existing file or a stream of commands to stdin. Gnuplot is used as a back-end graphics driver by such higher-level mathematical packages as Octave, and can easily be wrapped in a cgi script for use as a web-driven plot generator.

1.5 Can I run gnuplot on my computer?

Gnuplot is available for a number of platforms. These are: Unix (X11 and NeXTSTEP), Linux, VMS, OS/2, MS-DOS, Amiga, MS-Windows, OS-9/68k, Atari ST, BeOS, and Macintosh.

Please notify the FAQ-maintainer of any further ports you might be aware of.

You should be able to compile the **gnuplot** source more or less out of the box on any reasonable standard (ANSI/ISO C, POSIX) environment.

1.6 Legalities

Gnuplot is freeware authored by a collection of volunteers, who cannot make any legal statement about the compliance or non-compliance of **gnuplot** or its uses. There is also no warranty whatsoever. Use at your own risk.

Citing from the README of a mathematical subroutine package by R. Freund:

> For all intent and purpose, any description of what the codes are doing should be construed as being a note of what we thought the codes did on our machine on a particular Tuesday of last year. If you're really lucky, they might do the same for you someday. Then again, do you really feel *that* lucky?

1.7 Does gnuplot have anything to do with the FSF and the GNU project?

Gnuplot is neither written nor maintained by the FSF. It is not covered by the General Public License, either. It used to be distributed by the FSF, however, due to licensing issues it is no longer.

Gnuplot is freeware in the sense that you don't have to pay for it. However it is not freeware in the sense that you would be allowed to distribute a modified version of your **gnuplot** freely. Please read and accept the `Copyright` file in your distribution.

1.8 Where do I get further information?

See the main gnuplot web page `www.gnuplot.info` and references therein, mainly gnuplot links `gnuplot.sourceforge.net/links.html`.

Some documentation and tutorials are available in other languages than English. See `gnuplot.sourceforge.net/help.html`, section "Localized learning pages about gnuplot", for the most up-to-date list.

2 Setting it up

2.1 What is the current version of gnuplot?

The current released version of **gnuplot** is 4.0.

2.2 Where can I get gnuplot?

The best place is definitely `www.gnuplot.info`. From there you find various pointers to other sites.

The source distribution ("gnuplot-4.0.0.tar.gz" or a similar name) is available from the official distribution site and its mirrors.

The main server is `ftp.gnuplot.info` in `/pub/gnuplot/`. This server is mirrored by several others, among those are

- `mirror.aarnet.edu.au` in `/pub/gnuplot/`
- `ftp.dartmouth.edu` in `/pub/gnuplot/`
- `ftp.irisa.fr` in `/pub/gnuplot/`
- `ftp.gnuplot.vt.edu` in `/pub/gnuplot/`

As of June 1999, the **gnuplot** distribution is also mirrored at the Comprehensive TeX Archive Network (CTAN) in the `graphics/gnuplot` directory. See

- `www.ctan.org/`.

The following platform-specific sites below still exist, but may or may not still hold gnuplot executables.

- Source and binary distributions for the Amiga are available on Aminet `ftp.wustl.edu` in `~aminet/` and its mirrors, for example ftp.uni-kl.de, oes.orst.edu or ftp.luth.se.
- MS-DOS and MS-Windows binaries are available from the above servers and are called for example gp37dos.zip, gp37dj.zip, gp37w16.zip, gp37mgw.zip, gnuplot3.7cyg.zip. OS/2 binaries are called gp37os2.zip.
- The NeXTSTEP front end can be found at `next-ftp.peak.org` in `/pub/next/binaries/plotting/Gnuplot1.2_bin.tar.Z`.
- A version for OS-9/68K can be found at `cabrales.cs.wisc.edu` in `/pub/OSK/GRAPHICS/gnuplot32x.tar.Z`; it includes both an X-Window Systems and a non - X-Window Systems version.
- Versions for the Atari ST and TT, which include some GEM windowing support, are available from `ftp.uni-kl.de` in `/pub/atari/graphics/`, as gplt35st.zip and gplt35tt.zip. They work best under MiNT.
- Executable files, plus documentation in Japanese, exist for the X680x0 on `ftp.csis.oita-u.ac.jp` in `/pub/x68k/fj.binaries.x68000/vol2`.

2.3 Where can I get current development version of gnuplot?

The development version of gnuplot is availble as a cvs source tree online for direct browsing from `www.sourceforge.net/projects/gnuplot/`, section "CVS". You can download all current sources according to the documentation therein; for example by a sequence of commands like

```
export CVSROOT=:pserver:anonymous@cvs.gnuplot.sourceforge.net:/cvsroot/gnuplot
cvs login
cvs -z3 checkout gnuplot
```

Further, before the `./configure` command of gnuplot compilation phase, you have to execute `./prepare` to create the up-to-date configure files.

There are no official preliminary binary releases of gnuplot: you have to compile it yourself. However, you may find unofficial binary releases for some platforms, like OS/2, Windows or Macintosh.

Important note: questions related to the development version should go to `gnuplot-beta@lists.sourceforge.net`.

2.4 How do I get gnuplot to compile on my system?

As you would any other installation. Read the files `README.1ST` and `README`.

- For Unix, use `./configure` (or `./configure --prefix=$HOME/usr` for an installation for a single user), `make` and finally `make install` or `make install-strip`, the latter for smaller executables without debugging information. If you want to make a RPM package, then replace the latest step by `checkinstall` or `checkinstall make install-strip`, supposing the package `checkinstall` on your machine.
- For DOS, if you are using bash and DJGPP, you can just run `djconfig.sh`.
- For other platforms, copy the relevant makefile (e.g. `makefile.os2` for OS/2, or `makefile.mgw` or `makefile.cyg` for Windows) from `config/` to `src/`, optionally update options in the makefile's header, then change directory to `src` and run `make`.

2.5 What documentation is there, and how do I get it?

The documentation is included in the source distribution. Look at the docs subdirectory, where you'll find

- a Unix man page, which says how to start **gnuplot**
- a help file, which also can be printed as a manual
- a tutorial on using **gnuplot** with LaTeX
- a quick reference summary sheet for TeX only

The documentation is built during installation if you have LaTeX installed on your system, look in the directories `docs` and `tutorial`. `make pdf` in the docs subdirectory will make a `gnuplot.pdf` hypertext file ready for browsing or printing.

Online gnuplot documentation is available at `gnuplot.sourceforge.net/documentation.html`.

(Obsolete?) PostScript copies of the documentation can be ftp'd from `ftp.gnuplot.info` in `/pub/gnuplot`, as `manual.ps.Z` and `tutorial.ps.Z`. Documentation about **gnuplot** is available at the **gnuplot** distribution sites in the files `gpdoc.zip` and `gpdoc2.zip`.

2.6 Worked examples

There is a directory of worked examples in the the source distribution. These examples, and the resulting plots, may also be found at `gnuplot.sourceforge.net/demo/`.

2.7 How do I modify gnuplot, and apply 'patches'?

For this, you will need to recompile **gnuplot**.

Modifications people make are either done by replacing files, such as terminal drivers, or by 'patching'. If a file is a replacement, it will probably tell you in its README or in the lines at the beginning.

To patch a file, you need the `patch` utility, and possibly also `automake` and `autoconf`. On many UNIX systems these will already be installed; If they aren't, you can find them wherever GNU software is archived. Typical command for applying a patch is `patch -p0 <newfunctionality.diff`.

There is repository of contributed patches in the "Patches" section on gnuplot's sourceforge site http://www.sourceforge.net/p

3 Working with it.

3.1 How do I get help?

Read this document.

Give the `help` command at the initial prompt. After that, keep looking through the keywords. Good starting points are `plot` and `set`.

Read the manual, if you have it.

Look through the demo subdirectory; it should give you some ideas.

Ask your colleagues, the system administrator or the person who set up **gnuplot**.

If all these fail, please upgrade to the newest version of **gnuplot** or urge your system-administrator to do so. Then post a question to `comp.graphics.apps.gnuplot` or send mail to the gatewayed mailing list `gnuplot-info@lists.sourceforge.net`. Do not forget to cite the version number and the operating system. If you want to subscribe to the mailing list, send a mail to `majordomo@lists.sourceforge.net` with the body of the message being `subscribe info-gnuplot`. Please don't do this if you can get `comp.graphics.apps.gnuplot` directly. If you post a question there, it is considered good form to solicit e-mail replies and post a summary.

3.2 How do I print out my graphs?

The kind of output produced is determined by the `set terminal` command; for example, `set terminal postscript` will produce the graph in PostScript format. Output can be redirected using the `set output` command.

As an example, the following prints out a graph of sin(x) on a Unix machine running the X-Window System.

```
gnuplot> plot [-6:6] sin(x)
gnuplot> set terminal postscript
Terminal type set to 'postscript'
Options are 'landscape monochrome "Courier" 14'
gnuplot> set output "sin.ps"
gnuplot> replot
gnuplot> set output              # set output back to default
gnuplot> set terminal x11        # ditto for terminal type
gnuplot> ! lp -ops sin.ps        # print PS File (site dependent)
request id is lprint-3433 (standard input)
lp: printed file sin.ps on fg20.rz.uni-karlsruhe.de (5068 Byte)
!
gnuplot>
```

Using the platform-independent way of restoring terminal by `set term push/pop` commands, do it by

```
gnuplot> set terminal postscript eps color lw 15 "Helvetica" 20
gnuplot> set out 'a.eps'
gnuplot> replot
gnuplot> set term pop
```

The command `set term pop` without a previous corresponding `set term push` switches the terminal back to the startup terminal, e.g. x11, pm or win.

In Microschrott Windows you click in the upper left corner of the graph window and print directly from there.

3.3 How do I include my graphs in <word processor>?

Basically, you save your plot to a file in a format your word processor can understand (using `set term` and `set output`, see above), and then you read in the plot from your word processor. Vector formats (PostScript, emf, svg, pdf, TEX, LATEX, etc) should be preferred, as you can scale your graph later to the right size.

Details depend on which word processor you use; use `set term` to get a list of available file formats.

Many word processors can use Encapsulated PostScript for graphs. This can be generated by the `set terminal postsc` command. Note that it is a good idea to check and correct the bounding box of the graphs in the eps files (manually or by the fixbb script from gnuplot webpage), as you have to correct this box for any eps figure produced by whichever program. Some (most?) word processors do not preview the actual image in the eps file, and you have to add the preview image yourself. You can use the GSView viewer for this (available for OS/2, Windows and X11), or some Unix ps tool. Note that the preview image increases size of the eps file; the smallest increase you may get by choosing Tiff 6 Packbits.

Into some office applications, like OpenOffice.org, or into applications in the Windows world, you can insert vectorial images produces by the emf terminal type.

With TEX, it depends on what you use to print your dvi files. If you use dvips or dvi2ps, you can use Encapsulated PostScript. For emTeX (popular for OS/2 and MS-DOS), you can use emTeX, otherwise use the LATEX terminal type, which generates a picture environment. You can also use epslatex to separate the graphics and text parts. Other possibilities include pslatex or pstex terminals, and metafont or metapost terminals.

With TEX processed by pdftex or pdflatex, you can use png, jpeg and pdf terminal types. You can also use the postscript eps terminal and convert the eps file externally to pdf by `epstopdf`. Another choice is the epslatex terminal, after converting the eps part to pdf as above (the TEX part can remain unchanged).

Most word processors can import bitmap images (png, pbm, etc). The disadvantage of this approach is that the resolution of your plot is limited by the size of the plot at the time it is generated by gnuplot, which is generally a much lower resolution than the document will eventually be printed in.

Under IBM OS/2, MacOS and Micro$oft Windows you can use the clipboard to copy your graph and paste it into your favourite word processor.

The mif terminal type produces output for FrameMaker.

3.4 How do I edit or post-process a gnuplot graph?

This depends on the terminal type you use.

- X11 toolkits: You can use the terminal type fig and use the **xfig** drawing program to edit the plot afterwards. You can obtain the xfig program from its web site `www.xfig.org`. More information about the text-format used for fig can be found in the fig-package.

  You may use the tgif terminal, which creates output suitable for reading within `tgif` (`bourbon.cs.umd.edu:8001/tgif/`), an interactive 2-D drawing tool under X11.

- You may use the svg terminal (scalable vector graphics), which can be further edited by a svg editor, e.g. **Sodipodi** (`sodipodi.sourceforge.net`), **Sketch** (`sketch.sourceforge.net`) or **Dia** (`http://www.lysator.liu.se/~alla/dia`).

- PostScript or PDF output can be edited directly by tools such as Adobe Illustrator or Acrobat, or can be converted to a variety of other editable vector formats by the **pstoedit** package. Pstoedit is available at `www.pstoedit.net`.

- The mif terminal type produces an editable FrameMaker document.

- Bitmapped graphics (e.g. png, jpeg, pbm) can be edited using tools such as ImageMagick or Gimp. In general, you should use a vector graphics program to post-process vector graphic formats, and a pixel-based editing program to post-process pixel graphics.

3.5 How do I change symbol size, line thickness and the like?

Gnuplot offers a variety of commands to set line and point properties, including color, thickness, point shape, etc. The command `test` will display a test page for the currently selected terminal type showing the available pre-defined combinations of color, size, shape, etc. The `set style` command can be used to define additional combinations.

3.6 How do I generate plots in GIF format?

GIF support was provided by an external library, libgd. Version 1.6 of the gd library dropped support for gif in favour of the superior png format. More recent versions of libgd also support jpeg output. If your installation of gnuplot is linked to the gd library, you will get support for whatever formats (gif, png, jpeg) are in that version of libgd.

Thus, nowadays, the usual way is to output the image as png and then use your favourite program to convert it into gif, either using command line (e.g. `convert f.png f.gif` or `nconvert -out gif f.png`) or by any GUI program. Another possibility is to output the image as (encapsulated) postscript and convert (export) it into gif or png by ghostscript, e.g. `convert -density 150 f.eps f.gif` or by any ghostscript-based GUI like gsview, gv or kghostview.

3.7 Can I animate my graphs?

First have a look at animate.dem in the demo directory of **gnuplot**. Basically, animated graphs are a sequence of plots in a suitable format.

Then have a look at the tool whirlgif 3.04, available at `www.danbbs.dk/~dino/whirlgif`. It reads run-length encoded gifs and packs them into a minimal animation. On the web-pages you will find a manual and an example.

You can also write a small script to get **gnuplot** to output a family of GIF files, then have it execute some animator such as gifsicle: `www.lcdf.org/~eddietwo/gifsicle` or gifmerge `the-labs.com/GIFMerge`.

mpeg_encode will encode a sequence of images into an mpeg format movie.

3.8 How do I plot implicit defined graphs?

Implicit graphs or curves cannot be plotted directly in **gnuplot**. However there is a workaround.

```
gnuplot> # as example. Place your definition in the following line
gnuplot> f(x,y) = y - x**2 / tan(y)
gnuplot> set contour base
gnuplot> set cntrparam levels discrete 0.0
gnuplot> set nosurface
gnuplot> set term table
gnuplot> set out 'curve.dat'
gnuplot> splot f(x,y)
gnuplot> set out
gnuplot> set term pop
gnuplot> plot 'curve.dat' w l
```

The trick is to draw the single contour line z=0 of the surface z=f(x,y), and store the resulting contour curve to a **gnuplot** datafile.

3.9 How to fill an area between two curves

A plot with filled area between two given curves requires a parametric plot with `filledcurves closed`. The example below demonstrates this for two curves f(x) and g(x) with a tricky "folded" parameter `t`:

```
set parametric
f(x)=cos(x)
g(x)=sin(x)
xmax=pi/4
set xrange [0:xmax]
set trange [0:2*xmax]
path(t) = ( t<= xmax ? f(t) : g(2*xmax-t) )
fold(t) = (t <=xmax ? t : 2*xmax - t)
plot fold(t),path(t) with filledcurves closed
```

Note that the above code fills area between the two curves, not area satisfying inequality g(x)<f(x). If you want the latter, you should use the ternary operator in `path(t)` to return an undefined value (0/0) if the inequality is not satisfied.

See the documentation for `help parametric`, `help filledcurves`, and `help ternary`.

3.10 Pm3d splot from a datafile does not draw anything

You do `set pm3d; splot 'a.dat'` and no plot but colorbox appears. Obviously, there is no blank line in between two subsequent scans (isolines) in the data file. Add blank lines! If you are curious what this means, then don't hesitate to look to files like `demo/glass.dat` or `demo/triangle.dat` in the gnuplot demo directory.

You can find useful the following awk script (call it e.g. `addblanks.awk`) which adds blank lines to a data file whenever number in the first column changes:

```
/[:blank:]*#/ {next} # ignore comments (lines starting with #)
NF < 3 {next} # ignore lines which don't have at least 3 columns
$1 != prev {printf "\n"; prev=$1} # print blank line
{print} # print the line
```

Then, either preprocess your data file by command `awk -f addblanks.awk <a.dat` or plot the datafile under a unixish platform by `gnuplot> splot "<awk -f addblanks.awk a.dat"`.

3.11 Drawing a (color) map, i.e. 2D projection of 3D data

Use `set view map; unset surface` or `set pm3d map` rather than `set view 180,0`. The latter facilitates drawing matrices or data files as maps, even without the necessity for matrix-like data organization (gridding). It is possible to decrease the output postscript file size by postprocessing it by `pm3dCompress.awk` or `pm3dConvertToImage.awk`.

Note there is a new plotting style `with image` for plotting 2D color images with support for almost arbitrary text or binary files in "Patches" section on gnuplot's sourceforge site `www.sourceforge.net/projects/gnuplot/`.

3.12 How to overlay dots/points scatter plot onto a pm3d map/surface

Use the explicit (see also implicit) switch of the pm3d style:

```
gnuplot> set pm3d explicit
gnuplot> splot x with pm3d, x*y with points
```

3.13 How to draw black contour plot, and contours with labels

Drawing contour plot in one colour requires to write contours into a temporary file using the table terminal:

```
set contour base; set cntrparam levels 15; unset surface; set view map
splot x*x+y*y; pause -1
set term table; set out 'contour.dat'
replot; set out; set term pop
```

Now, for drawing it in 2D, do

```
reset
plot 'contour.dat' with line -1
```

end for contours in 3D do

```
reset
# Change single blank lines to double blank lines
!awk "NF<2{printf\"\n\"}{print}" <contour.dat >contour1.dat
plot 'contour1.dat' with line -1
```

See also the following question "How to overlay contour plot over pm3d map/surface".

Labelling contours by their z-value can be achieved by a suitable script generating automatically the appropriate `set label` commands; you can find one at gnuplot scripts page `gnuplot.sourceforge.net/scripts/index.html#tricks-here`.

3.14 How to overlay contour plot over pm3d map/surface

It is necessary to write contours into a temporary file using the table terminal, and then use this file in the final drawing without set contours. The following example demonstrates this for a map; for surface, remove `set pm3d map` and put `set ticslevel 0`.

```
# Write contours of function  x*x-y*y  to a (temporary) file
set term table
set out 'contour.dat'
set contour base; set cntrparam level 20
unset surface
splot x*x-y*y
set out; set term pop

# Change single blank lines to double blank lines
!awk "NF<2{printf\"\n\"}{print}" <contour.dat >contour1.dat

# Draw the plot
reset
set palette gray
set palette gamma 2.5
set pm3d map
set pm3d explicit
splot x*x+y*y with pm3d, 'contour1.dat' with line -1
!rm contour.dat contour1.dat
```

The last command deletes the two temporary files.

3.15 Color facets with pm3d

It is possible to draw colors facets of a 3D objects, organized in such a file:

```
# triangle 1
x0 y0 z0 <c0>
x1 y1 z1 <c1>

x2 y2 z2 <c2>
x2 y2 z2 <c2>


# triangle 2
x y z
...
```

Notice the positioning single and double blank line. <c> is an optional color.

Then plot it by (either of splot's):

```
set pm3d
set style data pm3d
splot 'facets.dat'
splot 'facets_with_color.dat' using 1:2:3:4
```

Note that you avoid surface lines by `set style data pm3d` or `splot ... with pm3d`.

In the above example, pm3d displays triangles as independent surfaces. They are plotted one surface after another as found in the data file. Parts overlapping in 2D projection are overdrawn.

Gnuplot is not 3D modeling program. Its hidden routines apply for points and lines, but not for faces. Without handling the data as a collection of faces, there would be no surface anything could be hidden behind.

The 'hidden3d' algorithm works by using the input data in two ways: first, to set up a collection of triangles (made from a mesh of quadrangles) that form the surface, second as a collection of edges. It then goes through all those edges, checking what parts of them are not hidden behind any faces, and draws those.

Consequently, gnuplot won't draw your surface or 3D object as a virtual reality. It works OK for `set pm3d map` but for true 3D you would be probably more happy writing a convertor of your facets into a VRML file.

3.16 Palette for printing my color map on color as well as blank&white printer?

I think it is this one, for example: `set palette rgbformulae -25,-24,-32`. Can somebody prove this?

4 Wanted features

4.1 What's new in gnuplot 3.7, 4.0 etc?

Too many things to be named here. Please refer to the `NEWS` file in the source distribution, or the News section in the gnuplot documentation.

4.2 Does gnuplot support a driver for <graphics format>?

To see a list of the available graphic drivers for your installation of **gnuplot**, type `set term`.

Some graphics drivers are included in the normal distribution, but are uncommented by default. If you want to use them, you'll have to change file `gnuplot/src/term.h`, and recompile.

4.3 Does gnuplot have hidden line removal?

Version 3.7 supports hidden line removal on all platforms; use the command `set hidden3d`.

The 16-bit binaries of **gnuplot** support the hidden line removal only partially as the hidden-line algorithm may hit the 640k memory limit.

4.4 Does gnuplot support bar-charts/histograms/boxes?

Use the style `with boxes` for bar charts. To get filled boxes, use `set style fill`.

Bernhard Reiter wrote an AWK script to post-process fig-terminal output. Please have a look at `www.usf.uni-osnabrueck.de/~breiter/tools/gnuplot/barcharts.en.html`.

There is a large patch (#606169) against the development version on SourceForge awaiting inclusion in a future release of **gnuplot**. This adds a new plotting style `with histograms`, and supports both clustered histograms and stacked histograms.

4.5 Does gnuplot support pie charts?

It's not possible in **gnuplot**, but have a look at `www.usf.uni-osnabrueck.de/~breiter/tools/piechart/piecharts.en.html`

4.6 Does gnuplot quarterly time charts?

It's not possible in **gnuplot**, but have a look at `ricardo.ecn.wfu.edu/~cottrell/qplot`. The corresponding file `qplot.zip` can be obtained from the contrib directory on any **gnuplot** server.

4.7 Can I put multiple pages on one page?

Yes. `set multiplot`.

If you use the postscript terminal and plot one graph per page you can use the program mpage (`www.mesa.nl/pub/mpage`) to print multiple logical pages per physical page. A similar program is the psnup program in the psutils package. This package is available at any CTAN mirror.

4.8 Does gnuplot support multiple y-axes on a single plot?

Yes. You can have 2 x- and 2 y-axes per plot. The additional axes are called x2 and y2. See `help plot`.

4.9 Can I put both commands and data into a single file?

This is possible by the new `plot "-"` possibility. The `plot "-"` command allows to read the data to be plot from standard input or the current batch job.

```
gnuplot> plot "-"
1 1
2 4
3 9
e
```

4.10 Can I put Greek letters and super/subscripts into my labels?

The enhanced option in some terminals (currently postscript, Presentation Manager, png, pdf, and x11) lets you use sub- and superscripts. It also allows to use Greek letters and symbols via symbol fonts, to the extent that these are supported by the underlying font libraries on your system. Versions through 2.0.15 of libgd (needed for png and jpeg) must be patched in order to handle symbol fonts.

You might try using the LaTeX terminal type and putting text like `"\\alpha_{3}"` or `'\alpha_{3}'` . If you include your **gnuplot**-graphs into a LaTeX document you can use the LaTeX-package psfrag to typeset any characters into your graphs.

One more possibility is to use the MetaPost terminal. It supports TeX syntax and is converted onto encapsulated PostScript by mpost.

4.11 How do I include accented characters

To obtain accented characters like ü or ñ in your labels you should use 8bit character codes together with the appropriate encoding option. See the following example:

```
gnuplot> set encoding iso_8859_1
gnuplot> set title "M\374nchner Bierverbrauch \374ber die Jahre"
gnuplot> plot "bier.dat" u 1:2
```

Consequently, you can type labels in Czech, French, Hungarian, Russian... by means of an appropriate `set encoding`. However, you cannot mix two encodings in one file (e.g. accents for west and east latin encodings). That you can only do with a TeX output.

4.12 Can I do 1:1 scaling of axes?

Use `set size square`.

4.13 Can I put different text sizes into my plots?

Some terminals can, others can't. Some allow you to choose a font size for the entire plot. Terminals supporting the "enhanced text" mode (like postscript, pm, x11, png, pdf) allow you to change fonts and text sizes within a plot. Look at the help for these terminals.

4.14 How do I skip data points?

By specifying `?` as a data value, as in

```
1 2
2 3
3 ?
4 5
```

See also `set missing`. See also `set datafile commentschars` for specifying comment characters in data files.

4.15 How do I plot every nth point?

This can be specified with various options for the command `plot`, for example `plot 'a.dat' every 2`.

4.16 How do I plot a vertical line?

Depending on context, the main methods are:

- `set arrow .... .... nohead` where you have to compute explicitly the start and the end of the arrow.
- generate (inlined) datapoints and plot them
- switch to parametric mode

4.17 How do I plot data files

Easily: by a command `plot 'a.dat'`. In 3D, use `splot 'a.dat'` – but don't forget to put a blank line in between two subsequent scans (isolines), otherwise you will get an error that the data is not gridded; see also question 3.10. If your data are not gridded, then use `set dgrid3d {many options}`.

4.18 How do I replot multiplot drawing

You cannot directly: gnuplot supports `replot` command, not `remultiplot`. You have to write the complete sequence of commands since `set multiplot` till `unset multiplot` into a script file. Then you can `load` the script into gnuplot as many times as you need for replotting the drawing to different terminals or output files.

5 Miscellaneous

5.1 I've found a bug, what do I do?

First, try to see whether it actually is a bug, or whether it is a feature which may be turned off by some obscure set–command.

Next, see whether you have an old version of **gnuplot**; if you do, chances are the bug has been fixed in a newer release.

Fixes for bugs reported since the release of the current version are held in the `patches` directory at **gnuplot** distribution sites. Before submitting a bug report, please check whether the bug in question has already been fixed.

If, after checking these things, you still are convinced that there is a bug, proceed as follows. If you have a fairly general sort of bug report, posting to `comp.graphics.apps.gnuplot` is probably the way to go. If you have investigated a problem in detail, especially if you have a context or unified diff that fixes the problem, please e-email a report to `gnuplot-bug@lists.sourceforge.net`.

The bug-gnuplot list is for reporting and collecting bug fixes, the `comp.graphics.apps.gnuplot` newsgroup will be more help for finding work arounds or actually solving **gnuplot** related problems. If you do send in a bug report, be sure and include the version of **gnuplot** (including patchlevel) as shown by the command `show version long`, terminal driver, operating system, an exact description of the bug and input which can reproduce the bug. Failure to indicate these details can render a solution to your problem almost impossible. Also, any context diffs should be referenced against the latest official version of **gnuplot** if at all possible.

5.2 Can I use gnuplot routines for my own programs?

On systems supporting pipes, you can pipe commands to **gnuplot** from other programs. Many applications with gnuplot as the graphics engine, like Octave (`www.octave.org`), uses this method. This also works from a cgi script to drive **gnuplot** from a forms-based web page.

John Campbell `jdc@nauvax.ucc.nau.edu` modified a much earlier version of **gnuplot** (3.5) to be a library of C subroutines callable from a C program. Gnuplot itself has changed radically since then, and we are not aware of any plans to create a similar library based on the current version.

5.3 What extensions have people made to gnuplot? Where can I get them?

Many extensions or patches are available on the "Patches" page of the gnuplot development site `sourceforge.net/tracker/?group_id=2055&atid=302055`. The current development version will generally include some of these being debugged for inclusion in a later official release of gnuplot.

Older extensions, which may or may not work with the current version, are available from `ftp.ucc.ie` in `/pub/gnuplot/contrib/`.

Some extensions available:

- `barchart_via_fig`: awk scripts to produce barcharts with filled boxes.
- `date-errorbar`: allows dates in the hi/lo fields for errorbars.
- `perltk`: A perl/tk canvas widget.
- `polyg.patch`: Implements a polygon plotting style.
- `xgfe`: graphical front end using the Qt widgets available at `www.tug.org/tex-archive/graphics/gnuplot/contrib/`.
- `Gnuplot.py`: A python package to create graphs from within python. More information at `gnuplot-py.sourceforge.net`.

5.4 I need an integration, fft, iir-filter,...!

Gnuplot has been and is a plotting program, not a data processing or mathematical program suite. Therefore **gnuplot** can't do that. Look into the demo file "bivariat.dem" for a basic implementation of an integration.

For more sophisticated data-processing read the next section.

5.5 Can I do heavy-duty data processing with gnuplot? or What is beyond gnuplot?

gnuplot alone is not suited very well for this. One thing you might try is `fudgit`, an interactive multi-purpose fitting program written by Martin-D. Lacasse (`isaac@frodo.physics.mcgill.ca`). It can use **gnuplot** as its graphics back end and is available from `ftp.physics.mcgill.ca` in `/pub/Fudgit/fudgit_2.33.tar.Z` and from the main Linux server, tsx-11.mit.edu and its numerous mirrors around the world as `/pub/linux/sources/usr.bin/fudgit-2.33.tar.z`. Versions are available for AIX, Data General, HP-UX, IRIX 4, Linux, NeXT, Sun3, Sun4, Ultrix, OS/2 and MS-DOS. The MS-DOS version is available on simtel20 mirrors (simtel20 itself has closed down) in the "math" subdirectory as `fudg_231.zip`.

Carsten Grammes has written a fitting program which has been merged into **gnuplot** 3.7.

Michael Courtney has written a program called lsqrft, which uses the Levenberg-Marquardt - algorithm for fitting data to a function. It is available from `hobbes.nmsu.edu` in `/pub/os2/apps/analysis/lsqrft15.zip`; sources, which should compile on Unix, and executables for MS-DOS and OS/2 are available. There is an interface to the OS/2 presentation manager.

You might also want to look at the applications developed by the Software Tools Group (STG) at the National Center for Supercomputing Applications. Ftp to ftp.ncsa.uiuc.edu and get the file README.BROCHURE for more information.

You can also try pgperl, an integration of the PGPLOT plotting package with Perl 5. Information can be found at `www.ast.cam.ac.uk/AAO/local/www/kgb/pgperl`, the source is available from `ftp.ast.cam.ac.uk` in `/pub/kgb/pgperl/` or `linux.nrao.edu` in `/pub/packages/pgperl/`.

Another possibility is **Octave**. To quote from its README: Octave is a high-level language, primarily intended for numerical computations. Octave is licensed under GPL, and in principle, it is a free Matlab clone. It provides a convenient command line interface for solving linear and nonlinear problems numerically. The latest released version of Octave is always available from `www.octave.org`. By the way, octave uses **gnuplot** as its plotting engine, so you get a data-processing program on top of **gnuplot**.

Finally, there is `scilab` at `www-rocq.inria.fr/scilab/` doing about the same as matlab. It is free but copyrighted software.

5.6 Mouse in my interactive terminal does not work

If your mouse is not working, try to hit 'm' in the interactive terminal to switch mousing on/off. See below for the list of supported interactive terminals.

If it still does not run, then either gnuplot has not been configured or compiled with mouse support, or you have not properly installed it, or running an older version of gnuplot (check your `PATH`).

If your gnuplot is running as the plotting engine of Octave under X11, then please put `gset mouse` into your `$HOME/.octaverc`. According to gnuplot's `help x11`, gnuplot under x11 running through a pipe needs `set mouse` to be executed before launching the x11 plot window.

5.7 How to use hotkeys in my interactive terminals

There are several hotkeys available in interactive terminals. Currently the following interactive terminals support hotkeys and mousing: OS/2 Presentation Manager, X11, Windows, and GGI. Hit 'h' in the terminal to get list of hotkeys. See `help new` or the **What is New in 4.0** section in the docs for a brief guide over mousing and hotkeys. Further, you may read `help mouse` and `help bind` for more information.

5.8 I have ported gnuplot to another system, or patched it. What do I do?

If your patch is small, mail it to `gnuplot-beta@lists.sourceforge.net`, with a thorough description of what the patch is supposed to do, which version of **gnuplot** it is relative to, etc. Well, please do it always with respect to the current development version of gnuplot (see 'cvs' above).

Nowadays, the preferred way of submitting, commenting and upgrading patches are via 'Patches' section on `www.sourceforge.net/projects/gnuplot/`. You may want to send a note to `gnuplot-beta@lists.sourceforge.net` for more lively discussion.

5.9 I want to help in developing the next version of gnuplot. What can I do?

Join the **gnuplot** beta test mailing list by sending a mail containing the line `subscribe gnuplot-beta` in the body (not the subject) of the mail to `Majordomo@lists.sourceforge.net`.

Also check with `sourceforge.net/projects/gnuplot` about latest source for beta releases for development.

5.10 Open questions for inclusion into the FAQ?

`gnuplot-beta@lists.sourceforge.net`.

Please submit your questions (along with the answer) to `gnuplot-beta@lists.sourceforge.net`.

6 Making life easier

6.1 How do I plot two functions in non-overlapping regions?

Use a parametric plot. An example:

```
gnuplot> set parametric
gnuplot> a=1
gnuplot> b=3
gnuplot> c=2
gnuplot> d=4
gnuplot> x1(t) = a+(b-a)*t
gnuplot> x2(t) = c+(d-c)*t
gnuplot> f1(x) = sin(x)
gnuplot> f2(x) = x**2/8
gnuplot> plot [t=0:1] x1(t),f1(x1(t)) title "f1", x2(t), f2(x2(t)) title "f2"
```

You can also use **gnuplot**'s ability to ignore mathematically undefined expressions: the expression `1/0` is silently ignored, thus a construction like

```
gnuplot> set xran [-10:10]
gnuplot> plot (abs(x)>0.5?1/0: x**2)
```

plots a quadratic function only for `|x| < 0.5`.

6.2 How do I run my data through a filter before plotting?

If your system supports the popen() function, as Unix does, you should be able to run the output through another process, for example a short awk program, such as

```
gnuplot> plot "< awk ' { print $1, $3/$2 } ' file.in"
```

The plot command is very powerful and is able to do some arithmetic on datafiles. See `help plot`.

The above filtering works seamlessly under Unixes and OS/2. It can work under MS Windows as well, but that is for experienced users: (A) When gnuplot has been compiled by cygwin with the unixish way of `./configure`; `make` with X11 terminal instead of the 'windows' terminal. You have to run this under an X-server. This procedure is out of knowledge for usual users, but powerful for others. (B) Compile gnuplot yourself by `makefile.mgw` or `makefile.cyg` and set `PIPES=1` therein. The drawback is that each `wgnuplot.exe` will be accompanied by a boring shell box.

6.3 How do I make it easier to use gnuplot with LaTeX?

There is a set of LaTeX macros and shell scripts that are meant to make your life easier when using **gnuplot** with LaTeX. This package can be found on `ftp.dartmouth.edu` in `pub/gnuplot/latex.shar`, by David Kotz. For example, the program "plotskel" can turn a gnuplot-output file plot.tex into a skeleton file skel.tex, that has the same size as the original plot but contains no graph. With the right macros, the skeleton can be used for preliminary LaTeX passes, reserving the full graph for later passes, saving tremendous amounts of time.

6.4 How do I save and restore my settings?

Use the `save` and `load` commands for this; see `help save` and `help load` for details.

You can save the current terminal and restore it later without touching the filesystem by `set term push` and `set term pop`, respectively.

6.5 How do I plot lines (not grids) using splot?

If the data in a data file for splot is arranged in such a way that each one has the same number of data points (using blank lines as delimiters, as usual), splot will plot the data with a grid. If you want to plot just lines, use a different number of data entries (you can do this by doubling the last data point, for example). Don't forget to `set parametric` mode, of course.

6.6 How do I plot a function f(x,y) that is bounded by other functions in the x-y plane?

An example:

```
gnuplot> f(x,y) = x**2 + y **2
gnuplot> x(u) = 3*u
gnuplot> yu(x) = x**2
gnuplot> yl(x) = -x**2
gnuplot> set parametric
gnuplot> set cont
gnuplot> splot [0:1] [0:1] u,yl(x(u))+(yu(x(u)) - yl(x(u)))*v,\
> f(x(u), (yu(x(u)) - yl(x(u)))*v)
```

6.7 How do I turn off <feature> in a plot?

Most gnuplot features are controlled by a corresponding set/unset command. If a feature is enabled by default, or by using `set <feature>`, then you should be able to turn it by using `set no<feature>`. However, the prefered syntax in version 4.0 is `unset <feature>`.

6.8 How do I call gnuplot from my own programs?

On unix-like systems, commands to gnuplot can be piped via stdin. Output from **gnuplot**'s `print` command can be read via a named pipe. On M$ Windows platforms, due to the lacking standard input (stdin) in GUI programs, you need to use the helper program `pgnuplot` which should be included in your gnuplot for M$W distribution package. Reading **gnuplot** output may be impossible.

6.9 What if I need h-bar (Planck's constant)?

There is no predefined variable like pi. However to put h-bar as a character into the label, you must use the PostScript terminal. You can play around with constructs like `@{/=56 -} {/=24 h}` or `{/=8 @{/Symbol=24 -} _` In the latter, the "-" (a long one in /Symbol) is non-spacing and 24-pt. The 14-pt "h" is offset by an 8-pt space (which is the space preceding the "_") but smaller, since it's written as a subscript. But these don't look too much like the hbar we're used to, since the bar is horizontal instead of sloped. I don't see a way to get that. I tried using an accent (character 264 in iso-latin-1 encoding), but I haven't found a way to scale and position the pieces correctly.

One more possibility would be `{/=14 @^{/Symbol=10 -}{/=14 h}}`.

(This is a hint by Richard Crawford).

6.10 How do a produce blank output page?

Well, you probably don't want a blank page, but page with a just a title (overprinting title in another graph in multiplot page):

```
reset; unset xtics; unset ytics
unset border; unset key
set title 'Title on an empty page'
plot [][0:1] 2
```

7 Common problems

7.1 Gnuplot is not plotting any points under X11! How come?

On VMS, you need to make several symbols:

```
$ gnuplot_x11 :== $disk:[directory]gnuplot_x11
$ gnuplot :== $disk:[directory]gnuplot.exe
$ def/job GNUPLOT$HELP disk:[directory]gnuplot.hlb
```

Then run **gnuplot** from your command line, and use `set term x11`.

If you run **gnuplot** on Unix systems, be sure that the newest `gnuplot_x11` is the first in your search path. Command `which gnuplot_x11` will help you.

7.2 My isoline data generated by a Fortran program is not handled correctly. What can I do?

Update to the newest **gnuplot**. **Gnuplot** 3.7 is able to read Fortran-style files where a blank line can contain more than a linefeed.

7.3 Why does gnuplot ignore my very small numbers?

Gnuplot treats all numbers less than 1e-08 as zero, by default. Thus, if you are trying to plot a collection of very small numbers, they may be plotted as zero. Worse, if you're plotting on a log scale, they will be off scale. Or, if the whole set of numbers is "zero", your range may be considered empty:

```
gnuplot> plot 'test1'
Warning: empty y range [4.047e-19:3e-11], adjusting to [-1:1]
gnuplot> set yrange [4e-19:3e-11]
gnuplot> plot 'test1'
              ^
y range is less than `zero`
```

The solution is to change **gnuplot**'s idea of "zero":

```
gnuplot> set zero 1e-20
```

For more information, type `help set zero`.

7.4 Gnuplot is not plotting on the screen when run from command line via '`gnuplot filename.gp`'

Obviously, it draws (unless there is an error in the script file), but the plot dissappears immediately when the script is completed.

Solution 1: Put a `pause -1` after the plot command in the file, or at the file end.

Solution 2: Use command `gnuplot filename.gp -` (yes, dash is the last parameter) to stay in the interactive regime when the script completes.

Solution 3A: On an X-Window System system, you can also use the `-persist` option, the X11 window is then not closed. Close the X11 window by typing "q" when the focus is on it.

Solution 3B: On M$ Windows, you can also use either `-persist` or `/noend`.

Solution 4: For OS/2 PM terminal, use `set term pm persist` or `set term pm server`. For X11 terminal, use `set term x11 persist`.

7.5 My formulas (like 1/3) are giving me nonsense results! What's going on?

Gnuplot does integer, and not floating point, arithmetic on integer expressions. For example, the expression 1/3 evaluates to zero. If you want floating point expressions, supply trailing dots for your floating point numbers. Example:

```
gnuplot> print 1/3
                0
gnuplot> print 1./3.
                0.333333
```

This way of evaluating integer expressions is shared by both C and Fortran.

7.6 Set output 'filename' isn't outputting everything it should!

You may need to flush the output with a closing `set output`.

7.7 When using the LaTeX–terminal, there is an error during the LaTeX–run!

Please upgrade to at least **gnuplot** 3.7. The LaTeX2ε-core no longer includes the commands "\Diamond" and "\Box"; they are included in the latexsym package, which is part of the base distribution and thus part of any LaTeX implementation. Please do not forget to use this package.

7.8 The `exit` command does not work as documented!

This is an old bug and is fixed in newer releases.

7.9 I can't find the demos and example files at the URLs in the documentation!

The examples have been removed from the NASA site mentioned in older documentation. You can currently find the version 3.7 examples at `www.gnuplot.vt.edu/gnuplot/gpdocs`. Version 4.0 examples are at `gnuplot.sourceforge.net/demo`.

7.10 Calling gnuplot in a pipe or with a gnuplot-script doesn't produce a plot!

You can call **gnuplot** by using a short Perl-script like the following:

```
#!/usr/local/bin/perl -w
open (GP, "|/usr/local/bin/gnuplot -persist") or die "no gnuplot";
# force buffer to flush after each write
use FileHandle;
GP->autoflush(1);
print GP,"set term x11;plot '/tmp/data.dat' with lines\n";
close GP
```

Gnuplot closes its plot window on exit. The `close GP` command is executed, and the plot window is closed even before you have a chance to look at it.

There are three solutions to this: first, use the `pause -1` command in **gnuplot** before closing the pipe. Second, close the pipe only if you are sure that you don't need **gnuplot** and its plot window anymore. Last, you can use the command line option `-persist`: this option leaves the X-Window System plot window open.

8 Credits

Gnuplot 3.7's main contributors are (in alphabetical order) Hans-Bernhard Broeker, John Campbell, Robert Cunningham, David Denholm, Gershon Elber, Roger Fearick, Carsten Grammes, Lucas Hart, Lars Hecking, Thomas Koenig, David Kotz, Ed Kubaitis, Russell Lang, Alexander Lehmann, Alexander Mai, Carsten Steger, Tom Tkacik, Jos Van der Woude, James R. Van Zandt, and Alex Woo. Additional substantial contributors to version 4.0 include Ethan Merritt, Petr Mikulík and Johannes Zellner.

This list was initially compiled by John Fletcher with contributions from Russell Lang, John Campbell, David Kotz, Rob Cunningham, Daniel Lewart and Alex Woo. Reworked by Thomas Koenig from a draft by Alex Woo, with corrections and additions from Alex Woo, John Campbell, Russell Lang, David Kotz and many corrections from Daniel Lewart. Again reworked for **gnuplot** 3.7 by Alexander Mai and Juergen v.Hagen with corrections by Lars Hecking, Hans-Bernhard Broecker and other people. Revised for **gnuplot** 4.0 release by Petr Mikulík and Ethan Merritt.