GNUPLOT Quick Reference

(Copyright(c) Alex Woo 1992 June 1)
(Traducció F. Udina, Agost 1995)

Arrencar GNUPLOT

per arrencar GNUPLOT	`gnuplot`
per arrencar GNUPLOT per lots	`gnuplot arxiu_ordres`
per entubar cap a GNUPLOT	`aplicació | gnuplot`

vegeu més avall variables d'entorn que potser voleu canviar abans d'entrar a GNUPLOT.

Sortir de GNUPLOT

tancar GNUPLOT	`quit`

Totes les ordres de GNUPLOT es poden abreujar a les primeres lletres que el determinen, normalment tres. Aquí utilitzem el nom complet per claretat.

Demanar ajuda

ajut introductori	`help plot`
ajut sobre un tema	`help <tema>`
llista de tot l'ajut disponible	`help o ?`
llistar l'entorn actual	`show all`

Edició de la linia d'ordres

Les versions de GNUPLOT per a UNIX, MS-DOS i VMS contenen un editor de línia i història de ordres. Es suporta un estil de edició tipus EMACS.

Line Editing:

recular un sol caràcter	`^ B`
avançar un sol caràcter	`^ F`
moure al començament de línia	`^ A`
moure al final de línia	`^ E`
esborrar el caràcter anterior	`^ H and DEL`
esborrar el caràcter següent	`^ D`
esborrar fins al final de línia	`^ K`
reescriure la línia	`^ L,^ R`
esborrar tota la línia	`^ U`
esborrar la darrera paraula	`^ W`

Història:

recular a travès de la història	`^ P`
avançar a travès de la història	`^ N`

Les tecles de fletxa segü ents es poden usar en la versió MS-DOS si s'utilitza READLINE.

Tecles de fletxa en els IBM PC:

Fletxa esquerra	`igual que ^ B`
Fletxa dreta	`igual que ^ F`
Ctrl Fletxa esquerra	`igual que ^ A`
Ctrl Fletxa dreta	`igual que ^ E`
Fletxa amunt	`igual que ^ P`
Fletxa avall	`igual que ^ N`

Dispositius gràfics

Els dispositius grà fics de sortida, també els de pantalla, s'especifiquen per un nom i unes opcions. La informació es pot llegir d'un arxiu d'arrencada (.gnuplot en UNIX, gnuplot.ini en DOS i en Mac) Si es canvia el dispositiu grà fic, cal refer el grà fic amb la ordre `replot`.

llistar els dispositius và lids	`set terminal`

Terminals grà fics:

AED 512 Terminal	`set term aed512`
AED 767 Terminal	`set term aed767`
Amiga	`set term amiga`
Adobe Illustrator 3.0 Format	`set term aifm`
Apollo graphics primitive, rescalable	`set term apollo`
Atari ST	`set term atari`
BBN Bitgraph Terminal	`set term bitgraph`
SCO CGI Driver	`set term cgi`
Apollo graphics primitive, fixed window	`set term gpr`
SGI GL windown	`set term iris4d [8 24]`
MS-DOS Kermit Tek4010 term - color	`set term kc_tek40xx`
MS-DOS Kermit Tek4010 term - mono	`set term km_tek40xx`
NeXTstep window system	`set term next`
REGIS graphics language	`set term regis`
Selanar Tek Terminal	`set term selanar`
SunView window system	`set term sun`
Tektronix 4106, 4107, 4109 & 420X	`set term tek40D10x`
Tektronix 4010; most TEK emulators	`set term tek40xx`
VAX UIS window system	`set term VMS`
VT-like tek40xx terminal emulator	`set term vttek`
UNIX plotting (not always supplied)	`set term unixplot`
AT&T 3b1 or 7300 UNIXPC	`set term unixpc`
X11 default display device	`set term x11`
X11 multicolor point default device	`set term X11`
Macintosh Graphic Window	`set term mac`
Macintosh QuickDrawGX Window	`set term gxmac`

Turbo C PC Graphics Modes:

Hercules	`set term hercules`
Color Graphics Adaptor	`set term cga`
Monochrome CGA	`set term mcga`
Extended Graphics Adaptor	`set term ega`
VGA	`set term vga`
Monochrome VGA	`set term vgamono`
Super VGA - requires SVGA driver	`set term svga`
AT&T 6300 Micro	`set term att`

MS Windows 3.x and OS/2 Presentation Manager are also supported.

Dispositius de cò pia:

Unknown - not a plotting device	`set term unknown`
Dump ASCII table of X Y [Z] values	`set term table`
printer or glass dumb terminal	`set term dumb`
Roland DXY800A plotter	`set term dxy800a`

Impresores d'agulla

Epson-style 60-dot per inch printers	`set term epson_60dpi`
Epson LX-800, Star NL-10	`set term epson_lx800`
NX-1000, PROPRINTER	`set term epson_lx800`
NEC printer CP6, Epson LQ-800	`set term nec_cp6 [monochrome color draft]`
Star Color Printer	`set term starc`
Tandy DMP-130 60-dot per inch	`set term tandy_60dpi`
Vectrix 384 & Tandy color printer	`set term vx384`

Impresores Laser

Talaris EXCL language	`set term excl`
Imagen laser printer	`set term imagen`
LN03-Plus in EGM mode	`set term ln03`
PostScript graphics language	`set term post [mode color ‘font’ size]`
CorelDraw EPS	`set term corel [mode color ‘font’ size]`
Prescribe - for the Kyocera Laser Printer	`set term prescribe`
Kyocera Laser Printer with Courier font	`set term kyo`
QMS/QUIC Laser (also Talaris 1200)	`set term qms`

Metafiles

AutoCAD DXF (120x80 default)	`set term dxf`
FIG graphics language: SunView or X	`set term fig`
FIG graphics language: Large Graph	`set term bfig`
SCO hardcopy CGI	`set term hcgi`
Frame Maker MIF 3.0	`set term mif [pentype curvetype help]`
Portable bitmap	`set term pbm [fontsize color]`
Uniplex Redwood Graphics Interface Protocol	`set term rgip`
TGIF language	`set term tgif`

Dispositius HP

HP2623A and maybe others	`set term hp2623A`
HP2648 and HP2647	`set term hp2648`
HP7580, & probably other HPs (4 pens)	`set term hp7580B`
HP7475 & lots of others (6 pens)	`set term hpgl`
HP Laserjet series II & clones	`set term hpljii [75 100 150 300]`
HP DeskJet 500	`set term hpdj [75 100 150 300]`
HP PaintJet & HP3630	`set term hppj [FNT5X9 FNT9X17 FNT13x25]`
HP laserjet III (HPGL plot vectors)	`set term pcl5 [mode font fontsize ]`

TeX picture environments

LaTeX picture environment	`set term latex`
EEPIC – extended LaTeX picture	`set term eepic`
LaTeX picture with emTeX specials	`set term emtex`
PSTricks macros for TeX or LaTeX	`set term pstricks`
TPIC specials for TeX or LaTeX	`set term tpic`
MetaFont font generation input	`set term mf`

Arxius

gràfic de les dades d’un arxiu	`plot ‘arxiu’`
carregar un arxiu d’ordres	`load ‘arxiu’`
guardar les ordres en un arxiu	`save ‘arxiu’`
guardar estat actual per posterior ú s	`save settings ‘fpec’`

ordres PLOT & SPLOT

plot i **splot** són les ordres principals per produir gràfics.

plot és per fer gràfics 2-D de funcions i dades, **splot** fa gràfics 3-D de superfícies i dades.

Sintaxi:

plot {ranges} <funcio> {titol}{estil} {,<funcio> {titol}{estil}...}

splot {ranges} <funcio> {titol}{estil} {,<funcio> {titol}{estil}...}

on <funcio> és o bé una expresió matemàtica, o bé el nom d’un arxiu de dades tancat entre cometes, o dues (per **plot**) o tres (per **splot**) en el cas de funcions paràmetriques. Les funcions o variables definides per l’usuari es poden definir aquí també.

Segueixen exemples.

Gràfics de dades

Per fer gràfics de dades que estan en un arxiu cal especificar el nom de l'arxiu (entre cometes) en la línia d'ordre de **plot** o de **splot**. Els arxius de dades han de tenir un punt per línia. Si una línia que comencen amb # (o ! en VMS) es trcacten com a comentaris, s'ignoren. Per a **plot**, cada línia de dades representa un parell (x,y). Per a **splot**, cada línia de dades representa un triplet (x,y,z). Per **plot** amb barres d'erorr (vegeu **plot errorbars**), cada línia de dades és o bé (x,y,ydelta) o bé (x,y,ybaix,yalt). En tots els casos, els nombres en cada línia de dades han d'estar separats per espai en blanc. Aquest espai en blanc divideix cada línia en columnes.

Per **plot**s es pot ometre el valor de x i per **splot**s es poden ometre els de x i y. En tots dos casos s'hi assigna el nombre de coordinada, que comença a zero i s'incrementa per cada dada.

Gràfics de superficie

Els arxius de dades per **splot** poden contenir sèries de punts, un a cada línia, que defineixen línies de nivell de la superfície. Si totes les isolínies tenen el mateix nombre de punts, s'entenen com formant una reixa, i es dibuixaran també isolínies ortogonals, la i-èsima de les quals passarà per cada i-'esim punt de les isolínies donades. Si no totes les isolínies tenen el mateix nombre de punts, no es dibuixaran aquestes isolínies auxiliars.

En **splot**, si s'utilitza una arxiu de dades 3-d i s'usa un format (vegeu **splot datafile using**) i s'especifica només la z (l'alçària), cal especificar un mode no paramètric. Si, en canvi, s'epecifiquen x, y, i z, cal triar el mode paramètric ja que les dades estan definint una superfície paramètrica.

exemple de gràfic de dades 3-d	`set parametric;splot 'glass.dat'`
exemple de gràfic explícit	`set noparametric;splot 'datafile.dat'`

Pipes

En sistemes operatius que tenen popen (UNIX), l'arxiu de dades es pot passar a través d'una ordre de *shell* si es comença el nom de l'arxiu de dades amb '<'. Per exemple:

pop(x) = 103*exp(x/10)

plot "< awk '{ print $1-1965 $2 }' population.dat", pop(x)

farà el gràfic de la mateixa niformació sobre la població però amb els anys des de 1965 en l'eix x.

De manera similar, la sortida (*output*) pot connectar-se cap a una altra applicació, per exemple,

set out "|lpr -Pmy_laser_printer"

Plot Data Using

El format de les dades que estan en un arxiu es poden llegir selectivament amb la opció **using**. Es pot utilitzar una cadena scanf, o bé es poden fer tries de columnes més senzilles.

```
plot ”datafile”      { using {<ycol> |
                     <xcol>:<ycol> |
                     <xcol>:<ycol>:<ydelta> |
                     <xcol>:<ycol>:<ylow>:<yhigh>}
                     {"<scanf string>"}}...
splot ”datafile”     { using {<xcol>:<ycol>:<zcol>}
                     {” <scanf string> ”}}...
```

<xcol>, <ycol>, i <zcol> especifiquen explícitament les columnes a utilitzar en el gràfic de les que hi en un arxiu de dades separades per espais o tabs. Si només s'especifica <ycol> en **plot**, <xcol> pren el valor 1. Si només s'especifica <zcol> en **splot**, només es llegeix aquesta columna de l'arxiu. Si <xcol> és 0 <ycol> serà dibuixat respecte el seu número.<xcol>, <ycol> i <zcol> poden ser constants o expressions.

Si s'utilitzen barres d'error en **plot**, (vegeu també **plot errorbars**), cal donar ydelta (per exemple, un error +/-) com a tercera columna, o bé ybaix i yalt com a tercera i quarta columnes. Aquestes columnes han de seguir les columnes x i y.

Les strings de scanf dominen sobre les especificacions de <xcol>:<ycol>(:<zcol>) excepte pel que fa a l'ordre.

```
plot ”datafile”      using 2:1 "%f%*f%f"
```

farà que la primera columna sigui y i la tercera x.
Si s'omet la string de scanf, els valors per omissió es generen a partir de les especificacions <xcol>:<ycol>(:<zcol>).

Si la opció **using** no s'especifica, s'usarà ”%f%f” per **plot** (”%f%f%f%f” per **plot**s amb **errorbar**) i ”%f%f%f” per **splot**.

```
plot ”MesDades”      using "%*f%f%*20[^\n]%f" w lines
```

Les dades es llegiran de l'arxiu “MesDades” a través del format ”%*f%f%*20[ˆ\n]%f”. El significat d'aquest format es: ”%*f” ignora el primer nombre, ”%f” després llegueix el segon i assigna'l a x, ”%*20[ˆ\n]” llavors ignora 20 caràcters que no siguin newline, ”%f” i finalment llegeix el valor y.

Plot With Errorbars, amb barres d'error

Es poden usar barres d'error en gràfics de dades 2-d d'arxiu, llegint-se una columna addicional que especifiqui ydelta o be dues columnes addicionals que especifiquin ybaix i yalt respectvament. No es poden usar barres d'error en **splot**s.

En la situació normal, GNUPLOT espera veure tres o quatre nombres en cada linia de l'arxiu de dades, o bé (x,y, delta) o bé (x, y, ybaix, yalt). S'ha d'especificar la coordinada x. L'ordre dels nombres ha de ser exactament el que s'ha donat aquí. Arxius de dades d'aquesta mena es poden graficar amb barres d'error fàcilment amb:

plot ”data.dat” with errorbars

La barra d'error es una linia vertical dibuixada des de (x, ylow) a (x, yhigh). Si s'especifica ydelta enlloc de ybaix i ydalt, es calculen ybaix=y-ydelta i yalt=y+ydelta. Si només hi ha dos nombres en la linia, yalt i ybaix es prenen iguals a y. Per obtenir linies que uneixin el punts de les dades, caldrà usar **plot** dues vegades, una amb barres d'error i l'altra mb linies.

Si l'escala-automàtica per y està posada, el rang de les y s'ajustarà per incloure les barres d'error.

x, y, ybaix i yalt de les columnes 1,2,3,4	`plot "data.dat" using 1:2:3:4 w errorbars`
x de la 3a, y de la 2ona, ydelta de la 6	`plot "data.dat" us 3:2:6 with errorbars`

Plot Ranges, intervals del gràfics

Els intervals opcionals especifiquen la regió del gràfic que es visualitzarà.

Els intervals es poden donar en les ordres **plot** i **splot** i afectaràn nomes a aquest gráfic, o en ordres **set xrange**, **set yrange**, etc., per canviar el intervals per omissió per gràfics futurs.

[{<var-muda>=}{<xmin>:<xmax>}]　　　`{ [{<ymin>:<ymax>}] }`

on <var-muda> és la variable independent (els valors inicials són X i y, però això es pot canviar amb **set dummy**) i els termes min i max poden ser expressions constants.

Tan els termes min com els max són opcionals. El ':' es també opcional si no es donen els termes min ni max. Això permet donar '[]' com a especificació nul.la d'interval.

En especificar un interval en la ordre **plot** treu l'escala-automàtica per l'eix en qüestió per aquell gràfic. L'ús d'una de les ordres **set -range** treu l'escala-automàtica per l'eix en qüestió per futurs gràfics, a no ser que es canvii més tard. (Vegeu **set autoscale**).

Usar els intervals actuals	`plot cos(x)`
Es dóna només l'interval x	`plot [-10:30] sin(pi*x)/(pi*x)`
Es donen tant l'interval x com y	`plot [-pi:pi] [-3:3] tan(x), 1/x`
Es dóna només l'interval y i es treu l'auto en tots dos eixos	`plot [ ] [-2:sin(5)*-8] sin(x)**besj0(x)`
Es dóna només xmax i ymin	`plot [:200] [-pi:]  exp(sin(x))`
Es donen els intervals x, y i z	`splot [0:3] [1:4] [-1:1] x*y`

Plot With Style: especificar l'estil del gràfic

El gràfics es poden fer amb un de sis estils: linies **lines**, punts **points**, linies i punts **linespoints**, impulsos **impulses**, puntets **dots**, graons **steps**, o barres d'error **errorbars**. L'estil **lines** connecta punts adjacents amb linies. L'estil **points** mostra un petit símbol en cada punt. L'estil **linespoints** fa **lines** i **points**. L'estil **impulses** dibuixa una linia vertical des de l'eix x (o des de la reixa base en cas de **splot**) **dots** dibuixar un petit puntet en cada punt, cosa que és util per diagrames de dispersió amb molts punts.

L'estil **errorbars** només té sentit per gràfics de dades 2-d. Es tracta com **points** en **splot**s i en **plot**s de funcions. Per **plot**s de dades, **errorbars** és com **points** però s'hi dibuixa una barra d'error: per cada punt (x,y) es dibuixa una línia des de (x,ybaix) fins a (x,yalt). Es dibuixa una ratlleta als extrems de les barres d'error. Els valors ybaix i yalt es llegeixen de les columnes de l'arxiu de dades segons s'especifiqui a **plot** en la opció **using**. Vegeu **plot errorbars** per més informació.

Els estils per omissió es determinen amb les ordres **set function style** i **set data style**.

Per omissió, cada funció i cada arxiu de dades usarà un tipus de línia i de punts diferent, fins al màxim nombre de tipus disponibles. Tots els drivers de terminals suporten com a mínim sis tipus diferents de punts i els tornen a utilitzar en ordre si en calen més de sis. El driver de LaTeX proporciona sis tipus de punts adicionals (totes les variants d'un cercle) i per tant només repetirà després de dibuixar dotze corbes amb punts.

Si es desitja es pot especificar el tipus de línia i (opcionalment) el de punt per una corba.

with <estil>	{`<tipuslinia>` {`<tipuspunt>`}}

on <estil> és un de **lines**, **points**, **linespoints**, **impulses**, **dots**, **steps**, or **errorbars**. <tipuslinia> i <pointtype> són constants o expressions de valor enter positiu i especifiquen el tipus de linia i de punt que es vol en el gràfic. El tipus de linia 1 és el primer que s'utilitza per omissió, el 2 és el segon que s'utilitza per omissió, etc

gràfic de sin(x) amb impulsos	`plot sin(x) with impulses`
x*y amb punts, x**2 + y**2 per omissió	`splot x*y w points, x**2 + y**2`
gràfic de tan(x) amb l'estil per omissió de funcions	`plot [ ] [-2:5] tan(x)`
gràfic de l'arxiu "data.1" amb linies	`plot "data.1" with l`
gràfic de "leastsq.dat" amb impulsos	`plot 'leastsq.dat' w i`
gràfic de "exper.dat" amb barres d'error i amb linies unint els punts	`plot 'exper.dat' w l, 'exper.dat' w err`

Aquí 'exper.dat' hauria de tenir tres o quatre columnes de dades.

gr'afic de x**2 + y**2 i x**2 - y**2 amb igual tipus de l'ınia	`splot x**2 + y**2 w l 1,\` `x**2 - y**2 w l 1`
gràfic de sin(x) i cos(x) amb punts-i-linies, amb el mateix tipus de linia però diferents punts	`plot sin(x) w linesp 1 3, \` `cos(x) w linesp 1 4`
gràf. arxiu "data" amb estil de punts 3	`plot "data" with points 1 3`

Observi's que cal especificar l'estil de línia per donar un estil de punt, encara que sigui irrelevant. Aquí l'estil de lína és 1 i el de punt és 3, i l'estil de linia és irrelevant.

Vegeu **set style** per canviar els estils per omissió.

Plot Title, títols de les corbes

En la llegenda pot aparèixer un títol per cada corba. Per omissió el títol és la funció o el nom d'arxiu tal com figuren en la ordre de **plot**. Es pot canviar el títol amb la opció **title**. Aquesta opció ha de anar devant de qualsevol opció **with**.

title "<titol>"

on <title> és el nou títol per a la corba, i ha d'anar tancada entre cometes. Les cometes no apareixeràn en la llegenda.

gràf. de y=x amb el títol 'x'	`plot x`
gràf de l'arxiu "glass.dat" amb el títol 'sup. revolució'	`splot "glass.dat" tit 'sup.  revolucio'`
plots x al quadrat amb el títol "xˆ2" i "data.1" amb títol 'dades obtingudes'	`plot x**2 t "xˆ2", \` `"data.1" t 'dades obtingudes'`

Les ordres Set-Show

Totes les ordres d'aquí comencen amb **set** o amb **show** per establir o veure l'estat de les variables internes que es llisten a la dreta. Per més detalls, **help set** *var*.

Els claudàtors denoten opcionalitat.

mesura dels angles en polars	angles [degrees|radians]
fletxa des de	arrow [<tag>][from <sx>,<sy>,<sz>]
fins a	[to <ex>,<ey>,<ez>][nohead]
forçar escala automàtica	autoscale [<axes>]
modus paramètric	[no]parametric
dibuixar vora	[no]border
retallar punts/línies prop de vores	[no]clip <clip-type>
parametres per gràf de contorns	cntrparam [spline][points][order][levels]
contorns en gràfic splot	[no]contour [base|surface|both]
estil de gràf per omissió per dades	data style <style-choice>
especificar nom de variable per gràfs	dummy <dummy1>,<dummy2>...
format per les etiquetes de les marques en eixos	format [<axes>]["format-string"]
estil per omissió per funcions	function style <style-choice>
dibuixar una reixa en les marques d'eixos	[no]grid
supressió de línies ocultes	[no]hidden3d
nombre de isolínies	isosamples <expression>
llegenda en el gràfic	key <x>,<y>,<z>
escala logarítm. en eixos, base opcional	[no]logscale <axes> [<base>]
sistema de coordenades 3-D	mapping [cartesian|spherical|cylindrical]
marges a les vores	offsets <left>,<right>,<top>,<bottom>
coordenades 2D	[no]polar
interval per radis	rrange [<rmin>:<rmax>]
nombre de punts a computar per funcions	samples <expression>
factors d'escala del gràfic	size <xsize>,<ysize>
dibuixar contorns a la superfície	[no]surface
dispositiu de gràfics	terminal <device>
direcció de les marques en eixos	tics <direction>
ajustar alçada relativa de l'eix vertical	ticslevel <level>
posar data i hora	[no]time
posar títol centrat al gràf.	title "title-text" [<xoff>,<yoff>]
interval pel parametre de corbes	trange [<tmin>:<tmax>]
intervals pels parametres de superf.	urange or vrange
el punt de vista per **splot**	view <rot_x>,[<rot_z>,<scale>,<scale_z>]
etiqueta per l'eix x	xlabel "<label>" <xoff>,<yoff>
interval horitzontal	xrange [<xmin>:<xmax>]
marques en l'eix x	xtics <start>,<incr>,<end>, "<label>" <pos>
dibuixar eix x	[no]xzeroaxis
etiqueta per eix y	ylabel "<label>" <xoff>,<yoff>
interval per eix y	yrange [<ymin>:<ymax>]
marques eix y	ytics <start>,<incr>,<end>, "<label>" <pos>
dibuixar eix y	[no]yzeroaxis
dintell per omissió per valors prop de 0	zero <expression>
dibuixar eixos	[no]zeroaxis
etiqueta per eix z	zlabel "<label>" <xoff>,<yoff>
interval eix z	zrange [<zmin>:<zmax>]
marques eix z	ztics <start>,<incr>,<end>, "<label>" <pos>
dibuixar eix z	[no]zzeroaxis

Contour Plots, gràfics de nivells o de contorns

set contour activa el dibuix de nivells en superfícies. Aquesta opció està disponble només per **splot**.

Sintaxi:

set contour { base | surface | both }

set nocontour

Si no es dóna cap opció per **set contour**, el valor inicial és **base**. Les tres opcions especifiquen on s'han de dibuixar els contorns: **base** dibuixa els contorns a la reixa de la base, on estan les marques x/y, **surface** els dibuixa sobre la pròpia superfície, i **both** dibuixa els contorns tan a la base com a la superfície.

Vegeu també **set cntrparam** pel control del paràmetres que afecten el dibuix dels contorns.

Contour Parameters, els paràmetres del contorns

set cntrparam controls els diferents paràmetres del gràfic de contorns o de nivells. Vegeu també **set contour**.

set cntrparam	`{{ linear | cubicspline | bspline }|`
	`points <n> |`
	`order <n> |`
	`levels { [ auto ] <n> |`
	`discrete <z1> <z2> ... |`
	`incr <inici> <increment> [ <n> ] }}`

5 nivells automàtics	`set cntrparam levels auto 5`
3 nivells discrets a 10%, 37% i 90%	`set cntrp levels discrete .1 1/exp(1) .9`
5 nivells progressius a 0, .1, .2, .3 i .4	`set cntrparam levels incremental 0 .1 5`
posar n = 10 mantenint auto, incr., o discr. com estan	`set cntrparam levels 10`
posar inini = 100 i increment = 50, amb el vell n	`set cntrparam levels incremental 100 50`

Aquesta ordre controla la manera en què els contorns es dibuixen. <n> ha de ser una expressió constant de valor enter i <z1>, <z2> expressions constants qualsevols. Els paràmetres són:

linear, **cubicspline**, **bspline** - Controlen el tipus d'aproximació o d'interpolació. Amb **linear**, els contorns es dibuixen amb trossos de línies extrets directament de la superfície. Amb **cubicspline** els contorns s'interpolen per formar contorns més llisos, que poden ser ondulats. La tercera opció és **bspline**, que només aproxima la línia poligonal i es garanteix que és llisa.

points - Aquest nombre controla el nombre de punts que s'utilitzen per aproximar la corba. És relevant només per **cubicspline** i **bspline**.

order - És l'ordre del b-spline a utilitzar. Com més gran, més llis serà el contorn resultant (naturalment, l'aproximació serà també menys acurada respecte l poligonal original Aquesta opció només és relevant pel mode **bspline**. Els valors permesos són enters des de 2 (lineal) fins a 10.

levels - Nombre de nivells, 'n'. La selecció dels nivells pot ser 'auto', 'discrete' i 'incremental'. Per 'auto', si la superficie està acotada per zmin i zmax, es generaran contorns des de zmin+dz fins a zmax-dz en salts de mida dz, on dz = (zmax - zmin) / (nivells + 1). Per 'discrete', els contorns es generaran per z= z1, z2 ... tal com s'especifiquin. La quantitat de nivells discrets està limitat a MAX_DISCRETE_LEVELS, definit a 30 en plot.h. En 'incremental', es generen contorns per <n> valors de z començant a <inici> i augmentant en <increment>.

Posant rètols

Es poden posar rètols arbitraris amb la ordre **set label**.

set label {<num>}{" <text-del-retol> "}	{at <x>,<y> {,<z>}} {<justificacio>}

set nolabel {<num>}
show label

El text per omissió és "", i la posició, 0,0,0. Els valors <x>, <y>, i <z> estan en el sistema de coordenades del gràfic. Si es dóna la coordenada z en un **plot** és ignorada; si falta en un **splot**, es prendrà com a 0. El num es un enter que s'utilitza per identificar el rètol. Si no es dóna cap <num>, s'utilitza automàticament el més petit dels disponibles. El num es pot utilitzar per esborrar o canviar un rètol específic. Per canviar algun atribut d'un rètol existent, cal usar la ordre **set label** amb el num adequat i especificar les parts del rètol que cal canviar.

Per omissió el text es col.loca arrenglerat a l'esquerra contra el punt x,y,z. Per ajustar la manera com el rètol es col.loca respecte el punt x,y,z, es pot afegir el paràmetre <justificacio>, que pot ser **left**, **right** or **center**, indicant que el punt ha d'estar a l'esquerra, dreta o centre del text. S'accepten els rètols fora dels límits del gràfic però poden interferir amb les etiquetes dels eixos o altre text.

rètol en (1,2) amb "y=x"	`set label "y=x" at 1,2`
rètol "y=x^2" amb el punt (2,3,4) a la dreta i assignar-li el num 3	`set label 3 "y=x^2" at 2,3,4 right`
canviar el rètol anterior a justificació centrada	`set label 3 center`
eliminar rètol número 2	`set nolabel 2`
eliminar tots els rètols	`set nolabel`
llistar tots els rètols (en ordre)	`show label`

(Els drivers EEPIC, Imagen, LaTeX, i TPIC drivers admeten \\ en el text per especificar canvi de línia.)

Ordres diverses

Per més informació sobre aquesta ordre, consulteu el manual de GNUPLOT o feu **help** ...

canviar el directori de treball	`cd`
esborrar pantalla o dispositiu actual	`clear`
sortir de GNUPLOT	`exit or quit or EOF`
mostrar un text i esperar	`pause <temps-en-segons> ["<text>"]`
escriure el valor de <expressio>	`print <expressio>`
escriure el directori actual	`pwd`
repetir el darrer **plot** o **splot**	`replot`
obrir una shell interactiva	`!  (UNIX) o $ (VMS)`

Environment Variables

Tot i que no són imprescindibles, hi ha diverses variables d'entorn que GNUPLOT pot entendre i que poden ser utils quan s'utilitza GNUPLOT en sistemes operatius Unix, DOS, o VMS.

Si GNUTERM està definit, es pren com el nom del tipus de terminal a utilitzar. Això sobrepassa el tipus de terminal que GNUPLOT troba normal utilitzar en arrencar, però queda superat per la especificació de l'arxiu d'arrencada .gnuplot o equivalent (vegeu **star-up**) i, per suposat per posteriors canvis explícits.

En Unix, AmigaDOS, and MS-DOS, GNUHELP es pot definir com el camí d'accés de l'arxiu d'ajuda (gnuplot.gih).

En VMS, el símbol GNUPLOT$HELP es pot definir com el nom de la llibreria d'ajut per GNUPLOT.

En Unix, HOME s'utilitza com el nom d'un directori on buscar l'arxiu .gnuplot, si no se n'ha trobat cap en el directori actual. En AmigaDOS i MS-DOS, s'utilitza GNUPLOT. i en VMS, SYS$LOGIN:. Vegeu help start-up.

On Unix, PAGER is used as an output filter for help messages.

On Unix and AmigaDOS, SHELL is used for the **shell** command. On MS-DOS, COMSPEC is used for the **shell** command.

On AmigaDOS, GNUFONT is used for the screen font. For example: "setenv GNUFONT sapphire/14".

On MS-DOS, if the BGI interface is used, the variable **BGI** is used to point to the full path to the BGI drivers directory. Furthermore SVGA is used to name the Super VGA BGI driver in 800x600 res., and its mode of operation as 'Name.Mode'. For example, if the Super VGA driver is C:\TC\BGI\SVGADRV.BGI and mode 3 is used for 800x600 res., then: 'set BGI=C:\TC\BGI' and 'set SVGA=SVGADRV.3'.

Expressions

In general, any mathematical expression accepted by C, FORTRAN, Pascal, or BASIC is valid. The precedence of these operators is determined by the specifications of the C programming language. White space (spaces and tabs) is ignored inside expressions.

Complex constants may be expressed as {<real>,<imag>}, where <real> and <imag> must be numerical constants. For example, $\{3,2\}$ represents $3 + 2\mathbf{i}$ and $\{0,1\}$ represents $\mathbf{i}$ itself. The curly braces are explicitly required here.

Functions

The functions in GNUPLOT are the same as the corresponding functions in the Unix math library, except that all functions accept integer, real, and complex arguments, unless otherwise noted. The **sgn** function is also supported, as in BASIC.

Function	Arguments	Returns
abs(x)	any	absolute value of x, $\|x\|$; same type
abs(x)	complex	length of x, $\sqrt{\mathrm{real}(x)\hat{}2+\mathrm{imag}(x)\hat{}2}$
acos(x)	any	$\cos\hat{}-1x$ (inverse cosine) in radians
arg(x)	complex	the phase of x in radians
asin(x)	any	$\sin\hat{}-1x$ (inverse sin) in radians
atan(x)	any	$\tan\hat{}-1x$ (inverse tangent) in radians
besj0(x)	radians	j_0 Bessel function of x
besj1(x)	radians	j_1 Bessel function of x
besy0(x)	radians	y_0 Bessel function of x
besy1(x)	radians	y_1 Bessel function of x
ceil(x)	any	$\lceil x \rceil$, smallest integer not less than x (real part)
cos(x)	radians	$\cos x$, cosine of x
cosh(x)	radians	$\cosh x$, hyperbolic cosine of x
erf(x)	any	Erf(real(x)), error function of real(x)
erfc(x)	any	Erfc(real(x)), 1.0 - error function of real(x)
exp(x)	any	$e\hat{}x$, exponential function of x
floor(x)	any	$\lfloor x \rfloor$, largest integer not greater than x (real part)
gamma(x)	any	Gamma(real(x)), gamma function of real(x)
ibeta(p,q,x)	any	Ibeta(real(p,q,x)), ibeta function of real(p,q,x)
igamma(a,x)	any	Igamma(real(a,x)), igamma function of real(a,x)
imag(x)	complex	imaginary part of x as a real number
int(x)	real	integer part of x, truncated toward zero
lgamma(x)	any	Lgamma(real(x)), lgamma function of real(x)
log(x)	any	$\log\_ex$, natural logarithm (base e) of x
log10(x)	any	$\log\_10x$, logarithm (base 10) of x
rand(x)	any	Rand(real(x)), pseudo random number generator
real(x)	any	real part of x
sgn(x)	any	1 if $x > 0$, -1 if $x < 0$, 0 if $x = 0$. imag(x) ignored
sin(x)	radians	$\sin x$, sine of x
sinh(x)	radians	$\sinh x$, hyperbolic sine x
sqrt(x)	any	$\sqrt{x}$, square root of x
tan(x)	radians	$\tan x$, tangent of x
tanh(x)	radians	$\tanh x$, hyperbolic tangent of x

Operators

The operators in GNUPLOT are the same as the corresponding operators in the C programming language, except that all operators accept integer, real, and complex arguments, unless otherwise noted. The ** operator (exponentiation) is supported, as in FORTRAN.

Parentheses may be used to change order of evaluation.