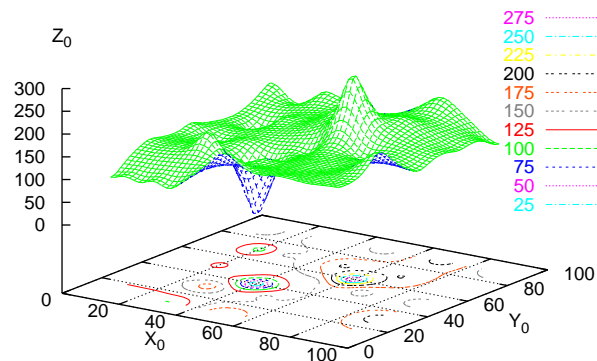


---

Universidade do Estado de Santa Catarina  
Centro de Ciências Tecnológicas  
Departamento de Física

## Introdução ao GnuPlot



Cleber Angelo Capellari  
Joinville, 13 de maio de 2003.

---

## Sumário

<b>1</b>	<b>Introdução</b>	<b>2</b>
<b>2</b>	<b>Comandos básicos</b>	<b>2</b>
<b>3</b>	<b>Elaborando um "script"</b>	<b>3</b>
<b>4</b>	<b>Lendo arquivo de dados</b>	<b>4</b>
<b>5</b>	<b>Ajustamento de curvas</b>	<b>7</b>
<b>6</b>	<b>Exemplos de "scripts"</b>	<b>8</b>
<b>7</b>	<b>Conclusão</b>	<b>10</b>

# 1 Introdução

GnuPlot é um programa de distribuição gratuita disponível para diversos sistemas operacionais (Windows, Linux, etc). Dentre as vantagens em sua utilização podemos citar a portabilidade e embora não pareça a princípio, a facilidade no seu manuseio. O mesmo se dá de forma interativa através da linha de comando ou elaborando um "script", contendo todas as instruções a serem executadas. O objetivo desse tutorial é dar ao usuário noções básicas de utilização do GnuPlot e condições, para que o mesmo possa prosseguir na sua utilização.

## 2 Comandos básicos

Começaremos executando o programa GnuPlot, partindo do pressuposto que os usuários tenham noções básicas de utilização do computador. Aqui, em específico, utilizarei Linux. Veremos de início um "prompt" onde aparece

```
gnuplot>
```

Façamos um teste inicial, digitando o comando de plotagem da função  $\sin(x)$ , já aproveitando para informar que o gnuplot tem as variáveis "x" e "y" definidas como padrão.

```
gnuplot> plot sin(x) <ENTER>
```

Veremos que abrir-se-á uma nova janela contendo o gráfico solicitado. Podemos ajustar a escala do eixo "x" simplesmente digitando

```
gnuplot> set xrange [-3.14:3.14]
```

Muitos comandos podem ser abreviados, então o comando seguinte surtirá o mesmo efeito

```
gnuplot> set xr [-3.14:3.14]
```

Plotamos novamente o gráfico utilizando o comando

```
gnuplot> replot
```

Prosseguindo, podemos incluir uma grade na janela do gráfico simplesmente digitando

```
gnuplot> set grid  
gnuplot> replot
```

Observe que no canto superior direito há uma legenda, que podemos retirar digitando

```
gnuplot> set nokey  
gnuplot> replot
```

Para adicionarmos um título ao gráfico,

```
gnuplot> set title "sen(x)"  
gnuplot> replot
```

Ajustaremos as legendas dos eixos digitando

```
gnuplot> set ylabel "y"  
gnuplot> set xlabel "x"  
gnuplot> replot
```

Podemos ainda ajustar o tamanho do gráfico com

```
gnuplot> set size 0.7,0.7
```

Para salvar o gráfico, digitamos

```
gnuplot> set term png small color
gnuplot> set out 'teste00.png'
gnuplot> replot
```

onde o formato escolhido foi png (Portable Network Graphics). Podemos ver uma lista dos formatos possíveis digitando

```
gnuplot> set term <ENTER>
```

O resultado final será a figura 1. Passo seguinte, veremos como elaborar um "script" contendo todos esses comandos para que sejam lidos de uma só vez.

### 3 Elaborando um "script"

Num editor de textos qualquer, digitaremos todas as instruções que utilizamos na "linha de comando", salvando no diretório de sua preferência com o nome "teste01.gnu". Para navegar pelos diretórios

```
gnuplot> pwd
```

Isso dará a sua localização atual. Para ir ao diretório desejado

```
gnuplot> cd '<CAMINHO>'
```

O arquivo deverá ficar assim

```
#Instr. para plotar sen(x)
#
reset
set xrange [-3.14:3.14]
set grid
set nokey
set title "sen(x)"
set ylabel "y"
set xlabel "x"
set size 0.7,0.7
set term png small color
set out 'teste01.png'
plot sin(x)
#
#Fim
```

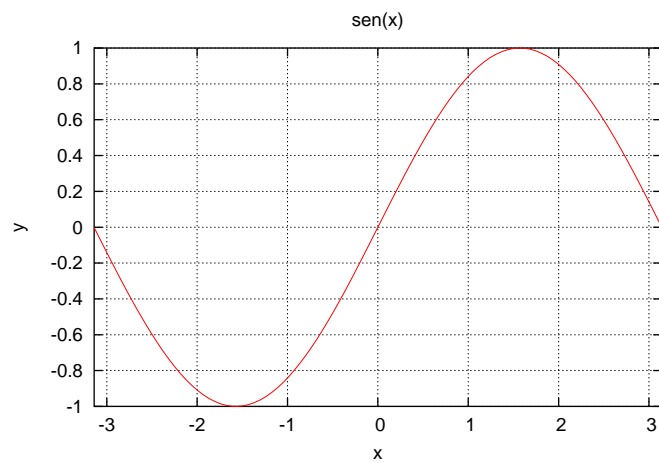
Demais instruções também podem ser adicionadas ao "script". Executamos o "script" com

```
gnuplot> load 'teste01.gnu'
```

Como resultado obteremos a figura 1.

Percebemos assim, que um "script" padrão pode ser criado e modificado conforme eventuais necessidades. A partir de agora, passaremos a utilizar sempre "scripts" para executarmos nossos comandos. Próximo passo, veremos como ler um arquivo de dados através de um "script".

Figura 1: teste01.eps



## 4 Lendo arquivo de dados

Observemos o arquivo de dados a seguir

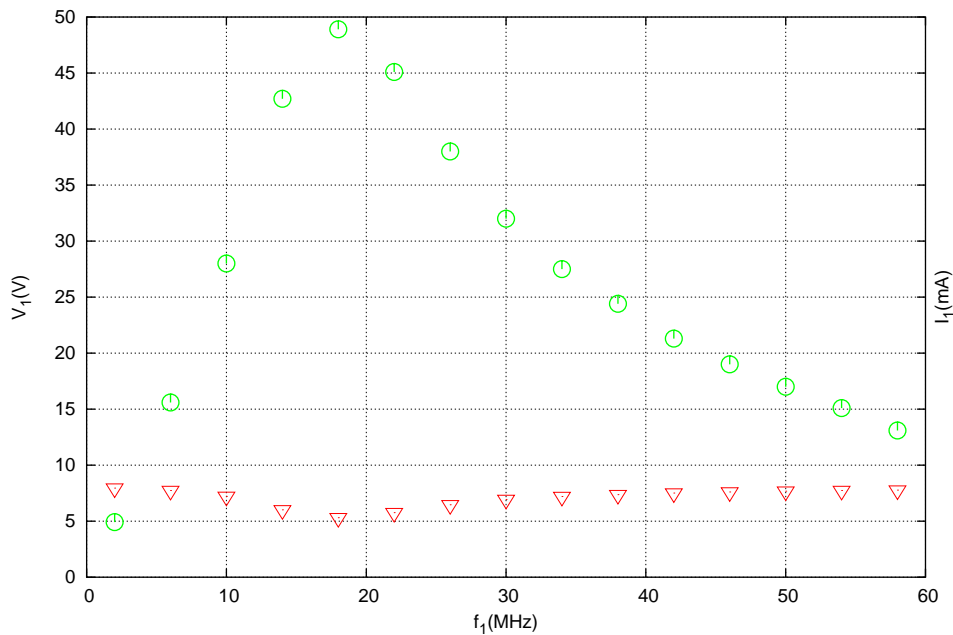
```
#Dados para o script teste02.gnu
#
#Freq.      Corrente      Tensão
02          7.95          4.92
06          7.74          15.6
10          7.21          28.0
14          6.01          42.7
18          5.31          48.9
22          5.76          45.1
26          6.46          38.0
30          6.94          32.0
34          7.20          27.5
38          7.37          24.4
42          7.51          21.3
46          7.61          19.0
50          7.68          17.0
54          7.74          15.1
58          7.77          13.1
#
#Fim
```

São dados obtidos de uma experiência envolvendo circuitos RLC. Agora, o "script" que irá ler os pontos

```
#Script para o arquivo teste02.dat
#
reset
set title " "
set xlabel "f_{1}(MHz)"
set ylabel "V_{1}(V)"
set y2label "I_{1}(mA)"
set nokey
set grid
set term post eps enhanced color
set out 'teste02.eps'
plot 'teste02.dat' u ($1):($2) w points pt 10 ps 2,\
      'teste02.dat' u ($1):($3) w points pt 16 ps 2
#
#Fim
```

Observemos que agora utilizamos outro formato para salvar o gráfico. A opção "post eps enhanced color" permite salvarmos gráficos no formato eps (Encapsulated Postscript), bem como escrevermos caracteres sobrescritos e subscritos. Esse formato é muito útil para inclusão em textos utilizando "LaTeX". Vejamos nosso resultado na figura 2.

Figura 2: teste02.eps



Da mesma forma podemos plotar superfícies. Vejamos um arquivo de dados

```
#Dados para o script teste03.gnu
#
#X      #Y      #Z
10      10      100
10      30      150
10      50      150
10      70      130
10      90      170
20      10      100
20      30      150
20      50      120
20      70      093
20      90      135
30      10      099
30      30      189
30      50      138
30      70      152
30      90      174
40      40      000
50      10      200
50      30      153
50      50      128
50      70      203
50      90      142
60      60      300
70      10      173
70      30      157
70      50      142
70      70      201
```

```

70      90      210
90      10      173
90      30      216
90      50      142
90      70      152
90      90      132
#
#Fim

```

O "script" para plotar os dados

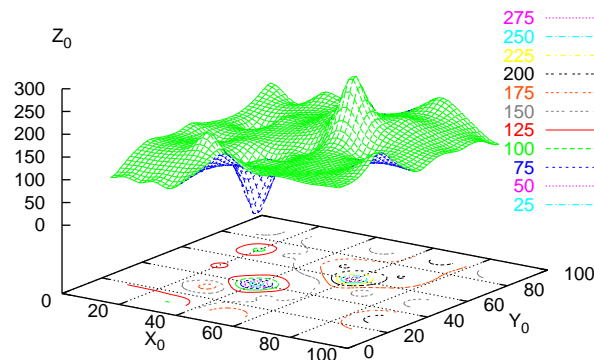
```

#Script para o arquivo teste03.dat
#
#Comandos preliminares: grid, rótulos e título
#
reset
set grid
set xtics 020
set ytics 020
set ztics 050
set title ""
set xlabel "X_{0}"
set ylabel "Y_{0}"
set zlabel "Z_{0}"
#
#Comandos específicos para visualização 3d
#
set hidden3d
set view 65,35,1,1
set data style points
set contour base
set cntrparam levels incremental 0,25,300;
set dgrid3d 50,50,2
#
#Comandos para plotar e salvar gráfico
#
set size .7, .7
set term post eps enhanced color
set out 'teste03.eps'
splot 'teste03.dat' u ($1):($2):($3) t"" w lines lt 2
#
#Fim

```

Veamos o resultado na figura 3

Figura 3: teste03.eps



Verificamos assim que, dados retirados de experimentos (ou de qualquer outra fonte) podem ser facilmente plotados através do GnuPlot. A seguir, veremos como proceder para o ajustamento de curvas.

## 5 Ajustamento de curvas

Vejamos agora o arquivo de dados teste04.dat a seguir

```
#Dados para o script teste04.gnu
#
#X  #Y
20  19,2
50  28,4
70  34,8
80  38,0
100 43,6
120 50,3
130 53,3
150 59,4
170 65,2
200 74,5
#
#Fim
```

O "script" teste04.gnu, para fazer o ajustamento e plotar os dados segue

```
#Script para o arquivo teste04.dat
#
reset
set nokey
set grid
set title " "
set xlabel "m_{0}(g)"
set ylabel "x_{eq}10^{-2}(m)"
f(x)=a*x+b
set size 0.7,0.7
set terminal post eps enhanced color
set out 'teste04.eps'
fit f(x) 'teste04.dat' via a,b
plot 'teste04.dat' u ($1):($2) w points pt 1 ps 3, f(x) lt 3 lw 3
#
#Fim
```

O arquivo de "log" do ajustamento pode ser encontrado no mesmo diretório do "script" com o nome "fit.log".

```
*****
Wed Apr 16 19:50:50 2003

FIT:   data read from 'teste04.dat'
      #datapoints = 10
      residuals are weighted equally (unit weight)

function used for fitting: f(x)
fitted parameters initialized with current variable values

Iteration 0
WSSR      : 54069          delta(WSSR)/WSSR   : 0
delta(WSSR) : 0           limit for stopping : 1e-05
lambda    : 85.7059

initial set of free parameter values
```

```
a      = 1
b      = 1
```

```
After 6 iterations the fit converged.
final sum of squares of residuals : 1.0324
rel. change during last iteration : -4.02624e-13
```

```
degrees of freedom (ndf) : 8
rms of residuals      (stdfit) = sqrt(WSSR/ndf)      : 0.359235
variance of residuals (reduced chisquare) = WSSR/ndf : 0.129049
```

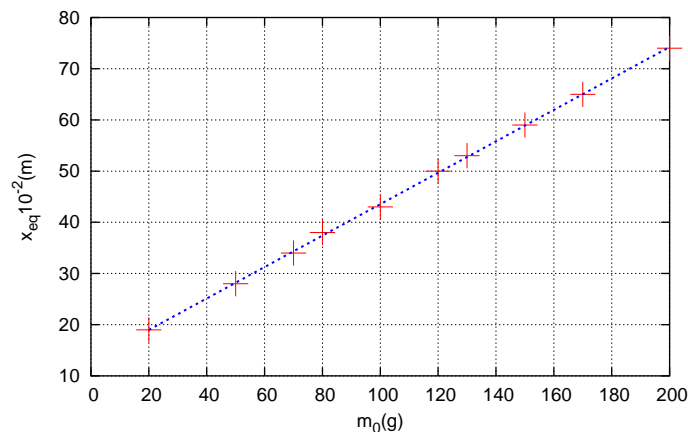
Final set of parameters	Asymptotic Standard Error		
=====	=====		
a	= 0.306978	+/- 0.002143	(0.6982%)
b	= 12.8394	+/- 0.2598	(2.023%)

```
correlation matrix of the fit parameters:
```

	a	b
a	1.000	
b	-0.899	1.000

No mesmo diretório encontraremos o arquivo de saída (figura 4).

Figura 4: teste04.eps



## 6 Exemplos de "scripts"

Seguem alguns exemplos de scripts com os respectivos arquivos de dados. Ajustamento de pontos através de equação quadrática.

```
#Script para o arquivo teste05.dat
#
reset
set title ""
set xlabel "l(nm)"
set ylabel "n"
set grid
set yrange [1.65:*
```

```
set ytics 0.01
set format x "%g"
set format y "%1.2f"
set nokey
f(x) =a+b*x+c*x*x
fit f(x) 'graph05.dat' u ($1):($2) via a, b, c
set term post eps enhanced color
set out 'teste05.eps'
plot "teste05.dat" u ($1):($2) w points pt 9 ps 3, f(x)
#
#Fim

#Dados para o script teste05.gnu
#
645  1.6632
517  1.6575
509  1.6758
460  1.6791
469  1.6828
441  1.7018
#
#Fim
```

Exemplo de ajustamento linear.

```
#Script para o arquivo teste06.dat
#
reset
set title ""
set xlabel "l^{-2}(nm)"
set ylabel "n"
set grid
set format x "%g"
set format y "%1.3f"
set nokey
set term post eps enhanced color
set out 'teste06.eps'
f(x)=a+b*x
fit f(x) 'teste06.dat' u 1:2 via a, b
plot "teste06.dat" u 1:2 w points pt 9 ps 2, f(x)
#
#Fim

#Dados para o script teste06.gnu
#
645  0.3615
517  0.3640
509  0.3561
460  0.3547
469  0.3531
441  0.3453
#
#Fim
```

Ajustamento de pontos no plano. Perceba no arquivo de dados uma coluna relativa ao erro.

```
#Script para o arquivo teste07.dat
#
reset
set title " "
set grid
set xlabel "x"
set ylabel "y"
set zlabel "z"
set xtics 0.02
set format x "%1.2f"
```

```

set format y "%1.1f"
set format z "%2.1f"
#set term post eps enhanced color
#set out 'teste07.eps'
#
set view 65, 35, 1, 1
f(x,y)=a+b*x+c*y
fit f(x,y) 'teste07.dat' u ($1):($2):($3):($4) via a,b,c
splot f(x,y) lt 3, 'teste07.dat' w points pt 9 ps 2
#
#Fim

#Dados para o script teste07.gnu
#
#X      #Y      #Z      #Erro
4.95    5      24.9873  0.544021
5.00    6      26.6115  0.999990
5.05    7      28.5612  0.536573
#
#Fim

```

Ajustamento através de uma função tangente hiperbólica.

```

#Script para o arquivo teste08.dat
#
reset
set title "Dados da força de deflecção".
set xlabel "Deflecção (m)"
set ylabel "Força (kN)"
set xr [0.001:0.005]
set yr [20:500]
set grid
set size 0.8,0.8
f1(x) = a1*tanh(x/b1)
a1 = 300
b1 = 0.005
fit f1(x) 'teste08.dat' u ($1):($2) via a1, b1
set term post eps enhanced color
set out 'teste08.eps'
plot 'teste08.dat' u ($1):($2) with points 1 ps 3,\
      'teste08.dat' u ($1):($3) with points 2 ps 3,\
      a1*tanh(x/b1)
#
#Fim

#Dados para o script teste08.gnu
#
#Deflecção          #Força elementar      #Força de raio
0.0000              000                    000
0.0010              104                    051
0.0020              202                    101
0.0030              298                    148
0.0031              290                    149
0.0040              289                    201
0.0041              291                    209
0.0050              310                    250
0.0100              311                    260
0.0200              280                    240
#
#Fim

```

## 7 Conclusão

Com o exposto, temos boa bagagem de instruções a respeito do programa GnuPlot. Usuários interessados em prosseguir na sua utilização, podem encontrar maiores informações dentro do próprio programa, simplesmente digitando

```
gnuplot> help
```

Outra fonte de documentação é a própria rede mundial de computadores. Esperamos ter alcançado o nosso objetivo. Que esse material seja de bom proveito, servindo também para futuras referências.