



COMPUTAÇÃO GRÁFICA

Prática 2 – Funções Gráficas (Matlab)

Ivan Nunes da Silva



Funções Gráficas Básicas

- Objetivos da Aula:
 - ♦ Demonstrar as funções elementares que são utilizadas pelo Matlab para a plotagem gráfica.
 - ♦ Destacar as potencialidades das funções do Matlab para propósitos de computação gráfica.
 - ♦ Entender a anatomia de uma plotagem gráfica em duas dimensões.
 - ♦ Como escolher diferentes tipos de plotagens visando um melhor efeito.
 - ♦ Realizar exercícios visando a familiarização das funções gráficas elementares.



Funções Gráficas Elementares

- O Matlab dispõe de 6 funções básicas para manipulação gráfica em duas dimensões, sendo diferenciadas principalmente pelo tipo de escala que utilizam nos eixos.
 - ♦ **plot(.)** → cria um gráfico a partir de vetores colunas de matrizes com escalas lineares sobre ambos eixos.
 - ♦ **loglog(.)** → cria um gráfico a partir de vetores colunas de matrizes com escalas logarítmicas sobre ambos eixos.
 - ♦ **semilogx(.)** → cria um gráfico a partir de vetores colunas de matrizes com escala linear no eixo das ordenadas e logarítmicas no eixo das abscissas.
 - ♦ **semilogy(.)** → cria um gráfico a partir de vetores colunas de matrizes com escala logarítmica no eixo das ordenadas e linear no eixo das abscissas.
 - ♦ **plotyy(.)** → gráfico com duas escalas na vertical (uma à esquerda e outra à direita)
 - ♦ **polar(.)** → gráfico em coordenadas polares.

3



Plotagem Gráfica (I)

- O Matlab dispõe de diversas outras funções básicas que permitam manipular gráficos de forma eficiente.

Símbolo	Cor	Símbolo	Estilo de Marcador	Símbolo	Estilo de Linha
y	amarelo	.	ponto	-	sólida
m	magenta	o	círculo	:	pontilhada
c	ciano	x	x	-.	traço e ponto
r	vermelho	+	sinal positivo	--	tracejada
g	verde	*	estrela		
b	azul	s	quadrado		
w	branco	d	losango		
k	preto	^	triângulo p/cima		

4



Plotagem Gráfica (II)

- As funções básicas `plot()`, `loglog()`, `semilogx()` e `semilogy()` podem ser configuradas da seguinte forma:
 - ♦ **`plot(y)`** → plota os valores de y no eixo das ordenadas sendo que o eixo x é incrementado de uma unidade para cada valor de y.
 - ♦ **`plot(x,y)`** → plota o valor de x no eixo das abscissas e y no eixo das ordenadas.
 - ♦ **`plot(x,y,'r')`** → plota o gráfico em cor vermelha referente aos valores de x e y.
 - ♦ **`plot(x,y,'b:d')`** → plota o gráfico em cor azul, estilo de linha pontilhada e estilo de marcador em losango.
 - ♦ **`plot(x1,y1,s1,x2,y2,s2,x3,y3,s3,...)`**
 - (x_n,y_n) são conjuntos de dados e s_n são strings de caracteres opcionais especificando cor, símbolos marcadores e/ou estilos de linha.

5



Manipulação Gráfica (I)

- Diversas outras funções básicas são definidas no Matlab visando a manipulação gráfica:
 - ♦ **`linspace(valorinicial, valorfinal, númeropontos)`**
 - Especifica diretamente o espaçamento entre os pontos entre o valor_{inicial} e o valor_{final} a fim de obter o número_{pontos} previamente especificado.
- Manipulação de textos em gráficos:
 - ♦ **`title('título')`** → insere o título no gráfico.
 - ♦ **`xlabel('label')`** → insere legenda no eixo x.
 - ♦ **`ylabel('label')`** → insere legenda no eixo y.
 - ♦ **`legend('leg1','leg2',...)`** → define rótulos para os plots do gráfico.
 - ♦ **`text(x,y,'texto')`** → insere 'texto' na posição (x,y).
 - ♦ **`gtext('texto')`** → insere 'texto' com o auxílio do mouse.

6



Manipulação Gráfica (II)

- Funções para controle de eixos em gráficos:

- ♦ **axis**($[x_{\min} \ x_{\max} \ y_{\min} \ y_{\max}]$) → define os valores mínimos e máximos dos eixos com base nos valores fornecidos pelo vetor linha.
- ♦ **v = axis** → guarda em v o vetor linha que contem os dados de escala do gráfico atual: $[x_{\min} \ x_{\max} \ y_{\min} \ y_{\max}]$.
- ♦ **axis xy** → usa o sistema de coordenadas cartesianas (default), de modo que a origem do gráfico apareça no canto inferior esquerdo.
- ♦ **axis ij** → usa o sistema de coordenadas matriciais, de modo que a origem do gráfico apareça no canto superior esquerdo.
- ♦ **axis square** → faz com que o gráfico atual tenha a forma de um quadrado em lugar do retângulo habitual.
- ♦ **axis equal** → define o fatores de escalamento para ambos os eixos como sendo iguais.
- ♦ **axis normal** → desfaz o efeito dos comandos axis square e axis equal.
- ♦ **axis off** → retira a cor de fundo e os nomes dos gráficos.
- ♦ **axis on** → devolve a cor de fundo e os nomes dos eixos.

7



Manipulação Gráfica (III)

- Funções auxiliares para a manipulação gráfica:

- ♦ **figure** → utilizada para criar múltiplas janelas de gráficos.
- ♦ **gcf** → fornece o número da janela de gráficos atualmente em uso.
- ♦ **clf** → limpa a janela de gráficos atual.
- ♦ **close** → fecha a janela de gráfico atual.
- ♦ **close all** → fecha todas as janelas de gráfico.
- ♦ **hold on/off** → em (on) mantém a tela atual para os próximos plots. Em (off) limpa a tela atual antes de traçar o próximo gráfico.

8



Manipulação Gráfica (IV)

• Exercício 1:

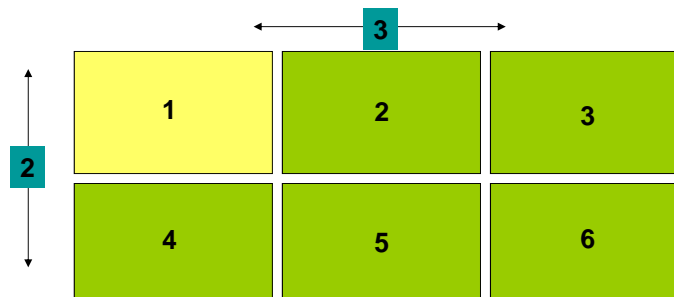
- ♦ Para as funções $\sin(x)$ e $\cos(x)$ faça as seguintes ações:
 - Defina o domínio para ambas entre 0 e 2π com 30 pontos usando coordenadas cartesianas.
 - Traçar no mesmo gráfico ambas as funções, tendo as seguintes características:
 - Função $\sin(x)$ na cor vermelha, estilo de linha cheia (sólido) e marcador em estrela.
 - Função $\cos(x)$ na cor verde, estilo de linha pontilhado e marcador em círculo.
 - Repita o item anterior usando as coordenadas matriciais.

9



Plotagem de Sub-gráficos (I)

- Em determinadas aplicações de computação gráfica é necessária a plotagem de diversos gráficos menores arranjados em uma ordem predefinida.
- O comando **subplot(m,n,k)** subdivide a janela de figuras atual em uma matriz com m por n regiões nas quais se pode traçar gráficos, ativando a região de ordem k:
 - ♦ Exemplo: subplot(2,3,1)



10



Plotagem de Sub-gráficos (II)

• Exercício 2:

- ♦ Dada as seguintes funções:
 - 1.) $f(x) = \sin(x)$;
 - 2.) $f(x) = \cos(x)$;
 - 3.) $f(x) = 2.\sin(x).\cos(x)$
 - 4.) $f(x) = \sin(x)/\exp(x)$
- ♦ Trace cada uma delas em sub-gráficos, com domínio entre 0 e $5\pi/2$, tendo ainda este intervalo 500 pontos e eixos dimensionados automaticamente.
- ♦ Coloque rótulos nos eixos das ordenadas.

11



Funções Gráficas Especiais

- Em determinadas aplicações de computação gráfica é necessário conhecer o valor do ponto de mínimo ou de máximo de uma função.
- Dado um vetor de pontos do eixo das ordenadas definido por fx , a função $fminbnd(fx, x_{inicial}, x_{final})$ procura o valor mínimo da função fx entre os valores de domínio especificados entre $x_{inicial}$ e x_{final} .
 - ♦ Sintaxe da função:
 - $[x_{min}, y_{min}] = fminbnd(fx, x_{inicial}, x_{final})$
- A construção de funções de usuário é também facilitada no matlab através do comando **inline('expressão')**, o qual cria uma função matemática a partir de sua 'expressão' string.
 - ♦ Sintaxe da função:
 - $x = 0:0.01:5$;
 - $y = '1/x'$;
 - $fx = inline(y); \rightarrow fx(5) = 0.2000$

12



Funções Gráficas Especiais

- **Exercício 3:**

- ♦ Dada a função $\frac{1}{(x-0.3)^2 + 0.01} + \frac{1}{(x-0.9)^2 + 0.04} - 6$
- ♦ Elabore um programa conforme as definições abaixo:
 - Gerar a curva acima com 1000 pontos no intervalo de -0.5 a 2.0 .
 - Traçar a curva com eixo horizontal definido entre -0.6 e 2.1, deixando o eixo vertical em escala automática.
 - Encontrar o ponto de mínimo referente ao domínio de 0.5 e 1.0.
 - Encontrar o ponto de máximo referente ao domínio de 0.0 e 0.5.
 - Colocar marcador 'o' em vermelho para o ponto de máximo e marcador 'o' em verde para o ponto de mínimo.
 - Colocar legendas para a curva e marcadores.