

## REPREZENTAREA GRAFICĂ A FUNCȚIILOR REALE DE O VARIABILĂ REALĂ



### 1. Obiective

Însușirea modului de utilizare a mediului de programare Matlab pentru reprezentarea grafică a funcțiilor reale de o singură variabilă reală, numită și reprezentare grafică 2D.



### 2. Noțiuni teoretice

#### 2.1. Reprezentări grafice elementare

##### 2.1.1. Reprezentarea grafică în coordonate carteziene

Funcțiile pentru reprezentări grafice în coordonate carteziene sunt:

1.  $plot(y)$ , se reprezintă grafic argumentul  $y$  funcție de numărul de indici, astfel:

- dacă  $y$  este *număr complex*, funcția reprezintă grafic valorile imaginare ale lui  $y$  funcție de valorile reale ale lui  $y$ :

**$imag(y) = funcție(real(y))$ ;**

- dacă  $y$  este *vector*, se reprezintă grafic valorile vectorului  $y$  funcție de numărul de ordine al elementelor vectorului  $y$ :  $y = y(i)$ ;

- dacă  $y$  este *matrice*, se trasează graficele  $y_j = y_j(i)$ , unde  $i$  este numărul de ordine al elementului de pe coloana  $j$ .

2.  $plot(x,y)$ , se reprezintă grafic vectorul  $y$  funcție de vectorul  $x$ , astfel:

- dacă  $x$  este *vector* și  $y$  este *matrice*, se reprezintă coloanele lui  $y$  funcție de vectorul  $x$ ;

- dacă  $x$  este *matrice* și  $y$  este *matrice*, se reprezintă coloanele lui  $y$  funcție de coloanele lui  $x$ ;

3.  $plot(x1,y1,x2,y2,\dots,xn,yn)$ , se reprezintă două sau mai multe grafice în același sistem de coordonate, respectiv  $y1$  funcție de  $x1$ ,  $y2$  funcție de  $x2$ , ...,  $yn$  funcție de  $xn$ .

4.  $plot(x,y,'specificație linie')$ , unde câmpul '*specificație linie*' cuprinde unul dintre simbolurile următoare:

Simbol stilul liniei

Simbol Matlab	Semnificație
-	linie continuă
--	linie întreruptă
:	linie puncte
-.	linie întreruptă punct

Simbol culoare

$r$  (red);  $g$  (green);  $b$  (blue);  $c$  (cyan);  $m$  (magenta);  $y$  (yellow);  $k$  (black);  $w$  (white).

Simbol marker

Simbol Matlab	Semnificație	Simbol Matlab	Semnificație
+	plus	^	Triunghi cu vârful în sus
o	cerc	v	Triunghi cu vârful în jos
*	steluță	>	Triunghi cu vârful la dreapta
.	punct	<	Triunghi cu vârful la stânga
x	înmulțire (cross)	p	pentagon
s	pătrat (square)	h	hexagon
d	romb (diamond)		

### 2.1.2. Reprezentarea grafică în coordonate logaritmice

Funcțiile Matlab pentru reprezentări grafice în coordonate logaritmice sunt:

1.  $loglog(x,y)$  - Reprezentare grafică în coordonate logaritmice pe axele  $x$  și  $y$
2.  $semilogx(x,y)$  - Reprezentare grafică în coordonate logaritmice numai pe axa  $x$
3.  $semilogy(x,y)$  - Reprezentare grafică în coordonate logaritmice numai pe axa  $y$ .

4.  $\log\log(x,y, \text{'specificație linie'})$

Câmpul  $\text{'specificație linie'}$  cuprinde aceleași componente ca și pentru funcția  $\text{plot}$  cu deosebirea că, axele au scală logaritmică.

### 2.1.3. Reprezentarea grafică în coordonate polare

Funcția de reprezentare grafică în coordonate polare este:

$\text{polar}(\text{theta},r)$  sau  $\text{polar}(\text{theta},r, \text{'specificație linie'})$ ,

unde,

$\text{theta}$  și  $r$  reprezintă argumentele funcției  $\text{polar}$  iar  $\text{'specificație linie'}$  este opțională și are aceeași semnificație ca și la funcția  $\text{plot}$ .

Primul argument ( $\text{theta}$ ) reprezintă vectorul care conține unghiurile față de direcția pozitivă a axei  $x$ , exprimate în radiani, ale componentelor vectorului rază ( $r$ ). Cei doi vectori trebuie să aibă aceeași dimensiune.

Sistemul de axe afișat va fi specific coordonatelor polare.

*Observații:*

1) Numai dacă se utilizează funcția  $\text{plot}$ , se mai pot preciza și alte caracteristici ale liniei în câmpul  $\text{'proprietățile liniei'}$ , sub forma:

$\text{plot}(x,y, \text{'specificație linie'}, \text{'proprietățile liniei'})$

unde:

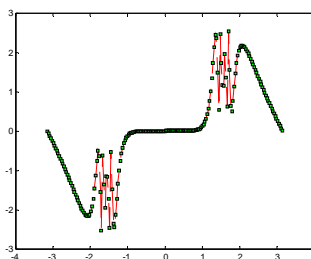
$\text{'proprietățile liniei'}$  este un câmp de caractere care reprezintă proprietățile liniei. Proprietățile liniei pot fi:

- `LineWidth` - scalar care semnifică grosimea liniei;
- `MarkerSize` - scalar care semnifică dimensiunea marcatorului;
- `MarkerEdgeColor` - simbol culoare pentru culoarea de contur a marcatorului;
- `MarkerFaceColor` - simbol culoare pentru zona interioară a marcatorului, dacă acesta este delimitat de un contur închis.

*Exemplu:*

$\text{plot}(x,y, \text{'LineWidth'},1, \text{'MarkerEdgeColor'}, \text{'k'}, \text{'MarkerFaceColor'}, \text{'g'}, \text{'MarkerSize'},4)$ ,

pentru care se obține un grafic care are grosimea liniei  $1$ , culoarea de contur a marcatorului, *negru*, culoarea pentru zona interioară a marcatorului, *verde*, dimensiunea marcatorului,  $4$ .



2) Fiecare tip de obiect grafic are o serie de proprietăți ce pot fi modificate (setate). Sistemul MATLAB asociază fiecărui obiect grafic creat un identificator ("handle"). Pentru a putea accesa proprietățile obiectului, este util ca, la crearea acestuia, să se rețină identificatorul într-o variabilă.

Modificarea proprietăților obiectelor grafice se realizează apelând comanda *set*, care are următoarea sintaxă:

*set (identificator, 'numele\_proprietatii', valoarea\_proprietatii).*

Pentru a obține valoarea curentă a unei proprietăți se apelează funcția *get* cu următoarea sintaxă:

*get (identificator, 'numele\_proprietatii' ).*

Pentru obiectul figură, în mod implicit identificatorul este un număr întreg afișat în bara de titlu a ferestrei grafice. În versiunile mai noi de MATLAB (începând cu 5), prin setarea proprietății *IntegerHandle* a figurii la valoarea *off*, identificatorul figurii poate fi un număr în virgulă flotantă. Identificatorii oricăror alte obiecte grafice sunt numere în virgulă flotantă.

Există și următorii identificatori predefiniți:

*gcf* – pentru fereastra figură curentă (“Get Current Figure”);

*gca* – pentru sistemul de axe curent (“Get Current Axes”).

*Exemple de utilizare a funcțiilor set și get:*

*set(gcf, 'color', 'w')* – stabilește culoarea alb pentru figura curentă;

*set(gca, 'color', 'b')* – stabilește culoarea albastru pentru sistemul de axe curent;

*get(gcf, 'color')* – returnează culoarea figurii curente;

*get(gca, 'color')* – returnează culoarea sistemului de axe curent.

## 2.2. Reprezentări grafice speciale

### 2.2.1. Reprezentarea grafică a poligoanelor

Funcția Matlab de reprezentare grafică a poligoanelor este:

1. *fill(x,y, c)* – Reprezentarea grafică a unui **poligon** care are coordonatele vârfurilor specificate prin perechile  $(x,y)$ , cu nuanțele de culoare precizate de simbolul *c*;

2. *fill(x1,y1, c1, x2,y2, c2, ..., xn,yn, cn)* – Reprezentarea grafică a două sau mai multe poligoane definite de perechile  $(x1,y1)$  și culoarea specificată de *c1*,  $(x2,y2)$  în culoarea specificată de *c2, ..., (xn,yn) în culoarea specificată de *cn*.*

Argumentele  $x$  și  $y$  sunt vectori ce conțin coordonatele  $x$ , respectiv  $y$  ale vârfurilor poligonului. Al treilea argument poate fi un caracter din lista de culori sau un vector cu trei componente  $[r\ g\ b]$ , componentele vectorului fiind numere reale în intervalul  $[0\ 1]$  cu semnificația ponderii culorilor de bază (roșu, verde și albastru) în culoarea respectivă.

Correspondența dintre culorile de bază și forma vectorului  $[r\ g\ b]$  este cea din tabelul următor:

Simbol	$[r\ g\ b]$	Culoare
y	[1 1 0]	Galben
m	[1 0 1]	Purpuriu
c	[0 1 1]	Albastru deschis
r	[1 0 0]	Roșu
g	[0 1 0]	Verde
b	[0 0 1]	Albastru
w	[1 1 1]	Alb
k	[0 0 0]	Negru

### 2.2.2. Reprezentarea grafică cu bare

Funcția Matlab de reprezentare grafică cu bare este:

1.  $bar(y)$  prin care se trasează un grafic de bare cu elementele vectorului  $y$ , adică  $y=y_i$ ;

2.  $bar(x,y)$ , se trasează graficul de bare cu elementele vectorului  $y$  în punctele specificate de vectorul  $x$ , adică  $y=y(x)$ . Valorile lui  $x$  trebuie să fie crescătoare și egal depărtate între ele;

3.  $bar(x,y,lățime\_bare)$  la fel cu funcția precedentă dar la care se mai adaugă lățimea barelor. Implicit, dacă nu se specifică, aceasta este 0.8;

4.  $bar(x,y, 'stilul\_bare')$ , unde câmpul  $'stilul\_bare'$  poate fi,  $'group'$  sau  $'stack'$  (stivă);

5.  $bar(x,y, 'specificație\_linie')$ , unde  $'specificație\_linie'$  are aceeași semnificație ca și la funcția  $plot$ .

### 2.2.3. Reprezentarea grafică a semnalelor discrete

Funcția de reprezentare grafică a semnalelor discrete are forma:

1.  $stem(y)$ , se reprezintă grafic linii terminate cu un cerc, care reprezintă elementele vectorului  $y$ ;

2.  $stem(x,y)$ , se reprezintă un grafic din linii terminate cu un cerc care reprezintă elementele vectorului  $y$  în punctele specificate de vectorul  $x$ , adică  $y=y(x)$ . Valorile lui  $x$  sunt *crescătoare și egal depărtate între ele*.

## 2.2.4. Reprezentarea graficelor în trepte

Aceste reprezentări grafice sunt utilizate, în general, pentru trasarea diagramelor sistemelor numerice de eșantionare și prelucrare a datelor.

Funcțiile Matlab pentru reprezentarea graficelor în trepte sunt de forma:

1.  $stairs(y)$ , se reprezintă graficul în trepte al elementelor vectorului  $y$ ;
2.  $stairs(x,y)$ , se reprezintă graficul în trepte al elementelor vectorului  $y$  în locurile indicate de valorile vectorului  $x$ . Valorile lui  $x$  trebuie să fie crescătoare și egal depărtate între ele;
3.  $stairs(x,y, 'specificație\_linie')$ , unde '*specificație\_linie*' are aceeași semnificație ca și la funcția *plot*.

## 2.2.5. Personalizarea graficelor

Personalizarea graficelor se referă la plasarea în câmpul reprezentărilor grafice a câmpurilor text, a etichetelor pe axe, a titlului, starea caracteristicilor axelor, ș.a.

### 2.2.5.1. Texte în câmpul reprezentărilor grafice

**1. Instrucțiunile pentru inserarea titlului** sunt:

a.  $title(\text{'șir de caractere'})$ , se plasează șirul de caractere în afară, deasupra și centrat față de axe;

b.  $title(date)$  - include valorile variabilelor în titlu și are forma generală:  $title([\text{'șir de caractere'}, num2str(x)])$ ,

unde,

câmpul '*șir de caractere*' conține numele variabilei iar *num2str* se referă la formatul de afișare al valorii variabilei.

*Observații*

a) Dacă variabila care apare în titlu este de tip întreg atunci forma funcției este:

$title([\text{'șir de caractere'}, int2str(x)])$

b) Dacă titlul conține caractere din alfabetul grecesc, pentru afișarea acestora, numele trebuie să fie precedate de caracterul  $\backslash$ .

*De exemplu* pentru literele grecești  $\omega\tau$  funcția devine:

$title([\backslash\omega\backslash\tau', num2str(x)])$

Pentru afișarea caracterele speciale (litere grecești) sau simboluri matematice se folosesc următoarele simboluri:

Simbol matematic	Caractere Matlab	Simbol matematic	Caractere Matlab
$\alpha$	<code>\alpha</code>	$\leftarrow$	<code>\leftarrow</code>
$\beta$	<code>\beta</code>	$\rightarrow$	<code>\rightarrow</code>
$\gamma$	<code>\gamma</code>	$\leftrightarrow$	<code>\leftrightarrow</code>
$\delta$	<code>\delta</code>	$\circ$	<code>\circ</code>
$\varepsilon$	<code>\epsilon</code>	$\div$	<code>\div</code>
$\sigma$	<code>\sigma</code>	$\pm$	<code>\pm</code>
$\eta$	<code>\eta</code>	$\geq$	<code>\geq</code>
$\theta$	<code>\theta</code>	$\leq$	<code>\leq</code>
$\lambda$	<code>\lambda</code>	$\infty$	<code>\infty</code>
$\pi$	<code>\pi</code>	$\partial$	<code>\partial</code>

c) Dacă se dorește scrierea unui caracter la o putere în titlu, atunci numele trebuie precedat de caracterul  $\wedge$ .

*De exemplu* scrierea caracterului  $\omega$  la puterea  $n$  în titlu :

`title(['\omega^n', num2str(x)])`

d) Dacă se dorește scrierea unui caracter indice în titlu, atunci indicele trebuie precedat de caracterul  $\_$ .

*De exemplu* pentru scrierea  $x_1$ , în titlu, funcția va fi:

`title(['x_1', num2str(x)])`.

## 2. Setare etichete pe axe

`xlabel('șir de caractere')` - Pe axa  $x$  afișarea câmpului 'șir de caractere'

`ylabel('șir de caractere')` - Pe axa  $y$  afișarea câmpului 'șir de caractere'

`zlabel('șir de caractere')` - Pe axa  $z$  afișarea câmpului 'șir de caractere'.

**3. Plasarea unui text pe grafic la o poziție impusă** se face prin funcția Matlab:

`text(x,y, 'șir de caractere')`,

unde  $x$  și  $y$  sunt coordonatele punctului pe graficul unde se plasează textul 'șir de caractere', respectiv,

`text(x,y,z, 'șir de caractere')`, unde  $x$ ,  $y$  și  $z$  sunt coordonatele punctului pe graficul în spațiu unde se plasează textul 'șir de caractere'.

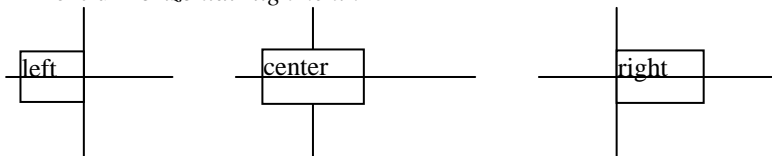
Pentru setarea altor proprietăți ale unui câmp *text* pe grafic, cum ar fi *alinierea* pe orizontală și pe verticală față de punctul de coordonate precizat și mărimea fontului, se folosește instrucțiunea:

`text(x,y, 'șir de caractere', 'VerticalAlignment', 'middle', ...  
'HorizontalAlignment', 'left', 'FontSize', 14 )`

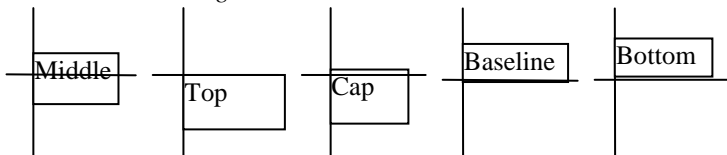
Opțiunile pentru proprietățile *'VerticalAlignment'* și *'HorizontalAlignment'* sunt următoarele:

VerticalAlignment	middle	cap
	bottom	baseline
	top	
HorizontalAlignment	right	
	left	
	center	

Pentru *'HorizontalAlignment'*:



Pentru *'VerticalAlignment'*:



*Alinierea implicită este:*

*HorizontalAlignment = left; VerticalAlignment = middle.*

În plus față de cele mai de sus, Matlab-ul permite controlul *caracteristicilor fontului* utilizat. Pentru aceasta, există următoarele sintaxe:

- `\bf` - font bold
- `\it` - font italic
- `\fontname{fontname}` – specifică numele familiei de fonturi utilizate
- `\fontsize{fontsize}` – specifică dimensiunea fontului în FontUnits.

Efectul acestor caracteristici se aplică numai până la sfârșitul șirului de caractere sau numai în interiorul contextului definit de acolade { }.



*De exemplu*

`text(0.1,0.2, \rightarrow\fontname{times new roman}{\itx}^{-2\{itt^2\}} ')`

se obține, în punctul de coordonate (0.1,0.2) textul:  $\rightarrow x^{-2t^2}$ .

**4. Plasarea unui text pe grafic, la o poziție selectabilă cu mouse-ul,** se face prin funcția Matlab:

`gtext('șir de caractere')`.

### 2.2.5.2. Caracteristicile axelor

#### 1. Limitele axelor

Pentru reprezentările grafice, MATLAB determină automat limitele axelor pe baza domeniului datelor ce sunt reprezentate. Pentru modificarea acestora se apelează comanda *axis*, cu sintaxa:

`Axis([xmin,xmax,ymin,ymax])`

unde:

`[xmin, xmax, ymin, ymax]` este un vector care conține noile limite pe cele două axe. În acest vector, *xmin* și *xmax* reprezintă limita inferioară, respectiv superioară pentru abscisă, iar *ymin* și *ymax* sunt limita inferioară, respectiv superioară pentru ordonată.

Dacă se dorește ca unele limite să fie autoscalate, în vectorul argument, la poziția respectivă se folosește variabila *Inf* pentru o limită superioară, respectiv *-Inf* pentru o limită inferioară.

Pentru a determina autoscalarea reprezentării grafice curente, prin alegerea de către sistemul MATLAB a limitelor reprezentării grafice, se folosește sintaxa:

`axis('auto')` sau `axis auto`.

#### 2. Gradarea axelor

MATLAB-ul localizează marcasele de gradare pe axe și valoarea gradațiilor în funcție de valorile extreme ale datelor de reprezentat. Modul de gradare al celor două axe poate fi schimbat prin modificarea valorilor proprietăților *'xtick'*, respectiv *'ytick'* ale obiectului grafic axe. Pentru acestea, se specifică vectori ale căror componente reprezintă valoarea gradațiilor ce vor fi utilizate pentru axe.

Setarea acestor proprietăți se face cu comanda *set*.

*Exemple:*

`set(gca,'xtick',4:9)` sau `set(gca,'ytick',[4 4.1 4.2 4.3 4.4])`

Referitor la semnul grafic (linii) folosite la gradarea axelor, sunt disponibile următoarele proprietăți și valori posibile ale acestora:

a) *'ticklength'* vector cu două elemente care specifică lungimea liniei de gradare prin raportul dintre lungimea ei și lungimea axei celei mai lungi. Primul element al vectorului se referă la reprezentările 2D, iar al doilea la cele 3D. Vectorul implicit este [0.01 0.025].

Setarea acestor proprietăți se face cu comanda *set*.

*Exemple: set(gca,'ticklength',[0.015 0.025])*

b) *'tickdir'* cu două valori posibile *'in'/'out'* care precizează orientarea liniuțelor de gradare spre interior (*'in'*) sau spre exterior (*'out'*).

### 3. Sistemul de axe

1. Modificarea modului de afișare a celor două axe se realizează cu comanda *axis* astfel:

a) *axis('square')* sau *axis square*

Această comandă face ca axele  $x$  și  $y$  să aibă aceeași lungime.

b) *axis('equal')* sau *axis equal*

Comanda determină aceeași factori de scală pentru ambele axe.

c) *axis('off')* sau *axis off*

Cu această comandă se șterge sistemul de axe și etichetele acestuia;

d) *axis('on')* sau *axis on*

Comandă care reface sistemul de axe și etichetele acestuia;

2. Setarea modului de prezentare a spațiului dreptunghiular asociat sistemului de axe este posibilă prin următoarele proprietăți ale obiectului grafic *axe*. Astfel,

a) Proprietatea *'Box'* cu două valori *'on'/'off'* realizează încadrarea sau neîncadrarea spațiului reprezentării grafice într-un dreptunghi.

b) Proprietatea *'Position'* specifică poziția și lungimea axelor în fereastra figura, prin vectorul de forma [*xinit yinit lngx lngy*].

Componentele *xinit* și *yinit* sunt distanțele, pe orizontală respectiv verticală de la colțul din stânga jos al ferestrei figură la colțul din stânga jos al reprezentării grafice. Componentele *lngx* și *lngy* sunt lungimile celor două axe.

Unitățile de măsură ale componentelor vectorului poziție sunt, în mod implicit, unitățile normalizate, pentru care colțul din stânga jos al ferestrei figură are coordonatele (0,0), iar colțul din dreapta sus are coordonatele (1,1). Acestea pot fi schimbate prin proprietatea '*units*'.

c) Proprietatea '*Units*'

cu valorile, '*pixels*'/'*normalized*'/'*inches*'/'*centimeters*'/'*points*' specifică sistemul de unități de măsură ce vor fi utilizate la interpretarea proprietății '*position*'.

Indiferent de unitățile specificate, măsurarea se face față de colțul din stânga jos al ferestrei grafice.

d) Proprietatea '*color*' specifică culoarea folosită pentru fondul reprezentării grafice din axele curente cu două valori '*symbol\_culoare*'/'*non*'.

e) Proprietatea '*Visible*' specifică vizibilitatea sistemului de axe prin două valori '*on*'/'*off*'.

#### 4. Reprezentări grafice succesive într-un sistem de axe existent

Prin comanda *hold* se pot adăuga grafice cu păstrarea celor existente într-un sistem de axe, cu sintaxa:

*hold on* permite adăugarea graficelor următoare peste cel curent, fără ștergere, cu rescalarea axelor dacă este cazul;

*hold off* revine la setarea inițială, când, la apelul unei funcții de reprezentare grafică, în același sistem de axe, graficul anterior este șters.

#### 5. Ștergerea axelor curente

Pentru ștergerea reprezentărilor grafice din axele curente se poate folosi funcția *cla* cu două sintaxe:

- *cla* - șterge toate graficele trasate în sistemul de axe curent;
- *cla reset* - în plus, resetează toate proprietățile obiectului *axe* la valorile implicite, cu excepția poziției acestuia.

6. Trasarea unei rețele de linii (*grid*) pe grafic se face cu funcția Matlab:

*grid on* – adaugă linii grid pe sistemul de axe

*grid off* – șterge linii grid pe sistemul de axe.

În funcție de dorința utilizatorului numărul liniilor de grid pe cele două axe poate fi modificat.

### 2.2.5.3. Divizarea ferestrei grafice

Sistemul MATLAB permite împărțirea ferestrei figură în mai multe ferestre grafice și afișarea, în fiecare dintre acestea, a câte unui sistem de axe. Crearea acestor ferestre și controlul lor se realizează cu funcția *subplot* care are următoarea sintaxă:

*subplot(m,n,i).*

Această comandă împarte fereastra figură în  $m \times n$  zone dreptunghiulare de afișare ( $m$  numărul de linii și  $n$  numărul de coloane) și specifică fereastra cu numărul  $i$  ca fiind cea curentă (în care se va face următoarea reprezentare grafică). Zonele de afișare sunt numerotate de la stânga la dreapta, începând cu linia de sus.

### 2.2.5.4. Determinarea coordonatele punctelor indicate cu mouse-ul pe o reprezentare grafică

Sistemul MATLAB permite indicarea coordonatelor punctelor dintr-o fereastră figură, utilizând mouse-ul pentru poziționarea cursorului, prin utilizarea funcției *ginput*.

Există trei sintaxe posibile ale acestei funcții:

1.  $[x,y] = \text{ginput}(n)$

prin care se determină coordonatele, în vectorii coloana  $x$  și  $y$ , pentru  $n$  puncte din fereastra figură, indicate cu ajutorul mouse-ului;

*Observație:* Încheierea operației înainte de selectarea celor  $n$  puncte se poate realiza apăsând tasta *Enter*.

2.  $[x,y] = \text{ginput}$

care permite obținerea, în vectorii coloana  $x$  și  $y$ , a coordonatelor unui număr nelimitat de puncte, din fereastra figură, până la apăsarea tastei *Enter*.

3.  $[x,y,z] = \text{ginput}(n)$

se permite, în plus față de celelalte două sintaxe, determinarea în vectorul  $z$  a informațiilor privind modul cum se realizează selectarea punctului.

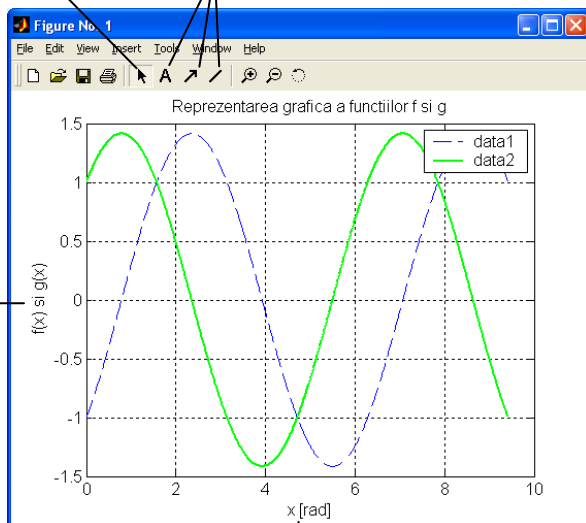
Vectorul  $z$  ia valorile 1, 2, respectiv 3 în funcție de butonul mouse-lui care a fost apăsat pentru indicarea punctului, stânga, mijloc respectiv dreapta sau codul ASCII asociat unei taste, dacă selectarea s-a realizat prin apăsarea unei taste.

Toate instrucțiunile prezentate mai sus pentru personalizarea graficelor (§2.2.5) se referă la modalitățile de comandă direct prin fișierul sursă (.m). Personalizarea graficelor se poate face și direct pe fereastra grafică prin utilizarea proprietăților din “Edit plot” (Fig.1), dar care sunt active atâta timp cât fereastra grafică este deschisă.

Deschiderea modului de editare a graficului prin apăsare buton "Edit plot"

Utilizare butoane pentru adăugare pe grafic: text, săgeată, linie

Eticheta pe axa y;  
se face click și se poate muta



Eticheta pe axa x;  
se face click și se poate muta sau edita



### 3. Probleme de rezolvat

#### 3.1. Reprezentări grafice în coordonate carteziene

Să se reprezinte grafic următoarele funcții reale de o variabilă reală:

1.  $f(x) := 1 + \sqrt{25 - x^2} + |1 - \sqrt{25 - x^2}|$ , pentru  $x \in [-5, 5]$
2.  $f(x) := \sqrt[3]{x^2} + |x^2|$ , pentru  $x \in [-3, 3]$
3.  $f(x) := |x^3 + 1| \cdot \sqrt{x^3 + 1} + x$ , pentru  $x \in [-1, 1]$

4.  $f(x) := \sqrt{|1-x^2|} + \sqrt{1+x^2}$ ,      pentru  $x \in [-2, 2]$
5.  $f(x) := x^2 \cdot \ln\left(\frac{1}{x^2}\right)$ ,      pentru  $x \in [-1, 1]$
6.  $f(x) := x \cdot \frac{1}{e^{|x|}}$ ,      pentru  $x \in [-10, 10]$
7.  $f(x) := \ln|x| \cdot \sin(x)$ ,      pentru  $x \in [-2\pi, 2\pi]$
8.  $f(x) = \frac{1}{x^2} \cdot \ln(x^2)$ ,      pentru  $x \in [1, 5]$
9.  $f(x) = \frac{1}{x^2} \cdot \sin(x^2)$ ,      pentru  $x \in [-5, 5]$
10.  $f(x) = \cos(x) \cdot \sin(x) - \sin(x)$ ,      pentru  $x \in [-4\pi, 4\pi]$

### 3.2. Reprezentări grafice în coordonate polare

Să se reprezinte în coordonate polare următoarele funcții:

1.  $\rho(\theta) = 2 \cdot \theta$  și  $\xi(\theta) = -2 \cdot \theta$ , unde:  $\theta \in [0, 2\pi]$ .
2.  $\rho(\theta) = e^{\frac{2}{15}\theta}$ ,  $\xi(\theta) = e^{-\frac{2}{15}\theta}$ , unde:  $\theta \in [0, 6\pi]$ .
3.  $\rho(\theta) = \sqrt[3]{\theta^2} + 3 \cdot \sqrt[5]{\theta^2} + 2 \cdot \sqrt[7]{\theta^2}$ , unde:  $\theta \in [0, 5\pi]$ .
4.  $f(t) = \cos(t)^2 + \sin(t)^2$ ;  $g(t) = 2 \cdot \sin(t) + 2 \cdot \cos(t)$ ,  
unde:  $t \in [0, 2\pi]$ .

### 3.3. Personalizare grafice

1. Linie continuă, culoare albastru, grosimea liniei 2, titlu: „Funcția f”, eticheta pe abscisă: „x”, eticheta pe ordonată: „f(x)”, legenda afișată;
2. Linie punct, culoare verde, grosimea liniei 3, titlu și etichete pe axe, legenda afișată;
3. Linie întreruptă, culoare purpuriu (magenta), grosimea liniei 1, titlu și etichete pe axe, legenda afișată;
4. Linie întreruptă-punct, culoare negru, grosimea liniei 3, titlu și etichete pe axe, legenda afișată;
5. Linie punct, culoare albastru deschis (cyan), grosimea liniei 4, titlu și etichete pe axe, legenda afișată;

6. Linie întreruptă, culoare albastru, grosimea liniei 2, titlu și etichete pe axe, fără legendă;
7. Linie continuă, culoare roșu, grosimea liniei 3, titlu și etichete pe axe;
8. Linie întreruptă-punct, culoare verde, grosimea liniei 4, titlu și etichete pe axe;
9. Linie continuă, culoare purpuriu (magenta), grosimea liniei 3, titlu și etichete pe axe;
10. Linie continuă, culoare albastru deschis (cyan), grosimea liniei 2, titlu și etichete pe axe;

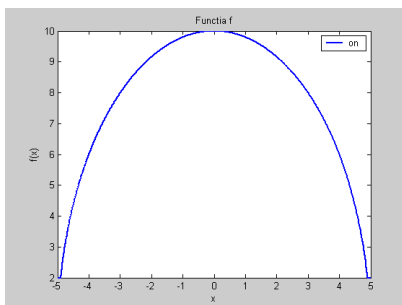


## 4. Probleme rezolvate

### 4.1. Reprezentări grafice în coordonate carteziene

Programul Matlab pentru §3.1.1 cu cerințele §3.3.1:

```
x=-5:0.01:5;
f=1+sqrt(25-x.^2)+abs(1-
sqrt(25-x.^2));
plot(x,f,'-b','LineWidth',2)
title('Funcția f')
xlabel('x')
ylabel('f(x)')
legend on
```



### 4.2. Reprezentări grafice în coordonate polare

Programul Matlab pentru §3.2.1:

```
theta = 0:.01:2*pi;
ro=2*theta;
psi=-2*theta;
polar(theta,ro,'--r');hold on
polar(theta,psi,'-b')
```

