

CAPITULO III

INTRODUCCION

Con los conocimientos adquiridos en los dos primeros capítulos y los profundizados con el estudio independiente ya se tienen las bases para poder utilizar las opciones gráficas que Matlab ofrece, pues no es simplemente graficar si no utilizar los comandos que se mostraran en este último capítulo con los aprendidos anteriormente para crear programas que demuestren el manejo de esta súper calculador; así mismo con este capítulo se cierra el ciclo de lo que todo ingeniero o interesado en el área de las matemáticas debe conocer.

GRAFICOS CON MATLAB

- **Graficas en 2-D**

Para realizar graficas en dos dimensiones en Matlab se cuenta con la función plot, en la cual los argumentos son vectores de la misma longitud. Así mismo esta función permite superponer graficas sobre los mismos ejes así como realizar las gráficas con distintos caracteres y colores, todo depende de la sintaxis con la que se haya escrito la función.

La sintaxis para esta función es la siguiente:

```
plot(Y)
plot(X1,Y1,...,Xn,Yn)
plot(X1,Y1,LineStyle,...,Xn,Yn,LineStyle)
plot(X1,Y1,LineStyle,'PropertyName',PropertyValue)
plot(axes_handle,X1,Y1,LineStyle,'PropertyName',PropertyValue)
h = plot(X1,Y1,LineStyle,'PropertyName',PropertyValue)
```

Para colocar las etiquetas o realizar cambios en la grafica se cuenta con estas funciones:

Etiqueta sobre el eje X de la gráfica actual: >> **xlabel('texto')**

Etiqueta sobre el eje Y de la gráfica actual: >> **ylabel('texto')**

Título en la cabecera de la gráfica actual: >> **title('texto')**

Texto en el lugar especificado por las coordenadas: >> **text(x,y,'texto')**

Texto, el lugar lo indicamos después con el ratón: >> **gtext('texto')**

Dibujar una rejilla: >> **grid**

Fija valores máximo y mínimo de los ejes: >> **axis([xmin xmax ymin ymax])**

Fija que la escala en los ejes sea igual: >> **axis equal**


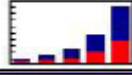
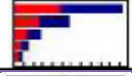


Fija que la gráfica sea un cuadrado: >> **axis square**


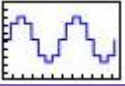
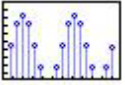
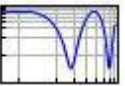
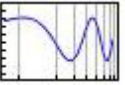
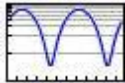
Desactiva **axis equal** y **axis square**: >> **axis normal**

Abre una ventana de gráfico: >> **hold on**

Borra lo que hay en la ventana de gráfico: >> **hold off**

Otros comandos utilizados con la función plot son:

COMANDO	SU FUNCIÓN ES	IMAGEN
Área	Pinta el área bajo la curva	
Bar	Muestra un diagrama de barras vertical	
Barh	Muestra un diagrama de barras horizontal	
Hist	Histograma	
Pie	Sectores	

Rose	Histograma polar	
Stairs	Gráfico de escalera	
Stem	Secuencia de datos discreto	
Loglog	Es un plot con escala logarítmica en ambos ejes	
Semilogx	Es un plot pero con escala logarítmica solo en x	
Semilogy	Es un plot pero con escala logarítmica solo en y	

Una ventana gráfica se puede dividir en m particiones horizontales y en n verticales, de modo que cada subventana tiene sus propios ejes, y para hacer esto vamos a usar subplot (m,n,p) donde p indica la subdivisión que se convierte en activa.

- **Graficas en 3-D**

Para crear graficas en tres dimensiones también se utiliza el comando plot sin embargo en este caso los datos se encuentran en ternas, y la función se escribe plot3.

La sintaxis para este caso esta descrita así:

```
plot3(X1,Y1,Z1,...)
plot3(X1,Y1,Z1,LineStyle,...)
plot3(...,'PropertyName',PropertyValue,...)
h = plot3(...)
```

Es muy común observar en las gráficas de Matlab figuras como mayadas, y de superficie y hacerlas no es realmente complicado aquí se muestran las ordenes

Grafica de mayado:

[X,Y]=meshgrid(x,y) %crea una matriz **X** cuyas filas son copias del vector x y una matriz **Y** cuyas columnas son copias del vector y.
Para generar la gráfica de malla se usa la orden **mesh(X,Y,Z)**, **mesh** %acepta un argumento opcional para controlar los colores.
También puede tomar una matriz simple como argumento: **mesh(Z)**.

Gráfica de superficie:

Es similar a la gráfica de malla, pero aquí se rellenan los espacios entre líneas. La orden que se utiliza usualmente es surf con los mismos argumentos que para mesh.

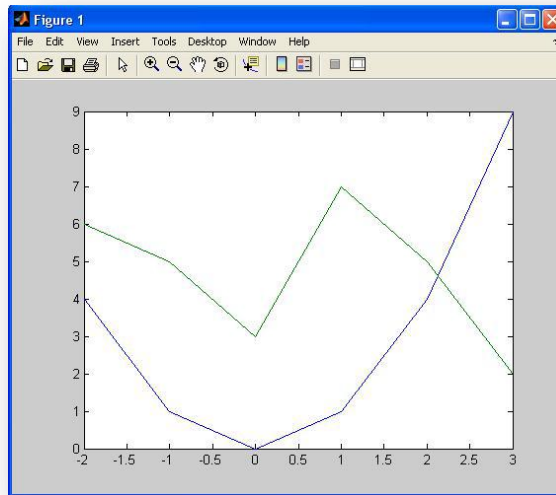
EJERCICIOS RESUELTOS

- **Graficas en 2-D**

Ejemplo sobre posición de varias graficas en los mismos ejes

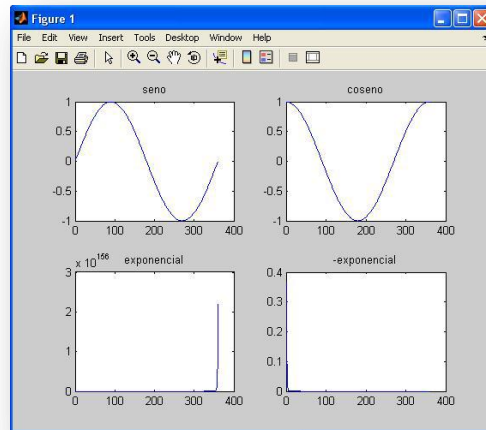
```
>> x = [-2 -1 0 1 2 3]; y = [4 1 0 1 4 9]; z = [6 5 3 7 5 2];
```

```
>> plot(x,y,x,z)
```



Ejemplo división de ventanas con la función plot

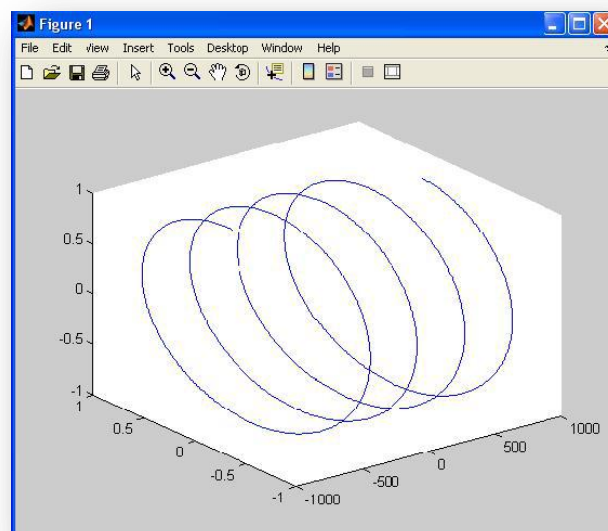
```
>> x = 1:360; y1 = sind (x); y2 = cosd (x); y3 = exp (x); y4 = exp (-x);  
>> subplot (2,2,1), plot (x,y1), title ('seno')  
>> subplot (2,2,2), plot (x,y2), title ('coseno')  
>> subplot (2,2,3), plot (x,y3), title ('exponencial')  
>> subplot (2,2,4), plot (x,y4), title ('-exponencial')
```



- **Gráficas en 3-D**

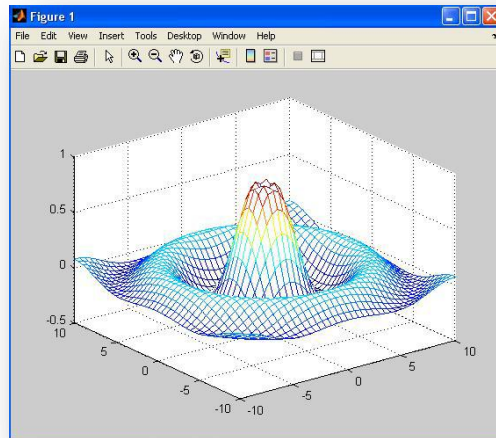
Ejemplo del plot3

```
>> x = -720:720; y = sind (x); z = cosd (x);  
>> plot3 (x,y,z)
```



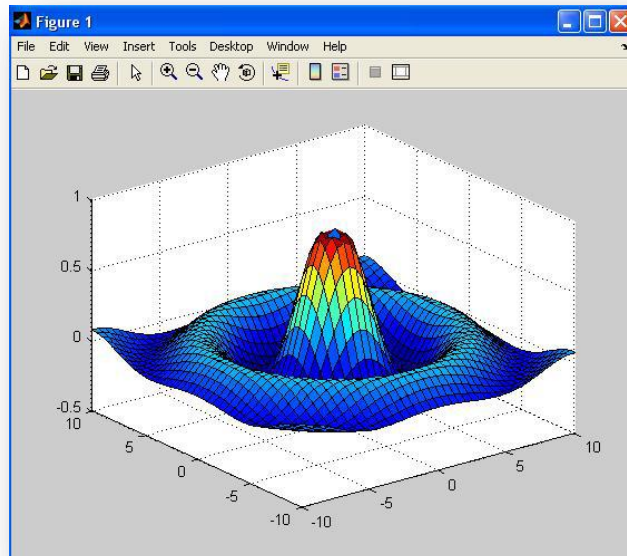
Ejemplo de maya:

```
>> [X,Y] = meshgrid(x,y); % crea matrices para hacer la malla  
>> Z = sin(sqrt(X.^2 + Y.^2)) ./ sqrt(X.^2 + Y.^2 + 0.1);  
>> mesh(X,Y,Z) % dibuja la gráfica
```



Ejemplo de superficie:

```
>> [X,Y] = meshgrid(-10:0.5:10);  
>> Z = sin(sqrt(X.^2 + Y.^2)) ./ sqrt(X.^2 + Y.^2 + 0.1);  
>> surf(X,Y,Z)
```



ESTUDIO INDEPENDIENTE

1. Representar un gráfico de barras definidas por la función $x \cdot e^{-x}$
`x = -2.9:0.2:2.9`
`bar(x, exp(-x.*x))`
`barh(x, exp(-x.*x))`
2. `x = [1 3 0.5 2.5 2];`
`pie(x)`
`y = [0 1 0 0 0];`
`plot(x,y)`
3. `X = linspace(0,1,10);`
`Y = X./2;`
`Z = sin(X) + cos(Y);`
`stem3(X,Y,Z,'fill')`
`sphere`
`axis equal`
4. Representar la superficie cuya ecuación explícita es la siguiente: $z = \sin(\sqrt{x^2+y^2})/\sqrt{x^2+y^2}$
`[x,y] = meshgrid(-7.5:.5:7.5);`
`z = sin(sqrt(x.^2+y.^2))./sqrt(x.^2+y.^2);`
`figure(1)`
`surf(x,y,z)`
`figure(2)`
`surfc(x,y,z)`
`figure(3)`
`figure(4)`