Tema 6: Introducción a los gráficos en 3 dimensiones

1. Introducción

Matlab está preparado para realizar diversos tipos de gráficos en tres dimensiones. Ya se ha comentado que por las características del programa, estos gráficos al igual que los de dos dimensiones, están orientados a la representación de vectores. Esto puede causar un poco de dificultad al principio y habrá que preparar los elementos a representar previamente.

Se pueden realizar gráficos 3D tanto de líneas como de mallas o superficies. Para ello los pasos a seguir serán siempre los mismos:

- Preparar los datos.
- Usar la función gráfica 3D adecuada.
- Retocar las propiedades del dibujo a nuestro gusto.

Destacar que la gran mayoría de los elementos estudiados para gráficos 2D tienen su correspondiente aplicación para 3D.

2. Funciones básicas para las gráficas 3D

La primera fórmula de gráficos en tres dimensiones es la función plot3 que resulta análogo al comando plot en gráficos bidimensionales. Dibuja puntos cuyas coordenadas están contenidas en tres vectores x,y,z. Su sintaxis puede ser: **plot3(x,y,z)**. Como iremos viendo, a este comando de le pueden añadir opciones de dibujo en la misma línea que se presentó para el comando plot en el caso 2D.

Ejemplo:

>> x=[1,2,3]; y=[-1,4,7]; z=[0,-3,5]; >> plot3(x,y,z,'*') Veremos con detalle como se pueden dibujar funciones de dos variables, z=f(x,y). Lo primero que debemos pesar es que por las características propias de Matlab, lo primero es crear el conjunto de puntos (x,y,z) de la gráfica de la función, es decir, con z=f(x,y), que se desean representar. Para ello es necesario considerar los vectores x e y que contienen las coordenadas en una y otra dirección de la retícula (grid) sobre la que se dibujará la función. A partir de ellos se genera la malla donde estarán todos los valores (x,y) sobre los que se evaluará a función para generar los puntos tridimensionales a representar. Así, creamos dos matrices X e Y a partir del comando **meshgrid(x,y)** que representan las coordenadas x e y de los puntos a representar:

>>x=[1,2,3,4]; y=[-1,0,1,2];

>>	[X,Y]=	=meshgrid(x,y)
----	--------	----------------

X =

1	2	3	4		
1	2	3	4		
1	2	3	4		
1	2	3	4		
Y =					
-1	-1	-1	-1		
0	0	0	0		
1	1	1	1		
2	2	2	2		

Como vemos en este caso hemos generado una malla formada por 16 puntos de coordenadas (1,-1),(1,0),(1,1),(1,2),(2,-1),...(2,2),....,(4,-1),...(4,2). Es bueno señalar que por el proceso indicado debe tenerse cuidado con las longitudes de los vectores x e y ya que al generar la malla, el número de puntos con los que se va a trabajar aumenta grandemente.

Sólo queda evaluar la función f sobre esta malla, es decir Z=f(X,Y) y representar los valores. Para ello se utilizan los comandos:

- Mesh(Z) que dibuja en perspectiva la función en base a una retícula de líneas de colores con eliminación de líneas ocultas (figura 23).
- Surf(Z) dibuja la superficie faceteada también con eliminación de líneas ocultas (figura 24).

Destacar que existen otras formas más generales de estas funciones para la representación paramétrica: mesh(x,y,z) dibuja una superficie de puntos (X(i,j),Y(i,j), Z(i,j)).



Figura 23



Figura 24

2.1. Opciones de dibujo

Hemos visto que las funciones anteriores dibujan los gráficos con unas ciertas características. De forma muy parecida a lo explicado para el caso bidimensional, podremos alterar estas características a nuestro gusto. Veremos como hacer algunos cambios desde la línea de comandos pero resulta más cómodo trabajar en la propia ventana de dibujo. Como la mayoría de las opciones de cambio en una gráfica, pueden realizarse desde la pantalla de dibujo, vuelve a resultar de gran utilidad inspeccionar cada uno de los menús de esta ventana y ejecutar las opciones directamente en ella.

- Ejes, puntos de vista, líneas ocultas

La localización del punto de vista o la dirección de observación de la figura representada puede alterarse mediante el comando view de sintaxis: **view(azimut, elevación)**. Donde azimut es el ángulo de rotación en un plano horizontal medido sobre el eje z a partir de un eje x en sentido antihorario, y elevación es el ángulo de elevación respecto al plano xy. Se miden en grados.

De la misma forma se altera desde el menú view, camera/toolbar.

Para el control de los ejes se utiliza el comando axis que ya se presentó en 2D, en su versión tridimensional. También se accede al tratamiento de los ejes desde el munú edit/axes properties.

- El color

En cuanto a la utilización del color, un mapa de colores en Matlab se defina mediante una matriz de tres columnas con valores entre 0 y 1, que representan la intensidad de rojo, verde o azul (R,G,B). El comando **colormap** actúa sobre la figura activa cambiando sus colores. No obstante, existen mapas de colores ya predefinidos en el programa. Pueden elegirse por ejemplo, desde el la pestaña colormap en el editor de propiedades de la superficie.

3. Líneas de contorno

Para los gráficos en tres dimensiones es de gran utilidad la representación de curvas de contorno o de nivel. Algo que puede verse como otra forma de representación, figura 25. Para ello Matlab cuenta con el comando contour. Puede usarse con diferentes sintaxis, por ejemplo:

 contour(Z,n), n representa el número de líneas de nivel. Si no se pone se utiliza un número por defecto. contour(Z,v), donde v es un vector con los valores de las curvas de nivel a dibujar.



Figura 25

Práctica 6: Gráficos 3D

- **1.** Representar gráficamente los puntos siguientes: $p_1=(1,1,1)$, $p_2=(3,2.0)$, $p_3=(0,4,2)$, $p_4=(-3,6,8)$ primeramente conectados y luego aislados.
- Representar las funciones f(x,y)=sen(x)cos(y) en el intervalo [0,2π]x[0,2π].
 Poner título al dibujo y etiquetas en los ejes.
 - a. Realizarlo con el comando mesh.
 - b. Representarlo con el comando surf.
 - c. Realizarlo de forma seguida primero con mesh y luego con surf para visualizar las diferencias. Utilizar el comando pause.
- Representar la gráfica anterior junto con la de g(x,y)=ycos(x) en el intervalo [0,2π]x[0,2π] pero en distintas ventanas.
- **4.** Representar en la misma figura y sobre los mismos ejes la función $\sqrt{x^2 + y^2}$ enfocada desde distintos puntos de vista.
- **5.** Representar gráficamente la función $f(x,y)=x^2-y^2$ en el cuadrado [-2,2]x[-2,2].
 - a. Representar en la misma ventana el plano tangente a la superficie en el origen de coordenadas.
 - b. Visualizarla desde distintos puntos de vista.
 - c. Cambiar el color.
 - d. Utilizar el zoom para visualizar ciertas zonas.
- **6.** Representar f(x,y)=sen $(x+y^2)$ en [-2,2]x[-2,2]. Poner rejilla y volverla a quitar.
 - a. Representar algunas curvas de nivel.
 - b. Representar las curvas a alturas -0.2,-0.1,0,0.1,0.2,0.3.
 - c. Representar en la misma ventana gráfica la superficie y las curvas de nivel
- 7. Dibujar una esfera de radio 1 con 30 puntos. Usar el comando sphere(n).
- Dibujar un cuadrado de vértices (1,0),(2,1),(1,2),(0,1) con el interior del mismo en verde. En el mismo dibujo representar un segmento de extremos (2,1) y (0,1) en color azul. Comando fill(x,y,'color').
- Dibujar el triángulo de vértices (0,0,0), (1,2,3), (3,5,4) con el interior en rojo. Comando fill3

10. Representar la hélice
$$\begin{cases} x(t) = 1 + \cos(t) \\ y(t) = 1 + 2sen(t) \\ z(t) = 4t \end{cases}$$
 y modificar algunas de sus

características gráficas, grosor, etiquetas, etc.