

An Example of Plotting Spheres in Matlab

This example will produce this 3-D plot.

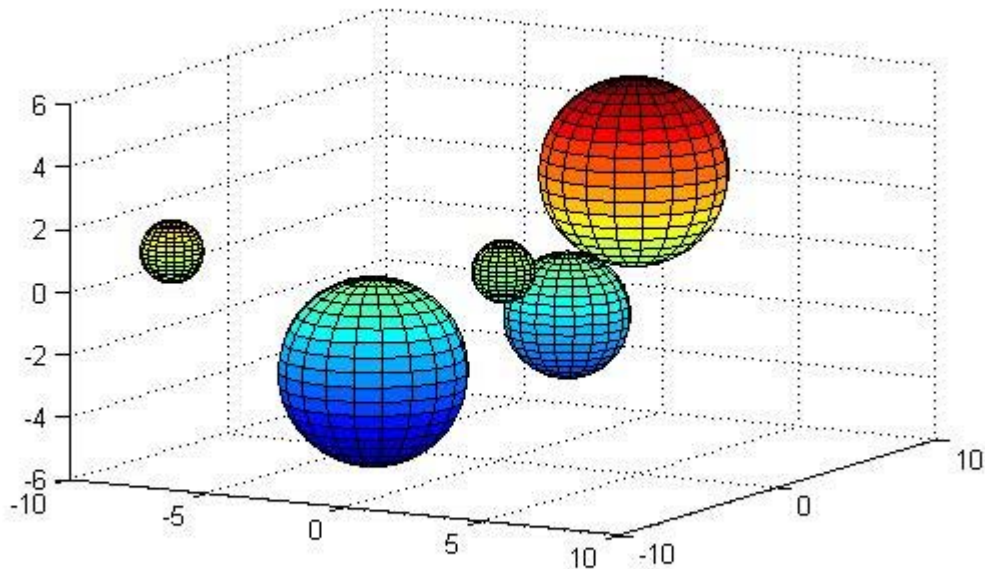


Figure 1. Plot of spheres in Matlab created using the 'sphere' function.

The Matlab function 'sphere' generates the x-, y-, and z-coordinates of a unit sphere for use with 'surf' and 'mesh'. 'Surf' and 'Mesh' are two functions that generate plots in 3-d, where 'surf' will create a 3-d surface plot and 'mesh' will create a wireframe mesh in 3-d. For this example we will be using the 'surf' function.

To use the sphere function it must be called in the m-file as

In the command window type `[x y z] = sphere;`

Since Matlab is matrix based math we need to create a 1 – row, 4 – column matrix representing the coordinates of the sphere (shown in figure 2).

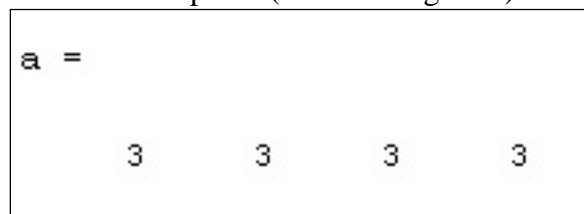


Figure 2. A 1 x 4 matrix representing the coordinates of the sphere.

Where the first column of the matrix represents the x – coordinate, the second column represents the y – coordinate, the third column represents the z – coordinate and the fourth column will represent the radius of the sphere. Typing in the code

```
a=[3 3 3 3]
```

will return a 1 x 4 matrix as shown in figure 2. The next step is to plot the sphere using 'surf'. The code of the plotting line will read

```
s1=surf(x*a(1,4)+a(1,1),y*a(1,4)+a(1,2),z*a(1,4)+a(1,3));
```

where s1 is the variable assigned to sphere 1, x, y and z are the variables x, y and z, a(1,4) represents the number in matrix 'a' in row 1 and column 4 and denotes the radius of the sphere. a(1,1) represents the number in row 1 and column 1 of matrix 'a' and is the x-coordinate of the center of the sphere. The matlab code should now read

```
clc
clear
[x y z] = sphere;
a=[3 3 3 3]
s1=surf(x*a(1,4)+a(1,1),y*a(1,4)+a(1,2),z*a(1,4)+a(1,3));
daspect([1 1 1])
view(30,10)
```

daspect([1 1 1]) changes the scale on each axis to 1 and view(30,10) twists the view around the z-axis. When ran, this code will produce a plot as shown in figure 3.

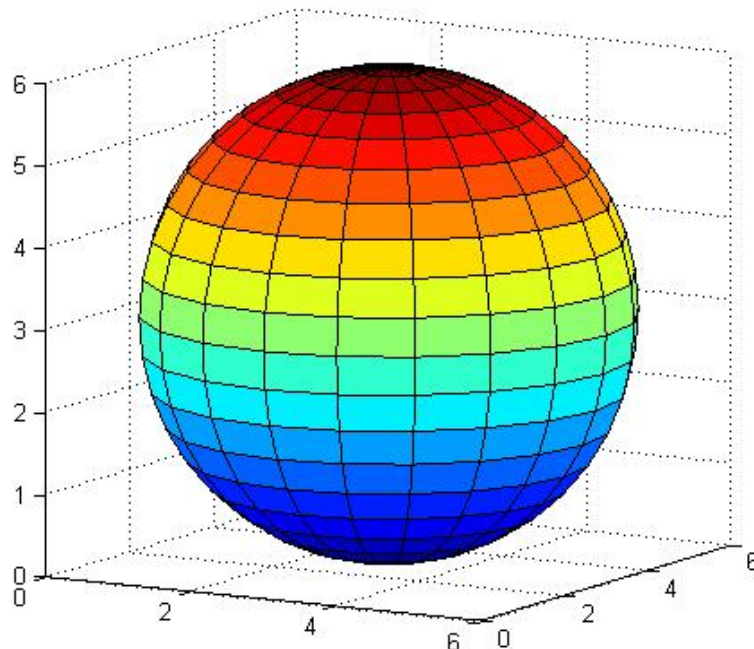


Figure 3. Plot of 1 sphere centered at coordinate (3,3,3) with radius of 3.

To make another sphere in the same plot the 'hold' function must be executed prior to the next plot. The matrix must also be updated with a new row representing the second sphere. Matrix 'a' with 2 rows and 4 columns is shown next.

```
a=[3 3 3 3;-3 -3 -3 3]
```

where the semicolon denotes a new row within the matrix. A new 'surf' must be inserted with reference to the correct number in the matrix 'a'. The code for this is shown next.

```
clc
clear
[x y z] = sphere;
a=[3 3 3 3;-3 -3 -3 3]

s1=surf(x*a(1,4)+a(1,1),y*a(1,4)+a(1,2),z*a(1,4)+a(1,3));
hold on
s2=surf(x*a(2,4)+a(2,1),y*a(2,4)+a(2,2),z*a(2,4)+a(2,3));
daspect([1 1 1])
view(30,10)
```

When ran, this code produces a plot shown in figure 4.

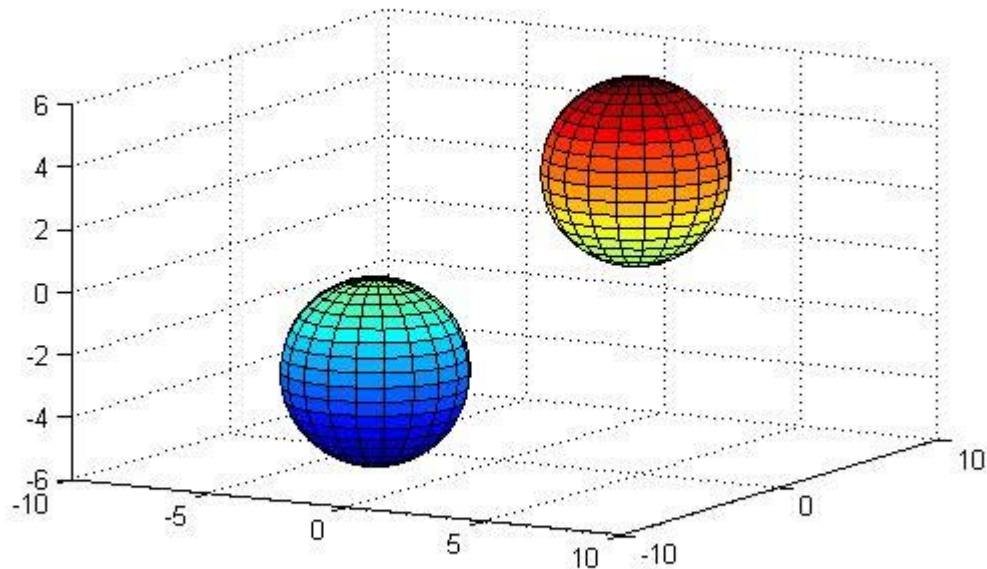


Figure 4. Plot of 2 spheres of radius 3 with one centered at (-3,-3,-3) and the other centered at (3,3,3).

To make another sphere with a smaller radius, simply insert another column into matrix 'a' with a different radius (the number in row 3 column 4) and a new 'surf' as shown next.

```
clc
clear
[x y z] = sphere;
a=[3 3 3 3;-3 -3 -3 3;0 4 -2 2]

s1=surf(x*a(1,4)+a(1,1),y*a(1,4)+a(1,2),z*a(1,4)+a(1,3));
hold on
s2=surf(x*a(2,4)+a(2,1),y*a(2,4)+a(2,2),z*a(2,4)+a(2,3));
s3=surf(x*a(3,4)+a(3,1),y*a(3,4)+a(3,2),z*a(3,4)+a(3,3));

daspect([1 1 1])
view(30,10)
```

The plot for this code is shown in figure 5.

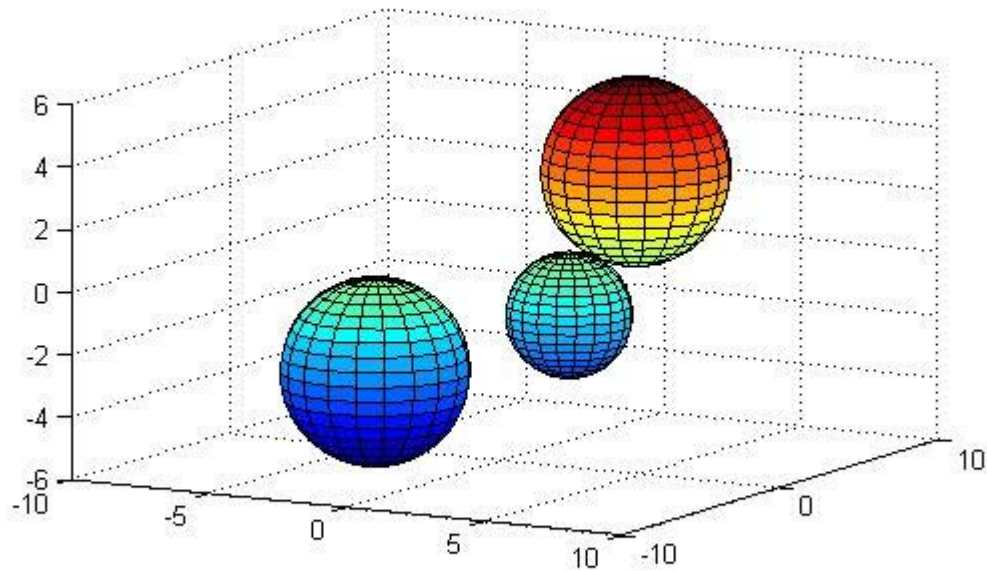


Figure 5. Plot of 3 spheres where, 2 have radii of 3 and one has a radius of 2.

To make more spheres simply add more columns to the matrix 'a' and more 'surf's to plot each extra sphere as shown next.

```
clc
clear
[x y z] = sphere;
```

```
a=[3 3 3 3;-3 -3 -3 3;0 4 -2 2;0 0 0 1;-8 -7 1 1]

s1=surf(x*a(1,4)+a(1,1),y*a(1,4)+a(1,2),z*a(1,4)+a(1,3));
hold on
s2=surf(x*a(2,4)+a(2,1),y*a(2,4)+a(2,2),z*a(2,4)+a(2,3));
s3=surf(x*a(3,4)+a(3,1),y*a(3,4)+a(3,2),z*a(3,4)+a(3,3));
s4=surf(x*a(4,4)+a(4,1),y*a(4,4)+a(4,2),z*a(4,4)+a(4,3));
s5=surf(x*a(5,4)+a(5,1),y*a(5,4)+a(5,2),z*a(5,4)+a(5,3));

daspect([1 1 1])
view(30,10)
```

The plot for this code is shown in figure 1.