Mini-Matlab Lesson 12: Advanced graphics

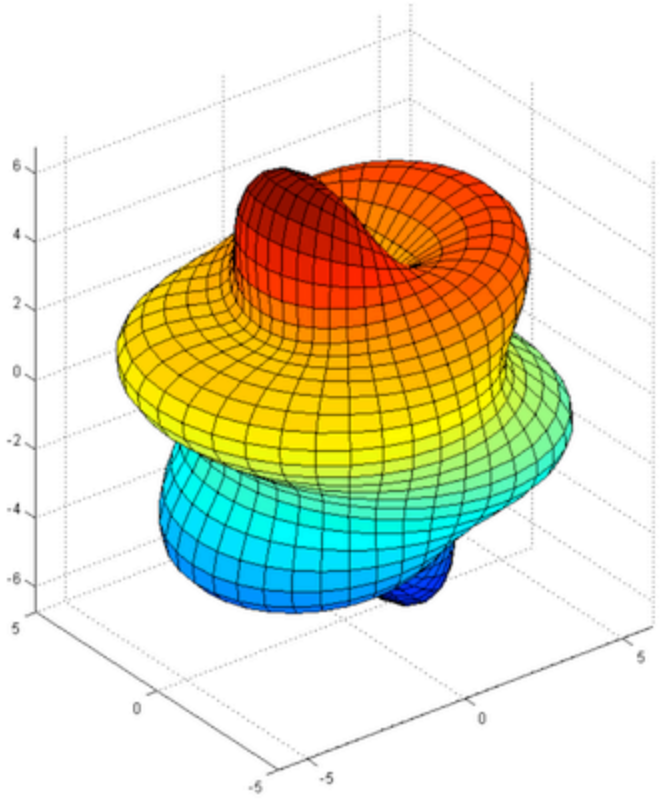
You know how to create basic figures (and how to use the publish command). Often, such as when you are writing papers, creating posters, or movies, you will need to know how to output 'pretty' graphics. We will talk about this.

Contents

Create a 3d surface

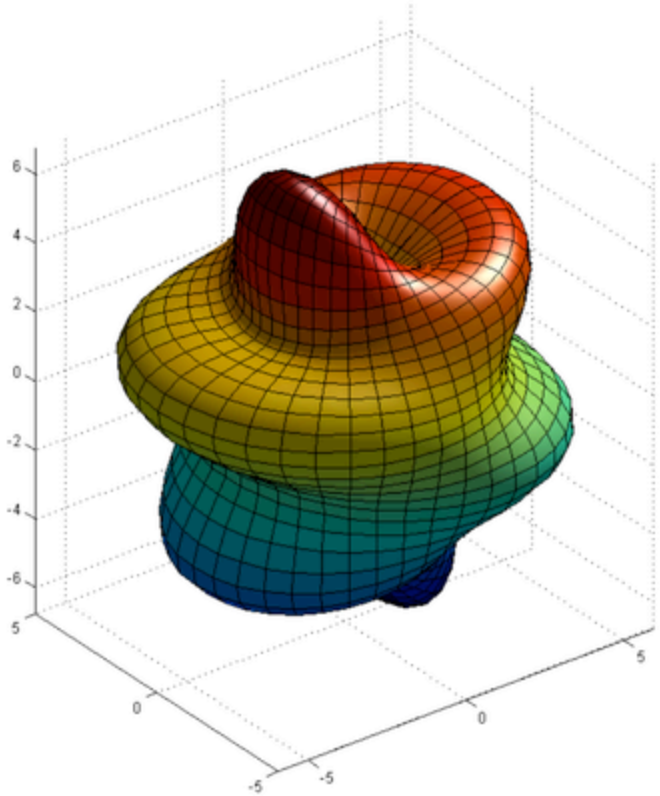
We shall create a surface of the spherical harmonic of degree 6, order 1, and amplitude 2 plotted on the surface of a sphere of radius 5. (these spherical harmonics are basically the series coefficients for a series representation of a certain differential equation (like the Bessel equation you studied in your problem sets).

```
clear;
close all;

[x, y, z] = graphics_spharm;

% Notice the use of a handle 'h_fig' so we may refer later
h_fig = figure(1);
clf(h_fig);

% Plot the surface
h_plot = surf(x, y, z);

% Make the axis tight (set axis limits to min/max of surface) and equal
% (aspect ratios of x,y,z axes are the same so spheres look like spheres)
axis tight equal

% Make the background black. Note that gcf is the current graphics handle.
% Alternatively, we could have used set(h_fig, ...)
set(gcf, 'Color', 'k');

% Make the axis background black. Note gca = current axis handle
set(gca, 'Color', 'k', 'XColor', 'w', 'YColor', 'w', 'ZColor', 'w');

% Set the figure size to something bigger, i.e. a 800 by 800 pixel window.
% The first two numbers [200, 200] refer to the lower left position of the
% window.
set(gcf, 'Units', 'pixels', 'Position', [200 200 800 800]);
```



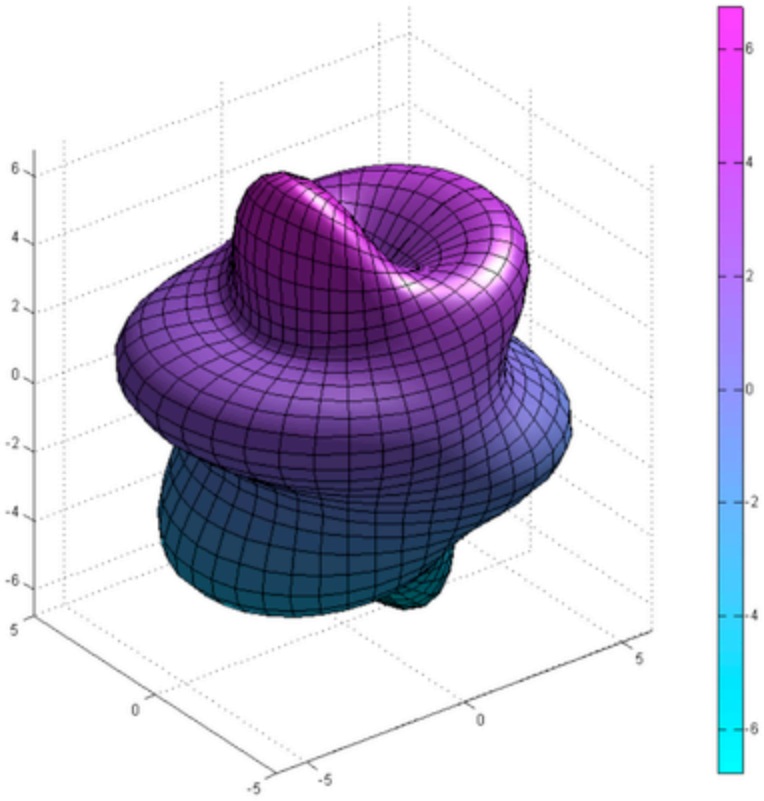
Lights!!!

```
% Create a light
light

% Create more realistic surface lighting using the 'phong' algorithm
lighting phong
```



Colours!!!

```
% Use a cool map and add a colorbar
colormap(cool);
colorbar;
```

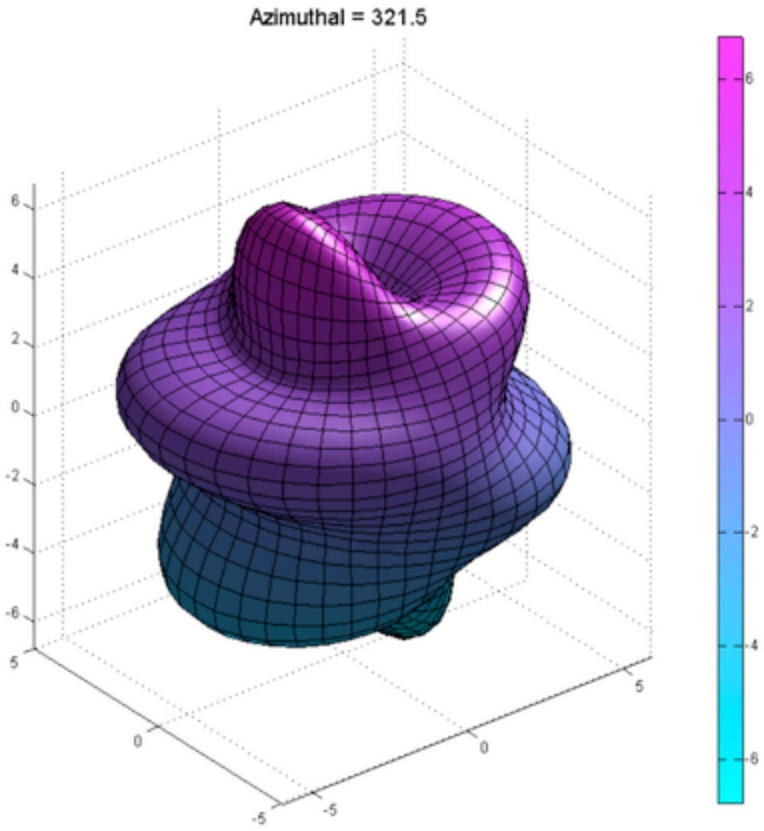


Camera and action!!!

Let's do some rotation. The command *__view__* controls the position of the camera. The view can be specified using spherical coordinates, and by specifying the aximuthal (horizontal) rotation and vertical elevation.

```
[az el] = view;
disp(['Current aximuthal = ', num2str(az)]);
disp(['Current elevation = ', num2str(el)]);

for j = 0:359
    view([az+j, el]);    % Move the camera
    drawnow;             % Refresh the output
    title(['Azimuthal = ', num2str(az+j)], 'FontSize', 18, 'Color', 'w');
end
```

```
Current aximuthal = -37.5
Current elevation = 30
```



Graphics Output

Matlab has very limited (read: sucky) ability to output graphics. Whether you are looking to output to pdf or postscript (eps, ps, etc.) for publication graphics, or jpeg, png, gif for movies or presentations, you are almost always better off using a third-party code or software.

Third party codes generally make use of ghostscript, which is a (free) suite software used to deal with images (mainly, to convert, crop, etc.). The best code I have seen thus far (and use myself) is Oliver Woodford's export_fig code, which can be downloaded here:

http://www.mathworks.com/matlabcentral/fileexchange/23629

```
% If you insist on creating graphics with Matlab, use the print command
print -dpng mysurf.jpg
```

Published with MATLAB® 7.12