Graphing Polar Equations in Matlab

Math 50C — Multivariable Calculus
David Arnold
August 1997

Abstract. Two techniques are used to graph polar equations of the form $r = f(\theta)$. Matlab's `polar` command is introduced and utilized. An alternative approach uses the polar to Cartesian transformations $x = r\cos\theta$ and $y = r\sin\theta$ and Matlab's `plot` command. Matlab script files enable students to organize and save their work for later use.
Prerequisites. Familiarity with vector operations in Matlab, particularly Matlab's element-wise operators. Some familiarity with Matlab's `plot` command is useful.

Matlab's Polar Command

The Matlab routine `polar` enables users to easily draw the graphs of polar equations having the form $r = f(\theta)$. For help on Matlab's `polar` command, type `help polar` at the Matlab prompt. You will find that the background of the plot is strikingly similar to the look and feel of polar coordinate graph paper.

Example *Sketch the graph of* $r = \sin 2\theta$.

Begin by defining a range for θ with Matlab's `linspace` command . The command `linspace(a,b,n)` will produce `n` equally spaced points between `a` and `b`. If you don't specify `n`, as in `linspace(a,b)`, Matlab defaults to 100 equally spaced points.

```
>> theta=linspace(0,2*pi);
```

Define r in terms of θ.

```
>> r=sin(2*theta);
```

Finally, use Matlab's `polar` command. The following command should produce an image similar to that in Figure 1.

```
>> polar(theta,r)
```

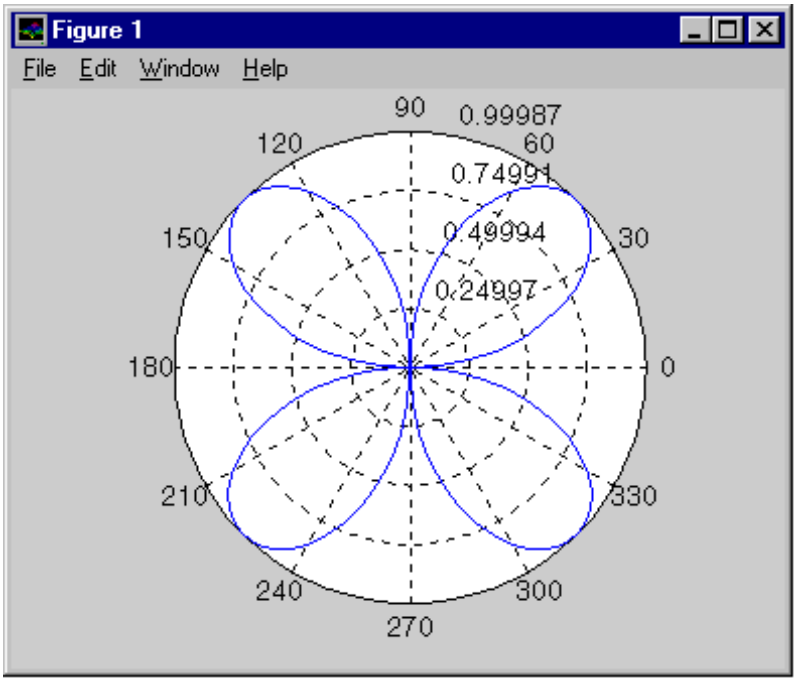

Figure 1.

Polar to Cartesian Transformations

The polar to Cartesian transformations

$$x = r\cos\theta$$
$$y = r\sin\theta$$

enable the user to employ Matlab's `plot` command.

Example *Use Matlab's* `plot` *command to sketch the graph of* $r = \sin 2\theta$.

Proceed as in the previous example. First define a range for θ, then compute r for each entry in the *vector* θ.

```
>> theta=linspace(0,2*pi);
>> r=sin(2*theta);
```

Next, use the polar to Cartesian transformations. In computing $x = r\cos\theta$, note that `r` and `cos(theta)` are *vectors*. Therefore, you must use Matlab's element-wise multiplication operator ("dot-star") to perform the multiplication of `r` and `theta`. A similar computation is used to compute $y = r\sin\theta$.

```
>> x=r.*cos(theta);
>> y=r.*sin(theta);
```

Matlab's `plot` command should produce an image similar to that in Figure 2.

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

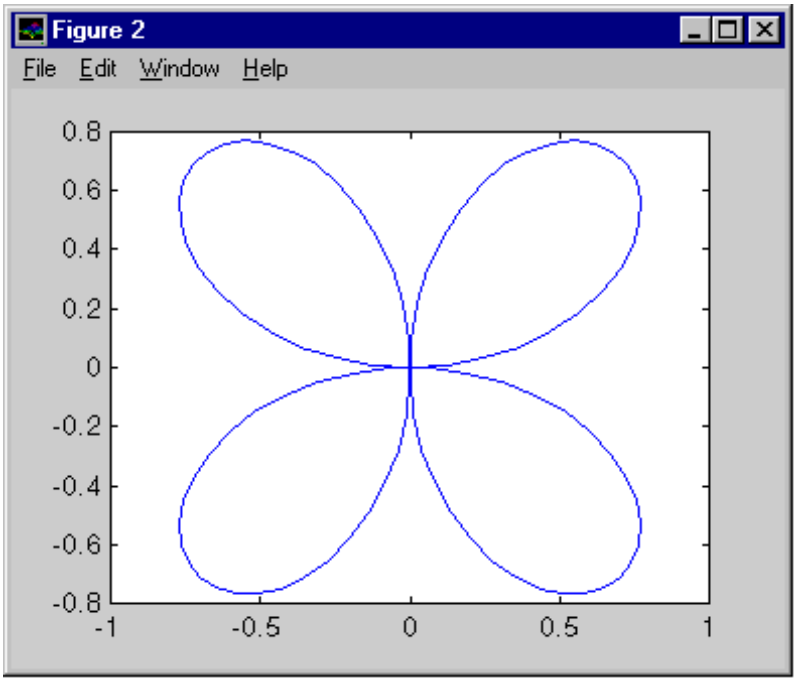

Figure 2.

Script Files in Matlab

Matlab script files offer a more efficient working environment and the opportunity to save your work for future use. Most users use Matlab's built-in editor-debugger to create script files. You can start the editor by using the mouse to click on the new document icon on the Matlab toolbar shown in Figure 3.

Figure 3. Click on first icon to start editor.

You can also start the editor by selecting File, New, M-file, as shown in Figure 4.

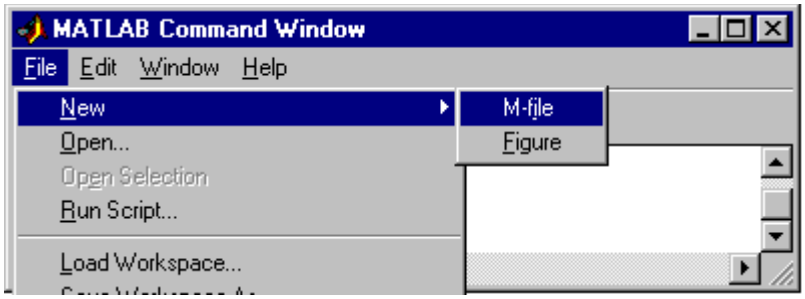

Figure 4. Starting the Matlab's editor.

Enter the code as shown in the editor in Figure 5.

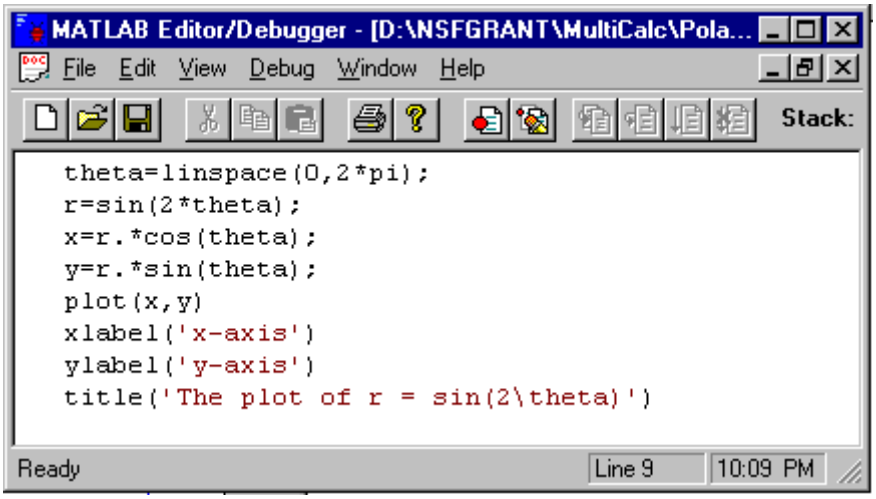

Figure 5. Enter the code shown.

While still in the debugger, click on File, then Save, and save the file as `mypolar.m` as shown in Figure 6.

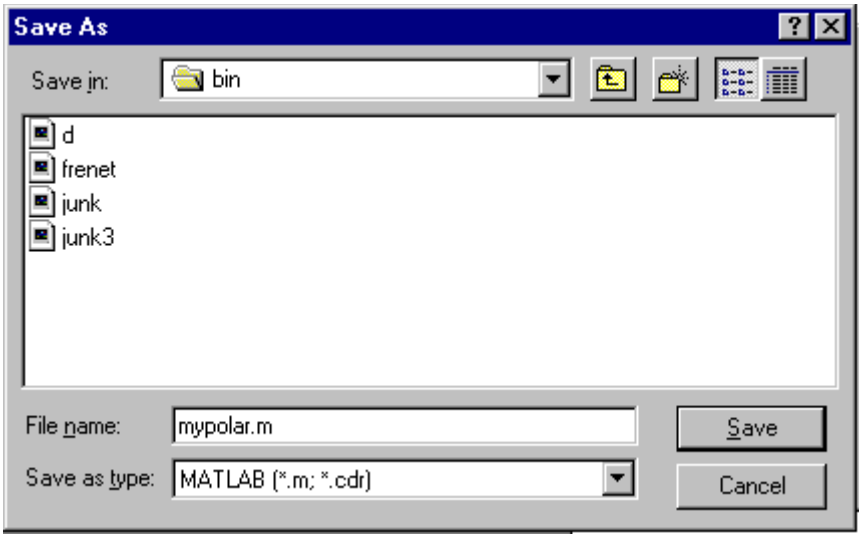

Figure 6. Saving your script file.

You can save the file in a directory of your choice. However, Matlab will not recognize the script file unless it is either (1) saved in the current working directory, or (2) the directory exists on the Matlab path.

- If you are working on the network at CR, click on the down-arrow in the Save As dialog box shown in Figure 6 and select the `h:` drive (your personal home directory). Save your file in this directory as `mypolar.m`. Return to Matlab and type `cd h:` at the Matlab prompt. This will make your personal directory the current directory, which guarantees that Matlab will find any file saved in your personal directory on the network.
- If you wish, you can also put a disk in the `a:` drive, save all files to your disk, then use the Matlab command `cd a:` to make the `a:` drive the current directory. This will lead to some degradation in performance as floppy drives are notoriously slower than hard drives.
- Typing `pwd` at the Matlab prompt will reveal the present working directory (current directory).

Should the need arise, you can change the current directory with Matlab's `cd` command. For example, `cd c:\mike\mfiles` makes the subdirectory `mfiles` in the directory `mike` on the `c:` drive the current directory. Type `help cd` for more help on this command.

- If you are working at home, save the file in a directory of choice, then use the Path Browser to add this directory to Matlab's path. Be sure to save your settings before exiting the Path Browser. *Important Note*: *Adding your h*: *drive to the path is causing problems on the school network and should be avoided*.

After saving your file and adjusting the path or current directory, you can return to Matlab and enter the command `mypolar` at the prompt.

```
>> mypolar
```

This should produce an image similar to that in Figure 7.

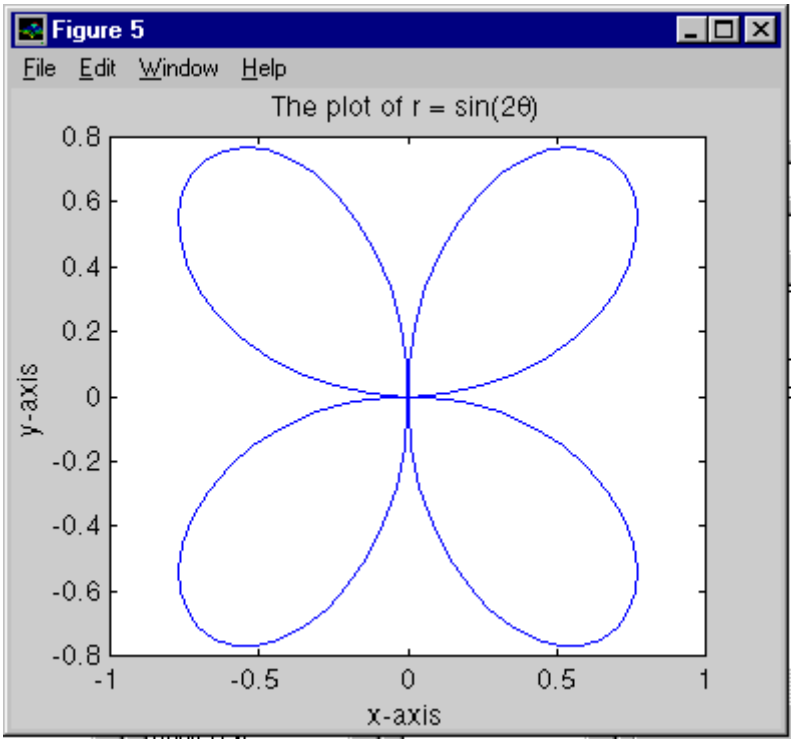

Figure 7. Plot produced by `mypolar`.

The advantage of script files is clear. Should you make a mistake, or wish to add further commands to your script file, simply edit and resave the file. Return to the Matlab prompt and type `mypolar` to examine the resulting changes.

Homework

Work through all of the examples in this activity. Bring the following items to the next class.

1. Use the Matlab editor to obtain a printout of the script file in Figure 5.
2. Obtain a printout of the image in Figure 7.