

Worksheet 11

November 15, 2009

1 Introduction

In the last class, you learned about using `ode45` on systems of first order equations. This is a really wonderful technique. All, n -th order odes can be broken down into a set of n first order odes. In this lab, we will learn a little more about matrices. We will then take that knowledge and create movies of pendulum motion.

2 Objective

We will learn about a class of matrices known as orthogonal matrices, and we will learn to use a very special orthogonal matrix to help visualize the pendulum problems we have been solving in previous worksheets. We will also be introduced to the basic code needed to run a movie in Matlab.

3 Rotation Matrices

In the space of 2×2 matrices, there are many patterns that reoccur. These matrices usually get some special name attached to them. You can spend what feels like a life time just studying certain classes of matrices and their applications. To name a few, Toeplitz, Hankel, Circulant, Hermitian, and Orthogonal. We will simply be looking at orthogonal matrices. We will really just be looking at one in particular. So what is an orthogonal matrix? The simple definition is a matrix whose transpose is its inverse (i.e. $A^T A = A A^T = I$). If we take the determinate of $A A^T$ we will easily see that A must have determinate ± 1 . For us, we will be using a special orthogonal matrix. It is the rotation matrix. The rotation matrix does exactly what it sounds like it does. It takes a vector and rotates it by some angle θ . The matrix in two and three dimensions is as follows:

$$R(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$$

$$R(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{pmatrix}$$

4 Movies in Matlab

There is an m-file given in conjunction with this lab that is supposed to be the movie file created for you already. We simply just need you to add in the missing pieces where ever you see comments for it. The basic idea of movies in Matlab is that we use a `getframe` command to collect each of the frames that will make up the movie we want to play. It often takes a clever manipulation to get the frames you want, but the basic code is given in the m-file for you to adjust for the purpose of the homework. As a quick example, we present the following code:

Animate the peaks function as you scale the values of Z:

```
Z = peaks;
surf(Z);
axis tight
set(gca,'nextplot','replacechildren');
% Record the movie
for j = 1:20
surf(sin(2*pi*j/20)*Z,Z)
F(j) = getframe;
end
% Play the movie ten times
movie(F,10)
```

5 Homework

The homework is very simple this week. Simply create a file for each solution of the odes last week, and create a movie from the solution via the rotation matrix. You should have 4 m-files this week. Two per ode.