Second Year Engineering Mathematics Laboratory

Hilary Term 2000

Ian Reid, 7/1/2000

Exercise 5: Eigensystems

The purpose of this exercise is to explore eigenvalues and eigenvectors. We begin by implementing the Rayleigh method for computing eigenvalues (and eigenvectors), then look at a couple of applications (out of the many which can be encountered in engineering).

1 Preparation

A. Modes of vibration

Consider the four unit masses mutually connected by springs in a square as shown in figure 1. The horizontal and vertical springs have unit stiffness and the diagonals are four times as stiff. Assume that zero potential is when the springs have zero extension, and that the spring extensions are small compared with their lengths.

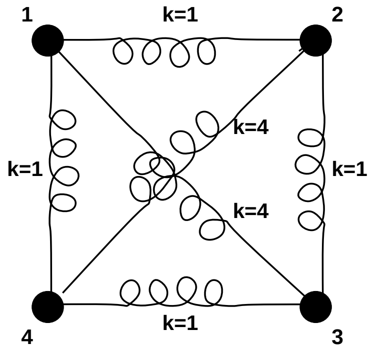


Figure 1:

Let us assume small motions. This then permits us to assume that the angles of the springs remain approximately constant relative to one another, and that the extension of the diagonal members is given to first order by:

$$\frac{\delta x + \delta y}{\sqrt{2}}$$

where δx and δy are the (positive) extensions in the x and y directions (optional exercise: prove this by Taylor expansion).

Hence show that show that the motion can be described by the differential equation

$$\ddot{\mathbf{x}} + \mathbf{K}\mathbf{x} = 0$$

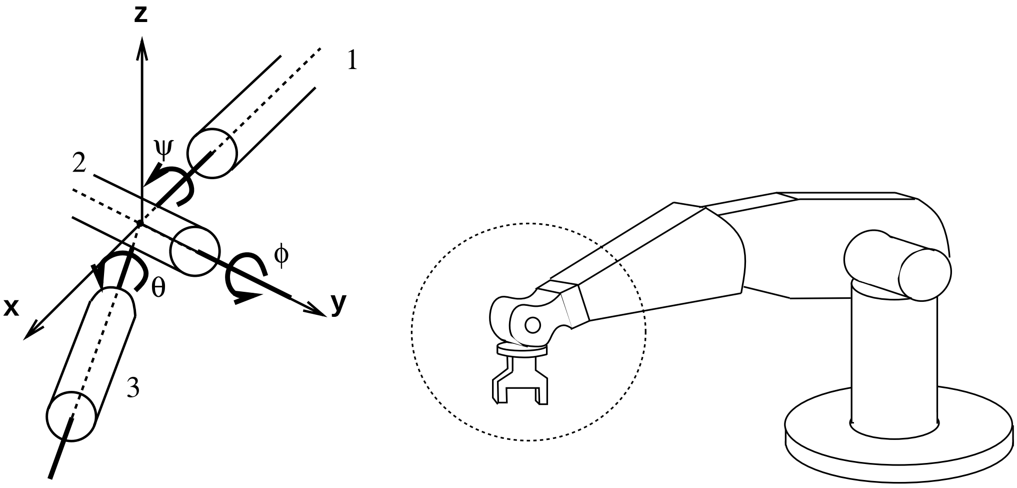
where

$$\mathbf{K} = \begin{bmatrix} 3 & -2 & -1 & 0 & -2 & 2 & 0 & 0 \\ -2 & 3 & 0 & 0 & 2 & -2 & 0 & -1 \\ -1 & 0 & 3 & 2 & 0 & 0 & -2 & -2 \\ 0 & 0 & 2 & 3 & 0 & -1 & -2 & -2 \\ -2 & 2 & 0 & 0 & 3 & -2 & -1 & 0 \\ 2 & -2 & 0 & -1 & -2 & 3 & 0 & 0 \\ 0 & 0 & -2 & -2 & -1 & 0 & 3 & 2 \\ 0 & -1 & -2 & -2 & 0 & 0 & 2 & 3 \end{bmatrix} \quad \text{and} \quad \mathbf{x} = \begin{bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ x_3 \\ y_3 \\ x_4 \\ y_4 \end{bmatrix}$$

with x_i and y_i the (small) horizontal and vertical displacements of mass i from zero potential.

B. Rigid transformations

Figure 1 shows the wrist of a Puma robot. Its orientation is determined by three joint angles as shown. Assume that the origin is at the mutual intersection of the three rotation axes, that the x-axis is parallel to rotation axis 1, and the y-axis is parallel to rotation axis 2.



Show that the rotation matrix which describes the overall orientation of the wrist in terms of θ, ϕ and ψ is given by

$$\mathbf{R} = \begin{bmatrix} \cos\phi & \sin\phi\sin\theta & \sin\phi\cos\theta \\ \sin\psi\sin\phi & \cos\psi\cos\theta - \sin\psi\cos\phi\sin\theta & -\cos\psi\sin\theta - \sin\psi\cos\phi\cos\theta \\ -\cos\psi\sin\phi & \sin\psi\cos\theta + \cos\psi\cos\phi\sin\theta & -\sin\psi\sin\theta + \cos\psi\cos\phi\cos\theta \end{bmatrix}$$

The easiest way to show this is to consider what the wrist looks like when all three angles are zero – in this configuration note that the rotation axes for links one and three are *coincident*. Now start at the end of the kinematic chain (link 3) and work your way back to link 1 composing the rotation along the way. The reason for doing it this way around is that a rotation about link3 *has no effect on the rotation axes higher up the chain*.

2 Lab Work

A. Computing eigenvalues/eigenvectors

In lectures you will probably have seen the power method for finding the eigenvalues and eigenvectors of a symmetric matrix. You will find an implementation of this method in the directory /packages/demo/comp/lab5 in a file called pow.m. Copy it across to your lab5 directory and try it out:

```
>> A = [ 2 2 6  6; 2 16 9 2; 6 9 8 7; 6 2 7 0 ];
>> [lambda,x] = pow(A,[1 1 1 1]',20)
```

Verify that the code is correct by comparing the answer to the result from **matlab**'s built in function **eig** (type help eig to see how to use this function).

Modify the code so that it implements the Rayleigh method instead and verify that it works for the matrix above.

Now use your Rayleigh code to find the largest eigenvalue of the matrix

$$\mathbf{B} = \begin{bmatrix} 1.8750 & 0.3062 & 0.1250 \\ 0.3062 & 1.2500 & -0.3062 \\ 0.1250 & -0.3062 & 1.8750 \end{bmatrix}$$

Try it for various starting vectors and note down what you observe about the estimated eigenvalues and particularly the eigenvectors. Explain what is going on.

Now write a simple **matlab** routine to deflate a matrix by a given eigenvalue/eigenvector pair using the (inferior but simpler) Hotelling deflation. Create a function

```
function Anew = deflate(A, lambda, v)
    Anew = ???????;
```

in a file `deflate.m`.

Try it out with the 4×4 matrix **A** above to obtain all its eigenvalues (and eigenvectors).

B. Modes of vibration

Consider again the four connected masses shown in figure 1. Use your programs from the previous section to compute the three largest eigenvalues, and the corresponding eigenvectors of the matrix $\mathbf{K}$.

We will now use some of the power of matlab to create an animation of these modes of vibration. Copy the files `ode.m` and `anim.m` from the usual place into your `lab5` directory.

Have a look at the code in `ode.m` and write down in your log-book the maths which this code is implementing to solve the ODE $\ddot{\mathbf{x}} = -\mathbf{K}\mathbf{x}$.

Now run it, using one of the eigenvectors of **K** (just computed above) as a starting point. The solution will be an oscillation whose form is determined by the eigenvector, and whose frequency is given by the square-root of the eigenvalue. The third argument to **ode** is a time over which to solve the equation. From the eigenvalue, determine a time which will correspond to exactly one period of the oscillation.

```
>> T = ??????;       % time
>> v = ??????;       % eigenvector
>> [t,y] = ode(K,0.2*v,T);
```

You can view the solution using a plot command

```
>> plot(t,y)
```

but rather more compelling is to create an animation. Have a look at the code in `anim.m` and try to understand it. Now create a movie by typing the **matlab** command

```
>> m = anim(t,y);
```

(don't forget the ";" or you will get lots of nonsense spewed out on the screen). The movie is stored in the matrix `m` with one frame per column. The command

```
>> movie(m,10)
```

will play the movie 10 times.

Try this for various eigenvectors, and for a more general starting vector – for example, try displacing just the bottom left mass – and record your observations in your log-book.

Explain the modes of vibration corresponding to the zero eigenvalues.

C. Rotations

Recall figure 1 showing the wrist of a Puma robot. Determine the effective axis and angle of rotation of the composite rotation which specifies the wrist orientation given by joint angles $\theta = \pi/6, \phi = \pi/4, \psi = \pi/3$. To do this, recall that the eigenvector corresponding to the real eigenvalue of a rotation matrix gives the axis, and that the the complex eigenvalues of the matrix are $\exp(\pm j\alpha)$ where α is the angle of rotation about the axis. Since we are now dealing with complex eigenvalues and eigenvectors, your Rayleigh implementation is inappropriate, so we use instead, the rather more sophisticated version supplied in **matlab** (type `help eig` to see how to use it).

What is the rotation matrix?

What is the axis of rotation, and the angle of rotation around this axis?