Introduction to Computation in Matlab

Prof. Brian L. Evans, Dept. of ECE, The University of Texas, Austin, Texas USA

Matlab's forte is numeric calculations with matrices and vectors. A vector can be defined as

vec = [1 2 3 4];

The first element of a vector is at index 1. Hence, vec(1) would return 1. A way to generate a vector with all of its 10 elements equal to 0 is

zerovec = zeros(1,10);

Two vectors, **a** and **b**, can be used in Matlab to represent the left hand side and right hand side, respectively, of a linear constant-coefficient difference equation:

a(3) y[n-2] + a(2) y[n-1] + a(1) y[n] = b(3) x[n-2] + b(2) x[n-1] + b(1) x[n]

The representation extends to higher-order difference equations. Assuming zero initial conditions, we can derive the transfer function. The transfer function can also be represented using the two vectors **a** (negated feedback coefficients) and **b** (feedforward coefficients). For the second-order case, the transfer function becomes

$$H(z) = \frac{b(1) + b(2)z^{-1} + b(3)z^{-2}}{a(1) + a(2)z^{-1} + a(3)z^{-2}}$$

We can factor a polynomial by using the roots command.

Here is an example of values for vectors **a** and **b**:

a = [1 6/8 1/8];
b = [1 2 3];

For an asymptotically stable transfer function, i.e. one for which the region of convergence includes the unit circle, the frequency response can be obtained from the transfer function by substituting z = exp(j ω). The Matlab command freqz implements this substitution:

[h, w] = freqz(b, a, 1000);

The third argument for freqz indicates how many points to use in uniformly sampling the points on the unit circle. In this example, freqz returns two arguments: the vector of frequency response values **h** at samples of the frequency domain given by **w**. One can plot the magnitude response on a linear scale or a decibel scale:

plot(w, abs(h));
plot(w, 20*log10(abs(h)));

The phase response can be computed using a smooth phase plot or a discontinous phase plot:

plot(w, unwrap(angle(h)));
plot(w, angle(h));

One can obtain help on any function by using the help command, e.g.

help freqz

As an example of defining and computing with matrices, the following lines would define a 2 x 3 matrix **A**, then define a 3 x 2 matrix **B**, and finally compute the matrix **C** that is the inverse of the transpose of the product of the two matrices **A** and **B**:

```
A = [1 2 3; 4 5 6];
B = [7 8; 9 10; 11 12];
C = inv((A*B)');
```

Matlab Tutorials and Availability

Here are excellent Matlab tutorials:

1. UT Austin: http://ssc.utexas.edu/training/software-tutorials#matlab
2. Mathworks: http://www.mathworks.com/academia/student_center/tutorials/

The following Matlab tutorial book is a useful reference:

Duane C. Hanselman and Bruce Littlefield, *Mastering MATLAB*, ISBN 9780136013303, Prentice Hall, 2011.

Matlab is available in the ECE Learning Resource Centers and through remote login. A student version of Matlab may be purchased at the bookstore for roughly $100.

Although the first few computer homeworks will help step you through Matlab, it is strongly suggested that you take the short courses that the Division of Statistics and Scientific Computing will be offering. The schedule of those courses is available online at

http://ssc.utexas.edu/training/software-short-courses

Technical support is provided through free consulting services from the Division of Statistics and Scientific Computation. Simple queries can be e-mailed to stats@ssc.utexas.edu. For more complicated inquiries, please go in person to their offices located in GDC 7.504. You can walk in or schedule an appointment online.

Running Matlab in Unix

On the Unix machines in the ECE Learning Resource Center, you can run Matlab by typing

```
module load matlab
matlab
```

When Matlab begins running, it will automatically execute the commands in your Matlab initialization file, if you have one. On Unix systems, the initialization file must be ~/matlab/startup.m where ~ means your home directory.