Modelling and control summaries

by Anthony Rossiter

USE OF MATLAB 2 – transfer functions

OVERVIEW: These notes gives a very narrow view of MATLAB and how to do a limited number of things. In general students need to become effective independent learners of MATLAB.

MATLAB NOTATION: Matlab has a very efficient mechanism for storing transfer functions and students are advised to use this, that is **tf**.

Although alternatives exist, I recommend using tf properly as it is then easier to embed the building of transfer functions into scripts and other files you may need later.

NOTE: MATLAB distinguishes between Laplace transforms and transfer functions by how you use the object. The object in itself could be either.

Students should be familiar with MATLAB notation for handling polynomials as this is assumed.

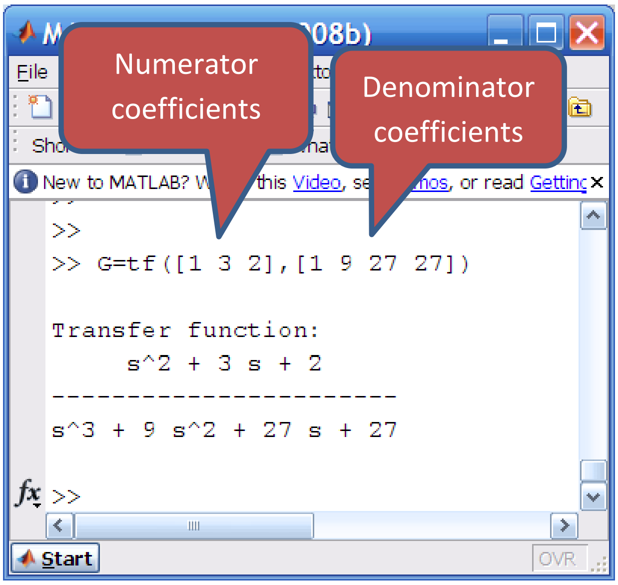
CREATING A TRANSFER FUNCTION OBJECT

1. A transfer function has a numerator polynomial and a denominator polynomial.
2. Represent both polynomials using vectors with standard MATLAB notation.
3. Enter both polynomials into tf.

Consider

$$G=\frac{s^2+3s+2}{s^3+9s^2+27s+27}$$

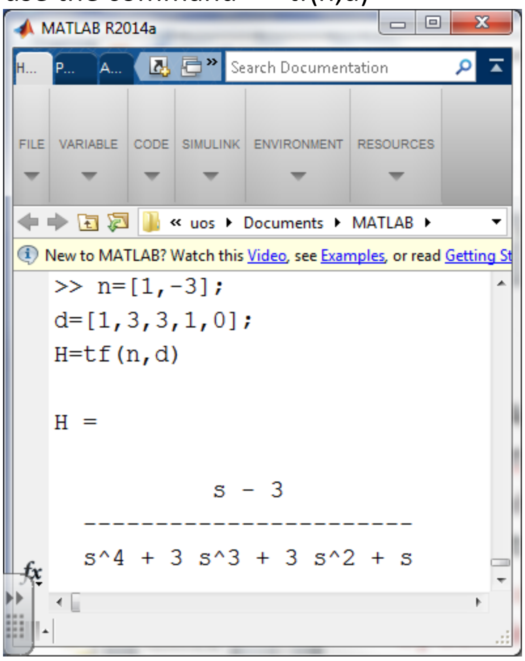
Numerator coefficients are n=[1 3 2]
Denominator coefficients are d=[1 9 27 27]
tf takes these vectors using the command tf(n,d)



Consider

$$G=\frac{3-s}{s^4+3s^3+3s^2+s}$$

Numerator coefficients are n=[-1,3]
Denominator coefficients are d=[1,3,3,1,0]
use the command tf(n,d)

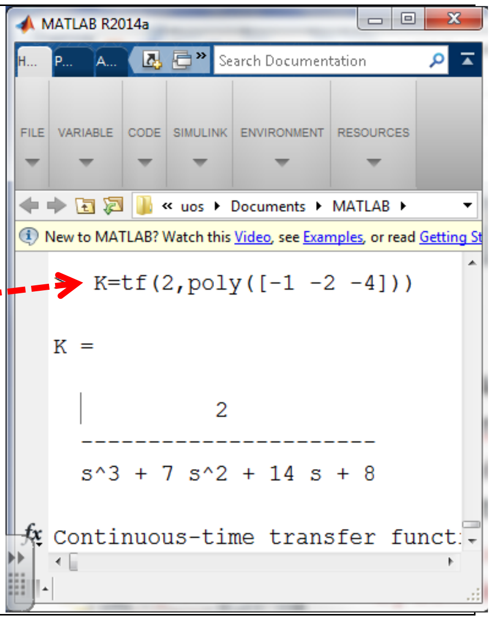


COMBINING POLYNOMIAL SHORT CUTS WITH TF

Assume that you know the poles of a transfer function, but not the denominator coefficients. You can still create the transfer function in a single line by embedding one MATLAB command within another.

The command poly([-1 -2 -4]) creates a polynomial with roots at -1,-2 and -4 and hence $(s+1)(s+2)(s+4)=s^3+7s^2+14s+8$

An example is given here.



EXTRACTING DATA FROM A TRANSFER FUNCTION OBJECT

The user may wish to extract different properties from an object such as:

1. Numerator/denominator coefficients
2. Poles and zeros.

To extract the coefficients use the command **tfdata**. However, note that a specific syntax is required to extract the coefficients as normal MATLAB vector array.

Users may note that in this example the numerator vector has been padded with extra zeros at the front – clearly these do not effect the implied polynomial.