Some Matlab Notes

Roots of a Polynomial

\((x+1)(x-1) = x^2 - 1 = 0\)

\[ p = \begin{bmatrix} 1 & 0 & -1 \end{bmatrix} \]

\[ r = \text{roots}(p) \implies 1, -1 \]

Multiplication of Polynomials

\((x+1)(x-1) \implies ?\)

\[ p_1 = \begin{bmatrix} 1 & 1 \end{bmatrix}; \quad p_2 = \begin{bmatrix} 1 & -1 \end{bmatrix}; \]

\[ c = \text{conv}(p_1, p_2) \]

\[ \text{ans} = \begin{bmatrix} 1 & 0 & -1 \end{bmatrix} \]  

[to divide use \text{deconv}]

Poles / Residues of Rational Fraction Polynomial

\[ [r, p, k] = \text{residue}(n, d) \]

\[ \text{direct term} \quad \text{denominator} \]

\[ \text{pole} \quad \text{residue} \]

\[ \frac{-0.5}{s - (-1)} + \frac{0.5}{s - (1)} \]

\[ = -0.5s + 0.5 + 0.5s + 0.5 \]

\[ \frac{(s+1)(s-1)}{(s+1)(s-1)} \]

Solve a Polynomial

\[ \text{sym} s; a, b, c, x \]

\[ x = \text{solve}(a * x^2 + b * x + c) \]

or \[ x = \text{solve}(a * x^2 + bx + c = 0) \]
Some MATLAB Notes (McKelligon)

General
\[ \frac{dy}{dt} + y = 0 \]

\[ \text{DSOLVE}('DY + Y = 0', 't') \]

Solution
\[ c_1 e^{-t} \]
\[ \frac{dy}{dt} = -c_1 e^{-t} \]
\[ -c_1 e^{-t} + c_1 e^{-t} = 0 \quad \text{CHECK} \]

Now apply I.C. \( y_0 = 1 \)

\[ y = \text{DSOLVE}('DY + Y = 0', 'Y(0)=1', 't') \]
\[ \text{ezplot}(y, 0:60) \]

Now \( x = \text{DSOLVE}('M*DX + C*DX + K*X = 0', 'x') \)
\[ x = \text{simple}(x) \]
\[ \text{pretty}(x) \]

\[ x = \text{dsolve}('M*D2X + C*DX + K*X = 0', 'DX(0)=0', 'X(0)=1', 'x') \]

characteristic equation

\[ \text{ezplot}(x, 0:60) \quad \text{DOESN'T WORK} \]

But

\[ y = \text{dsolve}('M*DY + L*DY + 100*K = 0', 'DY(0)=0', 'Y(0)=1', 'y') \]
\[ \text{ezplot}(y, 0:60) \quad \text{WORKS} \]
TRANSPORT RESPONSE IN MATLAB

\[ \frac{Y(s)}{U(s)} = \frac{25}{s^2 + 4s + 25} \]

\[ \text{NUM} = [0 \quad 0 \quad 25] \]
\[ \text{DEN} = [1 \quad 4 \quad 25] \]
\[ \text{STEP (NUM, DEN)} \quad \text{OR} \quad \text{STEP (NUM, DEN, \epsilon)} \]

**NOTE:** If both hard arguments are used

then plot is not automatically generated

\[
Y = \text{STEP(NUM, DEN, \epsilon)} \quad \text{OR} \quad [Y, X, \epsilon] = \text{STEP(NUM, DEN, \epsilon)}
\]

\[ \text{IMPULSE (NUM, DEN)} \]

\[ \text{GRID; \quad \% \quad adds \quad grid \quad to \quad plot} \]
\[ \text{TITLE ('TITLE TO PLOT')} \]

RAMP RESPONSE

\[
\frac{Y}{U} = \frac{1}{s^2 + 5s + 1} \quad \text{RAMP} \quad \frac{1}{s^2} \quad \text{STEP} \quad \frac{1}{s}
\]
\[
\therefore \quad \frac{Y}{U} = \frac{\frac{1}{s^2 + 5s + 1} \cdot \frac{1}{s}}{\frac{1}{s}}
\]

Then the transfer function is \( \frac{1}{s^2 + 5s + 1} \)
TRANSIENT RESPONSE - INITIAL CONDITION

Assume \( mx'' + cx' + kx = 0 \) with \( x(0) = 0.1 \text{m} + \dot{x}(0) = 0.05 \text{m/s} \)

\[ m = 1, \quad c = 3, \quad k = 2 \]

Laplace \((ms^2 + cs + k)x(s) = (ms + c)x_0 + mx_0\)

Substituting

\[ x(s) = \frac{MSx_0 + c x_0 + mx_0}{ms^2 + cs + k} = \frac{0.1s + 3.5}{s^2 + 3s + 2} \]

which can be written as

\[ x(s) = \frac{0.1s^2 + 0.35s}{s^2 + 3s + 2} \]

The step response of \((0.1s^2 + 0.35s)/(s^2 + 3s + 2)\)

is equivalent solution to initial condition
Root Locus in MATLAB

Rlocus(num, den)

Bode [mag, phase, w] = bode(num, den, w)
magdb = 20 * log10(mag)

Nyquist [re, im, w] = nyquist(num, den, w)
Bode plots in MATLAB

Magnitude and phase angles generated with

> bode(mom, den, w)

or

> [mag, phase, w] = bode(mom, den, w)

The magnitude can be converted to dB

> mag dB = 20* log 10(mag)

To specify the range, use command

logspace(d1, d2) or logspace(d1, d2, n)

n points generated between 10^{d1} and 10^{d2}

w = logspace(-1, 2)
generates 50 points between 0.1 and 100 rad/sec

w = logspace(0, 3, 100)
generates 100 points between 1 and 1000 rad/sec
NYQUIST PLOTS IN MATLAB

Real and Imag components of the FRF

> Nyquist (num, den)

or

> Nyquist (num, den, w)

With left hand arguments,

> [re, im, w] = Nyquist (num, den, w)

Example

> num = [0 0 1];
> den = [1 1 0];
> w = 0.1:0.1:100;
> [re, im, w] = Nyquist (num, den, w);
> plot (re, im)
> v = [-2 2 -5 5]; axis (v)
> grid
> title ('Nyquist Plot: G(s) = 1/[s(s+1)]')
> xlabel ('Real Axis')
> ylabel ('Imag Axis')
MATLAB for STATE-SPACE

Transfer Function to State Space

\[[A, B, C, D] = \text{tf2ss}(\text{num, den})\]

State Space to Transfer Function

\[[\text{num, den}] = \text{ss2tf}(A, B, C, D, iu)\]