

## Problem Set 2 SOLUTIONS

**Problem 1** – The MATLAB command “residue” can perform partial fraction expansion. (Type “help residue” at the MATLAB prompt for more info.) Below are some useful Laplace transforms and MATLAB commands which will find the required solutions.

	f(t)	F(s)
1	<b>d(t)</b> (unit impulse)	1
2	<i>u(t)</i> (unit step)	$\frac{1}{s}$
3	$t^n u(t)$	$\frac{n!}{s^{n+1}}$
4	$e^{lt} u(t)$	$\frac{1}{s - l}$
5	$t^n e^{lt} u(t)$	$\frac{n!}{(s - l)^{n+1}}$
6a	$e^{-st} \left[ A \cos(w_d t) + \frac{B - As}{w_d} \sin(w_d t) \right] u(t)$ <p style="text-align: center;">where: <math>w_d = \sqrt{w_n^2 - s^2}</math></p>	$\frac{As + B}{s^2 + 2s + w_n^2}$
6b	$K e^{-st} \cos(w_d t + f) u(t)$ <p style="text-align: center;">where: <math>w_d = \sqrt{w_n^2 - s^2}</math></p> $K = \text{sign}(A) \cdot \sqrt{A^2 + \left( \frac{B - As}{w_d} \right)^2}$ $f = \tan^{-1} \left[ \frac{-(B - As)/w_d}{A} \right] = \tan^{-1} \left[ \frac{s - (B/A)}{w_d} \right]$	$\frac{As + B}{s^2 + 2s + w_n^2}$

For all cases, the transfer function from input to output is:  $\frac{Y(s)}{U(s)} = \frac{1}{s^2 + 2s + 16}$

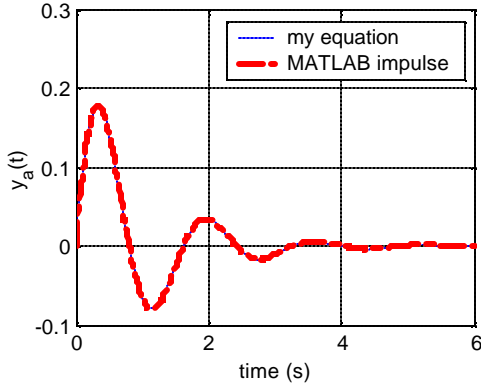
- a) unit impulse (row 1):  $U_a(s) = 1$ , so  $Y_a(s) = \frac{1}{s^2 + 2s + 16}$
- b) step of magnitude A (row 2):  $U_b(s) = \frac{A}{s}$ , so  $Y_b(s) = \frac{A}{s(s^2 + 2s + 16)}$
- c) unit parabola ( $t^2$ ) (row 3):  $U_c(s) = \frac{2}{s^3}$ , so  $Y_c(s) = \frac{2}{s^3(s^2 + 2s + 16)}$
- d) unit cosine (row 6a):  $U_d(s) = \frac{s}{s^2 + w_n^2}$ , so  $Y_d(s) = \frac{s}{(s^2 + w_n^2)(s^2 + 2s + 16)}$

Problem 1 (continued)

a)  $Y_a(s) = \frac{1}{s^2 + 2s + 16}$  Using row 6a,  $y_a(t) = e^{-t} \frac{1}{w_d} \sin(w_d t)$ ,

where  $w_d = \sqrt{16 - 1^2} = 3.873$  (rad/s).  $y_a(t) = 0.2582e^{-t} \sin(3.873t)$

Below are MATLAB commands. Note that the semicolon allows me to separate multiple commands to put them on a single line (just to save space!):



```
>> tau=1; t=6*tau*[0:.001:1];
>> figure(1); clf
>> plot(t,(1/sqrt(15))*exp(-t./tau).*sin(sqrt(15)*t),'b-')
>> grid on; xlabel('time (s)'); ylabel('y_a(t)')
>> hold on
>> [ya,ta]=impz(tf([1],[1 2 16]),t);
>> p1=plot(ta,ya,'r-')
>> set(p1,'LineWidth',2)
>> legend('my equation','MATLAB impulse')
```

I've intentionally introduced the MATLAB functions "impz" and "tf" above, which you should become familiar with, if you haven't already used them. (Use "help impz" and "help tf".) I've plotted my own solution against MATLAB's here as a check.

b)  $Y_b(s) = \frac{A}{s(s^2 + 2s + 16)}$ . First, we expand via partial fraction (using MATLAB), using

A=1 and then scaling every as necessary. MATLAB commands are shown below, at right, and give the following partial fraction expansion:

$$Y_b = A \left( \frac{-0.0625s - 0.125}{s^2 + 2s + 16} + \frac{0.0625}{s} \right)$$

Let's double-check this expansion in MATLAB:

```
>> tf1=tf(num,den)

Transfer function:
-0.0625 s - 0.125
-----
s^2 + 2 s + 16

>> tf2=tf(r(3),[1 0])

Transfer function:
0.0625
-----
s
```

```
>> tf1+tf2

Transfer function:
1
-----
s^3 + 2 s^2 + 16 s
```

```
>> [r,p]=residue([1],[1 2 16 0])

r =
-0.0313 + 0.0081i
-0.0313 - 0.0081i
0.0625

p =
-1.0000 + 3.8730i
-1.0000 - 3.8730i
0
>> num=r(1)*[1 -p(2)] + r(2)*[1 -p(1)]

num =
-0.0625 -0.1250

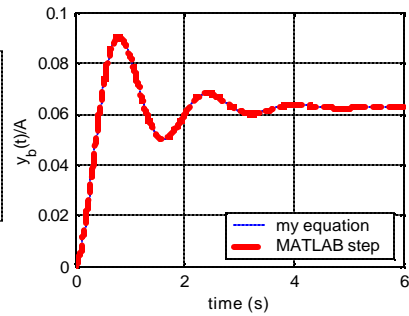
>> den=conv([1 -p(1)],[1 -p(2)])

den =
1.0000 2.0000 16.0000
```

Using row 2 (unit step) and either row 6a or 6b, we can find the total time response:

$$y_b(t) = A \left( \frac{1}{16} - 0.0645 e^{-t} \cos(3.873t - 0.2527) \right)$$

$$= A \left( \frac{1}{16} + e^{-t} [0.0625 \cos(3.873t) - 0.0161 \sin(3.873t)] \right)$$

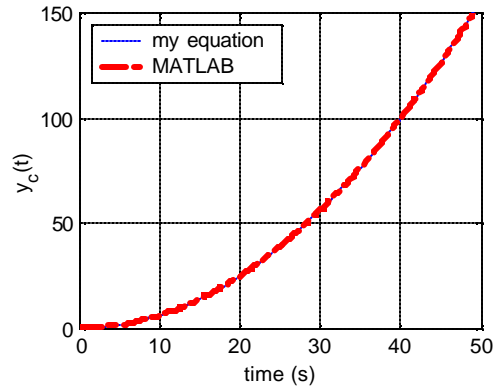
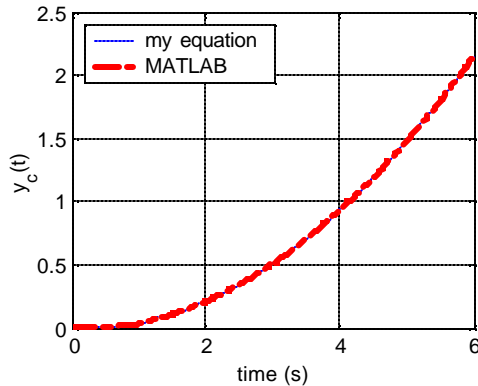


(c)  $Y_c(s) = \frac{2}{s^3(s^2 + 2s + 16)}$  This can be expanded into:

$$Y_c(s) = \frac{1}{8s^3} - \frac{1}{64s^2} - \frac{3}{512s} + \frac{3s+14}{512(s^2 + 2s + 16)}$$

Using rows 2, 3 and 6:

$$y_c(t) = \frac{1}{16} t^2 - \frac{1}{64} t - \frac{3}{512} + 0.0080687 e^{-t} \cos(3.873t - .7580)$$



(Approaches a parabolic response as t goes to infinity.)

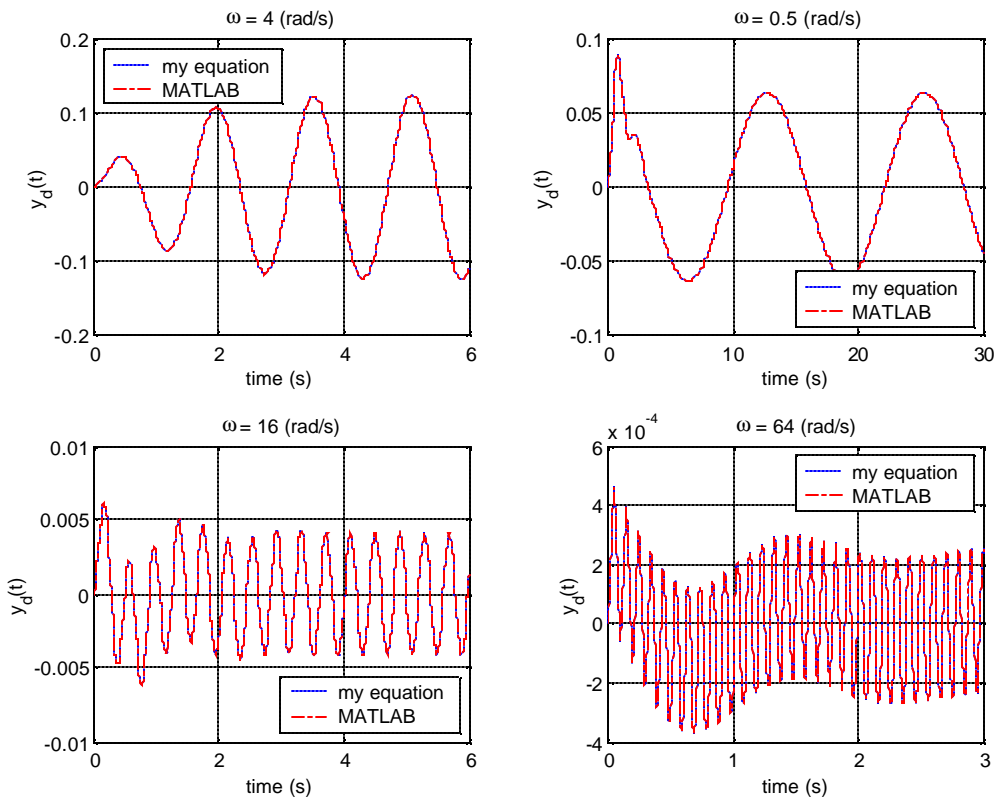
(d)  $Y_d(s) = \frac{s}{(s^2 + \omega_n^2)(s^2 + 2s + 16)}$  or  $Y_d(s) = \frac{As + B}{(s^2 + \omega_n^2)} + \frac{Cs + D}{(s^2 + 2s + 16)}$ ,

where  $A = \frac{(16 - \omega_n^2)}{(\omega_n^4 - 28\omega_n^2 + 256)}$ ,  $B = \frac{2\omega_n^2}{(\omega_n^4 - 28\omega_n^2 + 256)}$ ,  $C = \frac{(\omega_n^2 - 16)}{(\omega_n^4 - 28\omega_n^2 + 256)}$ ,

and  $D = \frac{-32}{(\omega_n^4 - 28\omega_n^2 + 256)}$ . Using row 6a,

$$y_d(t) = A \cos(\omega t) + \frac{B}{\omega} \sin(\omega t) + e^{-t} \left[ C \cos(3.873t) + \frac{D - C}{3.873} \sin(3.873t) \right]$$

Your plot will depend on your choice of  $\omega$ . I've plotted several options on the next page. (Note the time scales and y-axes are different, in general, from one to the next!)



**Problem 2 (Nise 2.8, plus find  $y(t)$ )** – Only ‘c’ was tricky, and you can use MATLAB to help with that partial fraction expansion. (No, it does not factor “nicely”, I know.)

a)  $\frac{X(s)}{F(s)} = \frac{1}{s^2 + 2s + 7}$  diff eq is:  $\ddot{x} + 2\dot{x} + 7x = f(t)$  Partial fraction expansion for  $X(s)$ ,

given a unit step in  $f(t)$  is:  $X(s) = \frac{1}{7} \left[ \frac{1}{s} - \frac{s+2}{s^2 + 2s + 7} \right]$

$$x(t) = \frac{1}{7} \cdot [1 - 1.0801e^{-t} \cos(\sqrt{6}t - .3876)] = \frac{1}{7} - .1543e^{-t} \cos(2.45t - .3876)$$

I sorta like the form on the lefthand side more, because you’ll generally expect to see something like the “static gain” (when all derivatives have gone to zero,  $x=(1/7)$  of  $f$ ) times the quantity “1 minus a cosine with frequency given by the “damped natural frequency” and with an amplitude somewhat greater than one and a phase shift between zero and negative  $\pi/2$ ”.

b)  $\frac{X(s)}{F(s)} = \frac{10}{(s+7)(s+8)}$  diff eq is:  $\ddot{x} + 15\dot{x} + 56x = 10f(t)$  Partial fraction expansion for

$X(s)$ , given a unit step in  $f(t)$  is:  $X(s) = \frac{10}{56} - \frac{10}{7(s+7)} + \frac{10}{8(s+8)}$ . Using row 4:

$$x(t) = \frac{10}{56} - \frac{10}{7}e^{-7t} + \frac{10}{8}e^{-8t}$$

Note the slope is zero at  $t=0$ . (Take a derivative...)

c)  $\frac{X(s)}{F(s)} = \frac{s+2}{s^3+8s^2+9s+15}$  diff eq is:  $\boxed{\frac{d^3x}{dt^3} + 8\frac{d^2x}{dt^2} + 9\frac{dx}{dt} + 15x = \frac{df(t)}{dt} + 2f(t)}$  Partial

fraction expansion is not pretty (again, given a step input in f(t)):

$$X(s) = \frac{2}{15s} + \frac{.016}{s+7.0226} - \frac{.1494s+.0334}{s^2+.9774s+2.136}$$

$$\boxed{x(t) = \frac{2}{15} + .016e^{-7.0226t} - .1521e^{-.4887t} \cos(1.3774t + .1903)}$$
 I can only believe my math

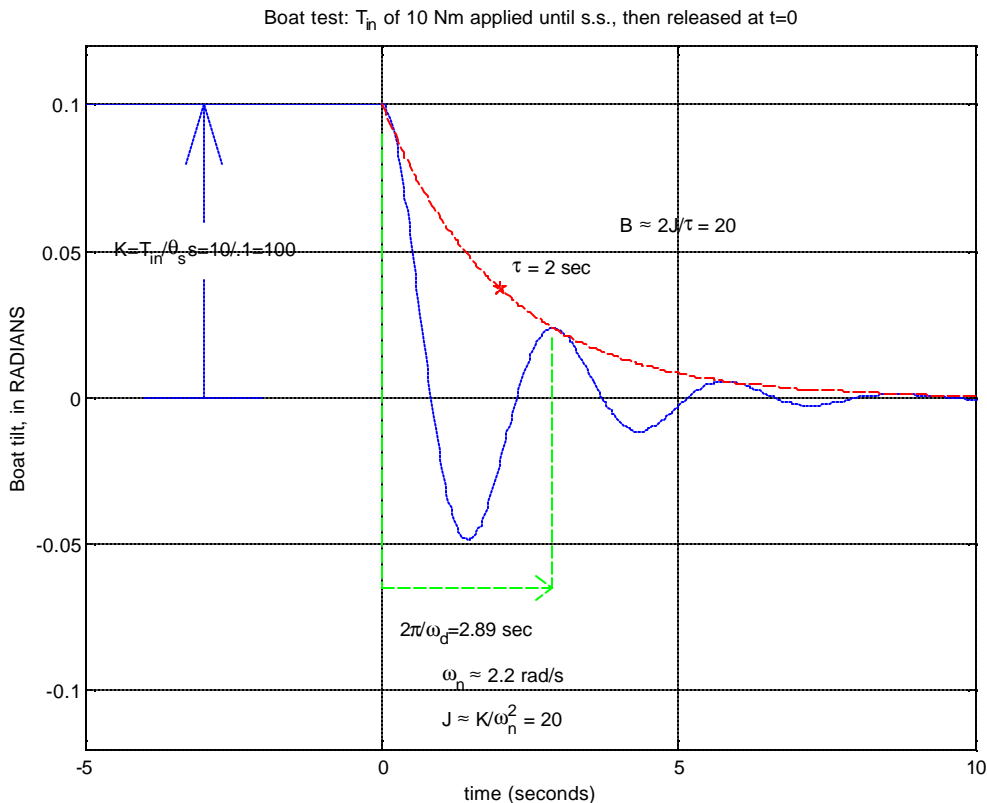
here, because the plot checks out against a MATLAB step response!

### Problem 3 – Ship roll stabilization

If we tap a boat that is floating in the water, we expect its response to show decaying sinusoidal oscillations. This suggests that a 2<sup>nd</sup>-order model may be appropriate. The boat has some effect inertia, J. Viscous damping (B) can be explained from the losses causes as various parts of the boats “swish” through the water, and the offset of the centers of buoyancy and gravity due to any “tilt” provide a spring force (K, which we can model as a linear spring, very similar to the spring effect of a pendulum for small theta). Input torques will come both from our commanded input torque T, applied by the fins, and from any disturbance torques, T<sub>d</sub>. The output will be the angle of the boat’s tilt, θ:

$$\boxed{J\ddot{q} + B\dot{q} + Kq = T_{in} + T_d}$$
 The transfer function from commanded input torque to output

theta is:  $\frac{\Theta(s)}{T(s)} = \frac{1}{Js^2 + Bs + K}$ .

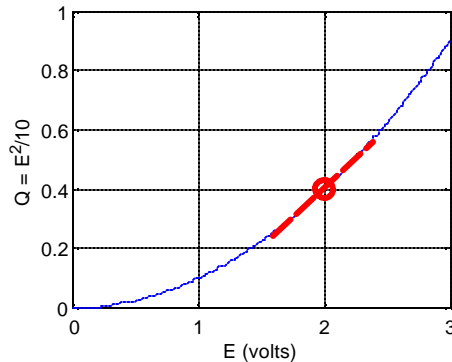


To find J, B and K: (1) Apply a known torque to the boat and measure the steady-state offset angle. K is the ratio “torque/angle” (Nm/rad).  $K = \frac{T_{in}}{q_{ss}}$  Then, we can find J and B

by giving the boat a “tap” (i.e. at impulse), or by suddenly releasing our constant input torque (i.e. a step). From the damped natural frequency and decay envelope, we should be able to get estimates for J and B. Practically speaking, the undamped natural frequency is pretty darn close to the damped natural frequency if we can actually see at least of couple of oscillations (zeta less than .2 or so), so  $w_d \approx \sqrt{K/J}$  or  $J = K/w_n^2 \approx K/w_d^2$ . If the decay envelope has a time constant “t” (in seconds), then define  $s = 1/t$ , and note that  $s = \frac{1}{2} \frac{B}{J}$  (this is the negative of the real part of the pole-pair) or  $B = 2sJ = \frac{2J}{t}$ .

#### Problem 4 – Transfer function for an over controller

We know electrical power is  $Q = EI = \frac{E^2}{R}$ . To approximate the relationship between the input, E, to the desired output, T, with a “transfer function”, we first need to linearize the relationship between E (voltage) and Q (power in) about the given operating point,  $E_o$ .



The diagram above uses an arbitrary example resistance of 10 Ohms and an arbitrary choice of  $E_o=2$  volts. In the region near the operating point, a linearized relationship between E and Q is:  $Q \approx (E_o^2 / R) + m(E - E_o)$ , where the slope is found by evaluating

the derivative of  $\frac{E^2}{R}$  at the operating point,  $E_o$ .  $m = 2 \frac{E_o}{R} = 2 \frac{2}{10} = 0.4$ .

So for this particular example:  $Q_{in} \approx (E_o^2 / R) + 2(E_o^2 / R)(E - E_o) = 2(E_o^2 / R)E - E_o^2 / R$ .

Therefore:  $C \frac{dq}{dt} + Kq = Q_{in} \approx \left[ 2 \frac{E_o}{R} \right] E - \frac{E_o^2}{R}$ .

To get the transfer function from the PARTICULAR INPUT “E” to the output theta, we want a linear, constant-coefficient differential equation with no additional “constants”,

“offsets”, “inputs” or whatever other names you like. Then:  $C \frac{dq}{dt} + Kq = \left[ 2 \frac{E_o}{R} \right] E$ , and

finally:  $\frac{\Theta(s)}{E(s)} = \frac{2E_o / R}{Cs + K}$

## Problem 5 – effect of a zero on the time response

Section 4.8 in Nise (“Nise” rhymes with “ice”, btw, according to the publisher) discusses just this example! What we are looking for here is a range of values for which the zero has a significant impact on the total response. Doing a partial fraction expansion:

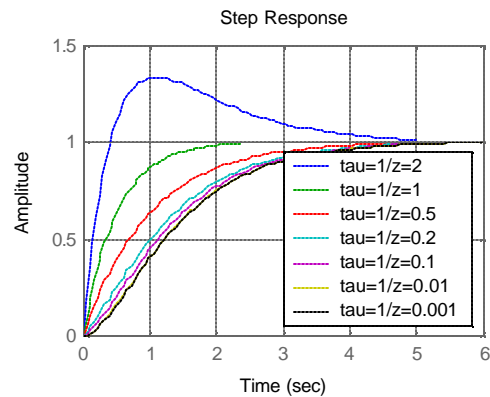
$$\frac{K(s+z)}{(s+1)(s+2)} = K \left( \frac{s}{s+1} + \frac{-s}{s+2} + \frac{-z}{s+1} + \frac{0.5z}{s+2} + \frac{0.5z}{s} \right)$$

Here, the two lefthand terms with “s” in the numerator represent the “effect of the zero” on the step response. The three remaining terms are all weighted by “z”, while those two lefthand terms are not. For the three righthand terms to “dominate” over any effects from the zero, a general rule is to have the zero roughly 10x further away from the origin than the pole(s).

```
figure(1); clf
tau_list=[2 1 .5 .2 .1 .01 .001];
legval='legend(';
for n=1:length(tau_list)
    step(tf(2*[tau_list(n) 1],[1 3 2])); hold on
    legval=[legval ''tau=1/z=' num2str(tau_list(n)) ''',''];
end
legval=[legval '4)'];
eval(legval)
grid on
```

This MATLAB code generates the plot at right. The step responses have all been normalized here to have the same steady-state value, 1.

by  $z=10$  or so, the effect of the zero is no longer significant.



## Problem 6 – Nise Problem 2.52

- $\ddot{x} + 15\dot{x} + 50x = -4$
- $\ddot{x} + 15\dot{x} + 50x = 2x$  or  $\ddot{x} + 15\dot{x} + 48x = 0$
- $\ddot{x} + 15\dot{x} + 50x = 4$