ENG204 - Signals and Linear Systems – Assignment 1.2

Baley Eccles - 652137

[2024-08-26 Mon 12:08]

Contents

1 ENG204 - Signals and Linear Systems – Assignment 1.2

1.1 b

```
clc
clear
pkg load symbolic
syms mt at ms as  ...
     Fkt Fks Fcs Fks
                                % sum F_y = ma
eq1 = mt*at == Fkt - Fks - Fcs;
eq2 = ms*as == Fks + Fcs;

                                % Sub in force equations
syms kt u zt zs dzs dzt ks cs
FktEqu = -kt*(u-zt);
FksEqu = -ks*(zt-zs);
FcsEqu = -cs*(dzt-dzs);
eq1=subs(subs(subs(eq1,Fkt,FktEqu),Fks,FksEqu),Fcs,FcsEqu);
eq2=subs(subs(subs(eq2,Fkt,FktEqu),Fks,FksEqu),Fcs,FcsEqu);
```

1.1.1 Discrete

```
                                % Sub in forward difference
syms zs zs1 zs2 zt zt1 zt2 Ts vs1 vt1 vs vt
% First forward difference of displacement
dzsEqu = (zs1-zs)/(Ts);
dztEqu = (zt1-zt)/(Ts);
% First forward differences of acceleration
dvsEqu = (vs1-vs)/(Ts);
dvtEqu = (vt1-vt)/(Ts);
eq1=subs(eq1,at,dvtEqu);
```

```
eq1=subs(eq1,dzs,dzsEqu);
eq1=simplify(subs(eq1,dzt,dztEqu))
eq2=subs(eq2,as,dvsEqu);
eq2=subs(eq2,dzs,dzsEqu);
eq2=simplify(subs(eq2,dzt,dztEqu))
```

1. Difference Equations

   - $\frac{m_t(-v_s[n+1]+v_s[n+1])}{T_s} = \frac{T_s(k_s(z_s[n]-z_t[n])+kt(u[n]-z_t[n]))-c_s(z_s[n]-z_s[n+1]-z_t[n]+z_t[n+1])}{T_s}$
   - $\frac{m_s(-v_t[n]+v_t[n+1[)}{T_s} = \frac{-T_sk_s(z_s[n]-z_t[n])+c_s(z_s-z_s[n+1]-z_t[n]+z_t[n+1])}{T_s}$

1.1.2 Continious

```
fprintf("at = \n")
latex(expand(solve([eq1,eq2],at)))
fprintf("as = \n")
latex(expand(solve([eq1,eq2],as)))
```

1. ODE

   - $a_t = -\frac{c_s\dot{z}_s}{m_t} + \frac{c_s\dot{z}_t}{m_t} - \frac{k_sz_s}{m_t} + \frac{k_sz_t}{m_t} - \frac{k_tu}{m_t} + \frac{k_tz_t}{m_t}$
   - $a_s = \frac{c_s\dot{z}_s}{m_s} - \frac{c_s\dot{z}_t}{m_s} + \frac{k_sz_s}{m_s} - \frac{k_sz_t}{m_s}$
   - $v_t = \dot{z}_t$
   - $v_s = \dot{z}_s$

1.2 c

Using:

- $\underline{q}[n+1] = \underline{A}q[n] + \underline{b}x[n]$
- $y[n] = \underline{C}q[n] + dx[n]$

We chose the following state varibles:

- $q_1[n] = z_t[n]$
- $q_2[n] = v_t[n]$
- $q_3[n] = z_s[n]$
- $q_4[n] = v_s[n]$
- $x[n] = u[n]$
- $y[n] = \begin{bmatrix} z_t[n] \\ z_s[n] \end{bmatrix}$

1.2.1 Continious

1. State Equation

```
syms q1n q1n1 q2n q2n1 q3n q3n1 q4n q4n1

eq1 = subs(eq1,zt,q1n);
eq1 = subs(eq1,zs,q3n);


eq1 = subs(eq1,dzt,q1n1);
eq1 = subs(eq1,at,q2n1);
eq1 = subs(eq1,dzs,q3n1);
eq1 = subs(eq1,as,q4n1);

eq2 = subs(eq2,zt,q1n);
eq2 = subs(eq2,zs,q3n);

eq2 = subs(eq2,dzt,q1n1);
eq2 = subs(eq2,at,q2n1);
eq2 = subs(eq2,dzs,q3n1);
eq2 = subs(eq2,as,q4n1);

eq1 = subs(eq1, q1n1, eqq1n1);
eq1 = subs(eq1, q3n1, eqq3n1);
eq2 = subs(eq2, q1n1, eqq1n1);
eq2 = subs(eq2, q3n1, eqq3n1);
expand(simplify(solve([eq1,eq2],q2n1)));
expand(simplify(solve([eq1,eq2],q4n1)));
```

- $q_1[n+1] = q_2[n]$
- $q_2[n+1] = \frac{c_s q_2[n]}{m_t} - \frac{c_s q_4[n]}{m_t} + \frac{k_s q_1[n]}{m_t} - \frac{k_s q_3[n]}{m_t} + \frac{k_t q_1[n]}{m_t} - \frac{k_t u[n]}{m_t}$
- $q_3[n+3] = q_4[n]$
- $q_4[n+1] = -\frac{c_s q_2[n]}{m_s} + \frac{c_s q_4[n]}{m_s} - \frac{k_s q_1[n]}{m_s} + \frac{k_s q_3[n]}{m_s}$

$$\underline{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{k_s}{m_t} + \frac{k_t}{m_t} & \frac{c_s}{m_t} & -\frac{k_s}{m_t} & -\frac{c_s}{m_t} \\ 0 & 0 & 0 & 1 \\ -\frac{k_s}{m_s} & -\frac{c_s}{m_s} & \frac{k_s}{m_s} & \frac{c_s}{m_s} \end{bmatrix}$$

$$\underline{B} = \begin{bmatrix} 0 \\ -\frac{k_t}{m_t} \\ 0 \\ 0 \end{bmatrix}$$

2. Output Equation Using

- $z_t = q_1$
- $z_s = q_3$

$$\underline{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$d = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

1.2.2 Discrete

1. State Equation Need to solve for the matricies in $\underline{q}[n+1] = \underline{A}q[n] + \underline{b}x[n]$

```
syms q1n q1n1 q2n q2n1 q3n q3n1 q4n q4n1

eq1 = subs(eq1,zt,q1n);
eq1 = subs(eq1,vt,q2n);
eq1 = subs(eq1,zs,q3n);
eq1 = subs(eq1,vs,q4n);
eq1 = subs(eq1,zt1,q1n1);
eq1 = subs(eq1,vt1,q2n1);
eq1 = subs(eq1,zs1,q3n1);
eq1 = subs(eq1,vs1,q4n1);

eq2 = subs(eq2,zt,q1n);
eq2 = subs(eq2,vt,q2n);
eq2 = subs(eq2,zs,q3n);
eq2 = subs(eq2,vs,q4n);
eq2 = subs(eq2,zt1,q1n1);
eq2 = subs(eq2,vt1,q2n1);
eq2 = subs(eq2,zs1,q3n1);
eq2 = subs(eq2,vs1,q4n1);

equq1n1 = q1n+Ts*q2n; % t
equq3n1 = q3n+Ts*q4n; % s

eq1 = subs(eq1, q1n1, equq1n1);
eq1 = subs(eq1, q3n1, equq3n1);
eq2 = subs(eq2, q1n1, equq1n1);
eq2 = subs(eq2, q3n1, equq3n1);

latex(expand(simplify(solve(eq1,q2n1))))
latex(expand(simplify(solve(eq2,q4n1))))
```

Which gives us the following equations:

- $q_1[n+1] = q_1[n] + T_s \cdot q_2[n]$
- $q_4[n+1] = -\frac{T_s c_s q_2[n]}{m_t} + \frac{T_s c_s q_4[n]}{m_t} - \frac{T_s k_s q_1[n]}{m_t} + \frac{T_s k_s q_3[n]}{m_t} - \frac{T_s k_t q_1[n]}{m_t} + \frac{T_s k_t u[n]}{m_t} + q_4[n]$
- $q_3[n+1] = q_3[n] + T_s \cdot q_4[n]$
- $q_2[n+1] = \frac{T_s c_s q_2[n]}{m_s} - \frac{T_s c_s q_4[n]}{m_s} + \frac{T_s k_s q_1[n]}{m_s} - \frac{T_s k_s q_3[n]}{m_s} + q_2[n]$

Therefore:

$$\underline{A} = \begin{bmatrix} 1 & T_s & 0 & 0 \\ \frac{T_s k_s q_1[n]}{m_t} + \frac{T_s k t q_1[n]}{m_t} & \frac{T_s c_s q_2[n]}{m_t} + 1 & -\frac{T_s k_s q_3[n]}{m_t} & -\frac{T_s c_s q_4[n]}{m_t} \\ 0 & 0 & 1 & T_s \\ -\frac{T_s k_s q_1[n]}{m_s} & -\frac{T_s c_s q_2[n]}{m_s} & \frac{T_s k_s q_3[n]}{m_s} & \frac{T_s c_s q_4[n]}{m_s} + 1 \end{bmatrix}$$

$$\underline{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{T_s k_t}{m_t} \end{bmatrix}$$

2. Output Equation Need to solve for the matricies in $y[n] = \underline{C}q[n] + dx[n]$

   Using:

   - $q_1[n] = z_t[n]$
   - $q_3[n] = z_s[n]$

   Therefore:

$$\underline{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$d = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

1.3 d

This code defines the array of systems that will be used

```
clear
clc
pkg load symbolic
pkg load control
% Using student IDs 652137 and 651790
%unsprungMass=236;
%sprungMass=2296;

ms=2296;
mt=236;
kt=250000;
ksMin = 10000;
ksMax = 250000;
csMin = 500;
csMax = 2000;
Ts = 0.1;
t = 0:Ts:1;
idx=1;
% Get an array of systems based on the possible values for ks and cs
numOfSys = 1;
for i =0:numOfSys;
    for j =0:numOfSys;
        % Get the ks and cs for the current system
        ks = ksMin + (i/numOfSys)*(ksMax-ksMin);
        cs = csMin + (i/numOfSys)*(csMax-csMin);

A=[0,1,0,0; ...
    (ks)/(mt)+(kt)/(mt),(cs)/(mt),-(ks)/(mt),-(cs)/(mt);...
    0,0,0,1;...
    -(ks)/(ms),-(cs)/(ms), (ks)/(ms),(cs)/(ms)];
    B=[0 ; ...
       (kt)/(mt) ; ...
       0 ; ...
       0];
        C=[1 , 0 , 0 , 0 ; ...
```

```
        0 , 0 , 1 , 0];
    D=[0 ; ...
        0];

    sysArray(idx).A = A;
    sysArray(idx).B = B;
    sysArray(idx).C = C;
    sysArray(idx).D = D;
    idx = idx +1;
  end
end
```

Check if the system is stable using the eigenvalues

```
for i = 1:length(sysArray)
  eigen=eig(sysArray(i).A);
  if (all(abs(eigen)) < 1)
    fprintf("The %i th system is stable\n", i)
  else
    fprintf("The %i th system is unstable\n", i)
  end
end
```

It appears that all of the systems are unstable. So we will expect the impulse response to get really large as t does.

The impuse response:

```
sys = cell(length(sysArray), 1);
for i = 1:length(sysArray)
    sys{i} = ss(sysArray(i).A, sysArray(i).B, sysArray(i).C, sysArray(i).D);
end

impulseResponses = cell(length(sysArray), 1);

for i = 1:length(sysArray)
    [y, t] = impulse(sys{i});
    impulseResponses{i} = [y,t];
end


figure;
hold on;
for i = 1:length(sysArray)
    plot(impulseResponses{i}(:,3), impulseResponses{i}(:,1), 'DisplayName',
    ↪ sprintf('System %d', i));
    plot(impulseResponses{i}(:,3), impulseResponses{i}(:,2), 'DisplayName',
    ↪ sprintf('System %d', i));
end
hold off;
xlabel('Time (s)');
ylabel('Impulse Response');
title('Impulse Responses of All Systems');
legend show;
grid on;
```

1.4 e

```
t = 0:Ts:25;
f = 10;
w0 = 2*pi*f;
um = 1;
u = um*sin(w0*t);
[y, t] = lsim(sys, u, t);

figure;
plot(t, y(:, 1), 'b')
hold on;
plot(t, y(:, 2), 'r')
title('Impulse response of the car suspension');
xlabel('Time (s)');
ylabel('Displacement (m)');
legend('Body Displacement', 'Wheel Displacement');
grid on;
```

```
warning: lsim: arguments number 1 are invalid and are being ignored
error: lsim: require at least one LTI model
error: called from
    lsim at line 94 column 5
warning: opengl_renderer: data values greater than float capacity.  (1) Scale data, or (2) Use g
warning: called from
    uimenu at line 97 column 8
    __add_default_menu__ at line 68 column 5
    figure at line 97 column 5
error: __plt2vv__: vector lengths must match
error: called from
    __plt__>__plt2vv__ at line 489 column 5
    __plt__>__plt2__ at line 248 column 14
    __plt__ at line 115 column 16
    plot at line 240 column 10
error: __plt2vv__: vector lengths must match
error: called from
    __plt__>__plt2vv__ at line 489 column 5
    __plt__>__plt2__ at line 248 column 14
    __plt__ at line 115 column 16
    plot at line 240 column 10
error: legend: no valid object to label
error: called from
    legend>parse_opts at line 776 column 7
    legend at line 216 column 8
```