Modified from Prelab 6 Simulation Code

Table of Contents

Caitlyn Caggia - ECE 4550 HW2.2d

```
clc, clear all, close all
% not included: 1) anti-windup compensation, 2) reference command
 shaping
```

plant simulation model parameters

```
L = 0.014;                          % motor inductance (fixed)
R = 7.6;                            % motor resistance (fixed)
K = 0.16;                           % motor magnetic coefficient
 (fixed)
J = 28e-6;                          % motor/load inertia (load
 dependent)
F = 1.9e-4;                         % motor/load friction (load
 dependent)
```

plant design model parameters

```
Jhat = J*1;                         % assumed vs true parameter
 mismatch
Fhat = F*1;                         % assumed vs true parameter
 mismatch
alpha = (K^2+Fhat*R)/(Jhat*R);      % design model A matrix
 coefficient
beta = K/(Jhat*R);                  % design model B matrix
 coefficient
```

controller design parameters

```
Vdc = 24;                           % power supply voltage
T = 0.001;                          % controller period
lambda_r = 5;                       % regulator bandwidth parameter
lambda_e = 4*lambda_r;              % estimator bandwidth parameter
```

simulation model coefficient matrices

```
As = [0,1,0;0,-F/J,K/J;0,-K/L,-R/L];
Bs = [0;0;1/L];
Cs = [1,0,0];
```

design model coefficient matrices

```
Ad = [0,1;0,-alpha];
Bd = [0;beta];
Cd = [1,0];
```

controller feedback gain matrices

```
LL = [2*lambda_e-alpha;lambda_e^2-2*alpha*lambda_e+alpha^2];
%LL = [23320 80; 157602 2398];
%KK1 = [3*lambda_r^2,3*lambda_r-alpha]/beta;
KK1 = 148;  %part b
%KK2 = lambda_r^3/beta;
KK2 = 502;
```

reference input

```
r = 1;
```

data converters

```
Qy = 2*pi/1000;     % sensor quantization
Qu = 48/1500;       % actuator quantization
```

time grid

```
h = T/100;
t = 0:h:1;
```

preallocation

```
U_a = zeros(1,length(t));
X = zeros(3,length(t));
Y = zeros(3,length(t));
```

initial conditions

```
x = [0;0;0];                    % plant states
xhat = [0;0];                   % estimator states
sigma = 0;                      % regulator state
u = 0;                          % first actuator input
y = 0;                          % first sensor output
```

simulation time loop

```
for n = 0:length(t)-1

    % microcontroller code (discrete-time update)
    if mod(n*h,T) == 0

        % controller output to actuator
        u_a = Qu*round(u/Qu);
        % controller input from sensor
        y_s = Qy*round(y/Qy);
        % compute controller output
        u = -KK1*xhat-KK2*sigma;
        if abs(u) > 0.95*Vdc, u = 0.95*Vdc*sign(u); end
        % update controller state variables
        xhat = xhat+T*(Ad*xhat+Bd*u-LL*(Cd*xhat-y_s));
        sigma = sigma+T*(y_s-r);
    end

    % store results (only for analysis)
    U_a(:,n+1) = u_a; X(:,n+1) = x; Y(:,n+1) = y;

    % plant physics (continuous-time update)
    x = x+h*(As*x+Bs*u_a);
    y = Cs*x;

end
```

```
Error using *
Inner matrix dimensions must agree.

Error in HW6Prob2 (line 81)
        xhat = xhat+T*(Ad*xhat+Bd*u-LL*(Cd*xhat-y_s));
```

closed-loop response plots

```
subplot(411), plot(t,Y(1,:)), xlabel('t [s]'), ylabel('x')
subplot(412), plot(t,Y(2,:)), xlabel('t [s]'), ylabel('sigma')
subplot(413), plot(t,Y(3,:)), xlabel('t [s]'), ylabel('epsilon')
subplot(414), plot(t,U_a), xlabel('t [s]'), ylabel('u(t)')
```

Published with MATLAB® R2017a