MAE5803 - HW#5 Part 3b Adaptive Controller

```matlab
function HW5P3b()

% Given
qFlag = [1 2];
pFlag = [1 2 3];
[A,B] = meshgrid(qFlag,pFlag);
c = cat(2,A',B');
caseFlag = reshape(c,[],2);

% Given
m1 = 1; l1 = 1; me = 2; de = pi/6; I1 = 0.12; lc1 = .5; Ie = .25; lce
 = .6;
a(1) = I1 + m1*lc1^2 + Ie + me*lce^2 + me*l1^2;
a(2) = Ie + me*lce^2;
a(3) = me*l1*lce*cos(de);
a(4) = me*l1*lce*sin(de);
Lambda = 20*eye(2);
Kd = 100*eye(2);
P{1} = diag([0.6,0.1,0.1,0.06]);
P{2} = 200*P{1};
P{3} = 0.1*P{1};

% Initial Conditions
tspan = [0 3];
X0 = [0,0,0,0,0,0,0,0];

for i1 = 1:length(caseFlag)
    [T{i1},X{i1}] = ode45(@EOM,tspan,X0,
[],a,Lambda,Kd,P,caseFlag(i1,:));

for i2 = 1:length(T{i1})
    [~,tau{i1}(:,i2),q_err{i1}(:,i2)] = EOM(T{i1}(i2),X{i1}
(i2,:)',a,Lambda,Kd,P,caseFlag(i1,:));
end

fh = figure(i1);
set(fh,'Position',[0 10 799 1079])
suptitle(['HW5 Problem #3b  qd' num2str(caseFlag(i1,1)) '  P'
 num2str(caseFlag(i1,2))])
subplot(221)
plot(T{i1},rad2deg(q_err{i1}(1,:)))
ylabel('Position Error 1 (deg)'); xlabel('Time (s)')
subplot(222)
plot(T{i1},rad2deg(q_err{i1}(1,:)))
ylabel('Position Error 2 (deg)'); xlabel('Time (s)')
subplot(223)
plot(T{i1},tau{i1}(1,:))
ylabel('Control Torque 1'); xlabel('Time (s)')
subplot(224)
plot(T{i1},tau{i1}(1,:))
```

```matlab
    ylabel('Control Torque 1'); xlabel('Time (s)')
    figureName = ['HW5P3b_qd' num2str(caseFlag(i1,1)) '_P'
     num2str(caseFlag(i1,2))];
    print(['HW5P3b_pdfs\' figureName],'-dpdf','-fillpage')

    fh2 = figure(i1+length(caseFlag));
    set(fh2,'Position',[800 10 799 1079])
    plot(T{i1},X{i1}(:,5:8));
    legend('a1','a2','a3','a4'); xlabel('Time');
    title(['aPlot HW5 Problem #3b  qd' num2str(caseFlag(i1,1)) '  P'
     num2str(caseFlag(i1,2))])
    figureName2 = ['aPlot_HW5P3b_qd' num2str(caseFlag(i1,1)) '_P'
     num2str(caseFlag(i1,2))];
    print(['HW5P3b_pdfs\' figureName2],'-dpdf','-fillpage')
end
end

function [dx,tau,q_err] = EOM(t,x,a,Lambda,Kd,Ps,caseFlag)
dx = zeros(size(x));
q = [x(1); x(2)];
q_d = [x(3); x(4)];
a_hat = x(5:8);

qFlag = caseFlag(1);
pFlag = caseFlag(2);
switch qFlag
    case 1
        qd = [1-exp(-t); 2*(1-exp(-t))];
        qd_d = [exp(-t); 2*exp(-t)];
        qd_dd = [-exp(-t); -2*exp(-t)];
    case 2
        qd = [1-cos(2*pi*t); 2*(1-cos(2*pi*t))];
        qd_d = [2*pi*sin(2*pi*t); 4*pi*sin(2*pi*t)];
        qd_dd = [4*pi^2*cos(2*pi*t); 8*pi^2*cos(2*pi*t)];
end
P = Ps{pFlag};
% I would like to use one of the triplets below so I don't have to use
% qFlag and I can just pass in an anonymous function for qd, but it
 takes
% too long to evaluate an anonymous function each time...
% qd    = eval(qd_fun(t));
% qd_d  = eval(subs(diff(qd_fun,1),t));
% qd_dd = eval(subs(diff(qd_fun,2),t));
% qd    = eval(qd(t));
% qd_d  = eval(qd_d(t));
% qd_dd = eval(qd_dd(t));
q_err = q - qd;
q_err_d = q_d - qd_d;

qr_d = qd_d - Lambda*q_err;
qr_dd = qd_dd - Lambda*q_err_d;

H(1,1) = a(1) + 2*a(3)*cos(q(2)) + 2 * a(4)*sin(q(2));
H(1,2) = a(2) + a(3)*cos(q(2)) + a(4)*sin(q(2));
```

```
H(2,1) = H(1,2);
H(2,2) = a(2);
h = a(3)*sin(q(2)) - a(4)*cos(q(2));

C(1,1) = -h*q_d(2);
C(1,2) = -h*(q_d(1)+q_d(2));
C(2,1) = h*q_d(1);
C(2,2) = 0;

Y(1,1) = qr_dd(1);
Y(1,2) = qr_dd(2);
Y(2,1) = 0;
Y(2,2) = qr_dd(1) + qr_dd(2);
Y(1,3) = (2*qr_dd(1) + qr_dd(2))*cos(q(2)) - (q_d(2)*qr_d(1) +
 q_d(1)*qr_d(2) + q_d(2)*qr_d(2))*cos(q(2));
Y(1,4) = (2*qr_dd(1) + qr_dd(2))*sin(q(2)) + (q_d(2)*qr_d(1) +
 q_d(1)*qr_d(2) + q_d(2)*qr_d(2))*sin(q(2));
Y(2,3) = qr_dd(1)*cos(q(2)) + q_d(1)*qr_d(1)*sin(q(2));
Y(2,4) = qr_dd(1)*sin(q(2)) - q_d(1)*qr_d(1)*cos(q(2));

s = q_err_d + Lambda*q_err;
tau = Y*a_hat - Kd*s;

dx(1:4) = [zeros(2) zeros(2); zeros(2) inv(H)]*[0;0;tau] ...
           + [zeros(2) eye(2); zeros(2) -inv(H)*C]*x(1:4);
dx(5:8) = -P*transpose(Y)*s;
end
```

Published with MATLAB® R2016a