Intelligent Systems Task 2: Adaptive Control - Chapter 1, Astrom

Nelson Campos

Contents

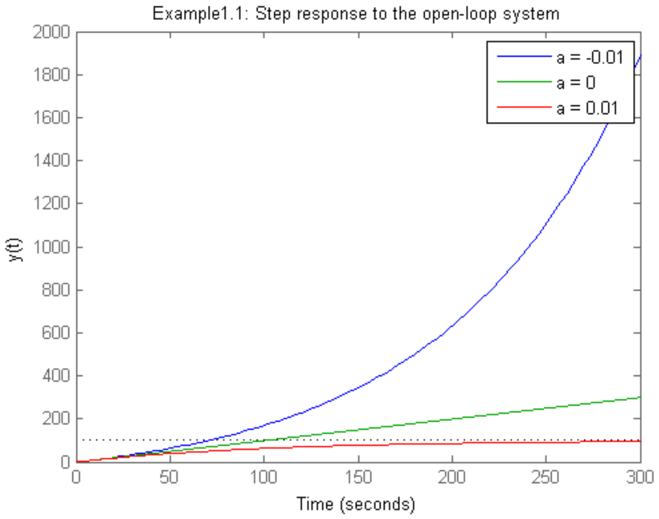

```
close all
```

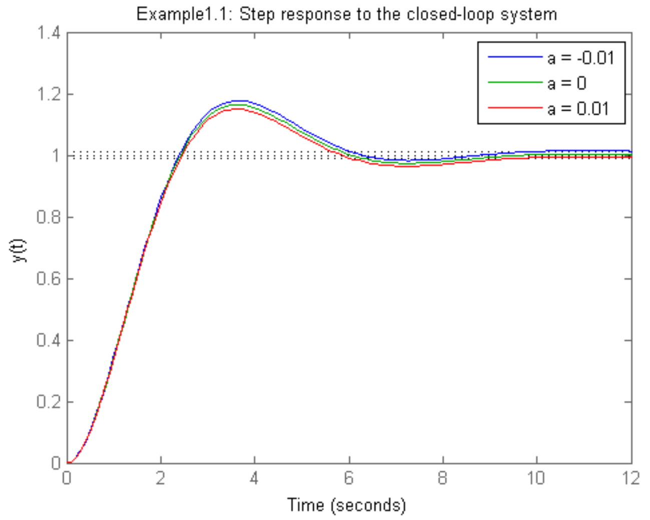
Example 1.1: Different open-loop responses

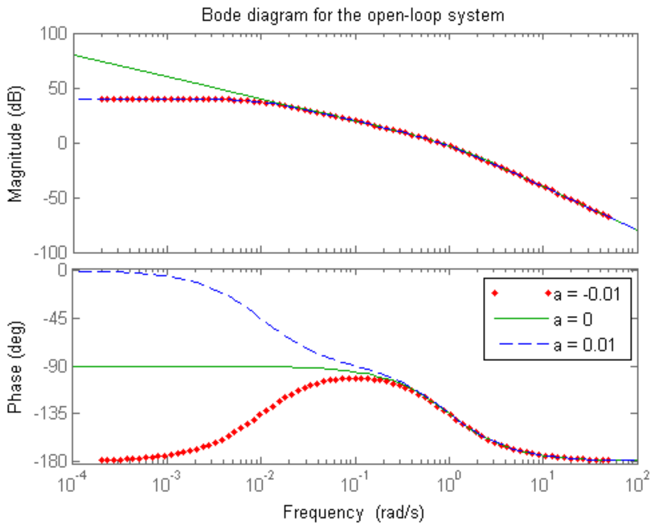
Consider the Transfer Function $Go(s) = \frac{1}{(s+1)(s+a)}$

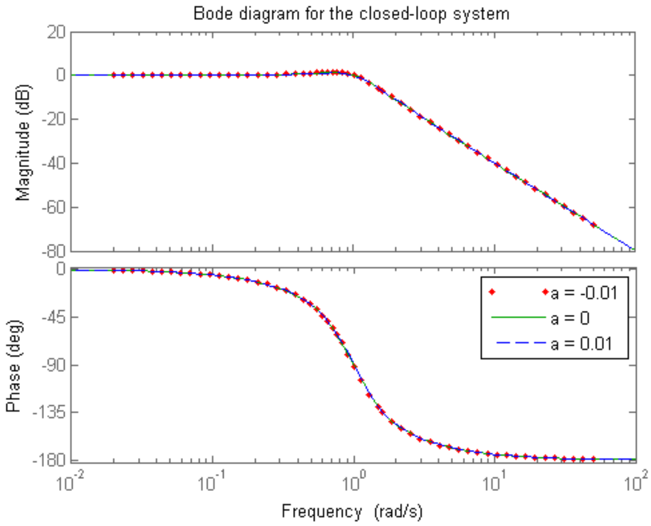
Conseider varying a like that:

```
a = [-0.01; 0; 0.01];

s = tf('s');

for i=1:3
    Go_s(i) = 1/((s+1)*(s+a(i)));
    %Step response to the open-loop system

end
figure (1), step(Go_s(1), Go_s(2), Go_s(3), 300),ylabel('y(t)'), xlabel('Time'),
title('Example1.1: Step response to the open-loop system'), legend('a = -0.01', 'a = 0', 'a = 0.01')

%Step response to the closed-loop system
figure (2), step(feedback(Go_s(1),1),feedback(Go_s(2),1),feedback(Go_s(3),1)),ylabel('y(t)'), xlabel('Time'),
title('Example1.1: Step response to the closed-loop system'), legend('a = -0.01', 'a = 0', 'a = 0.01')
%Bode diagram for the open-loop system
figure (3), bode(Go_s(1), '.r',Go_s(2), '-g', Go_s(3), '--b'), legend('a = -0.01','a = 0','a = 0.01'), title('Bode diagram for the open-loop system')

%Bode diagram for the closed-loop system
figure (4), bode(feedback(Go_s(1),1), '.r',feedback(Go_s(2),1), '-g', feedback(Go_s(3),1), '--b'), legend('a = -0.01','a = 0','a = 0.01'), title('Bode diagram
```


Example1.1: Step response to the open-loop system
a = -0.01
a = 0
a = 0.01
y(t)
Time (seconds)


Example1.1: Step response to the closed-loop system
a = -0.01
a = 0
a = 0.01
y(t)
Time (seconds)

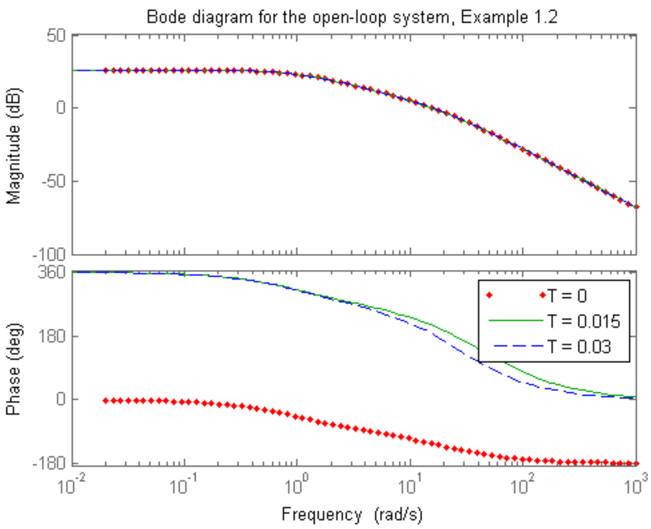

Bode diagram for the open-loop system
Magnitude (dB)
Phase (deg)
a = -0.01
a = 0
a = 0.01
Frequency (rad/s)

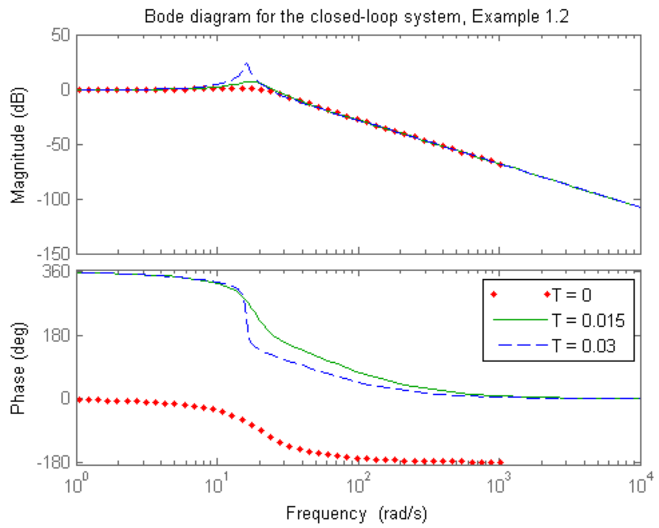


Example 1.2: Similar open-loop responses

```
%Consider the Transfer Function $$ Go(s) =
%400\frac{(1-sT)}{(s+1)(s+20)(1+T*s)} $$
```

```
%Consider varying T like that:
T = [0; 0.015; 0.030];

s = tf('s');

for i=1:3
    Go1_s(i) = 400*(1-s*T(i))/((s+1)*(s+20)*(1+T(i)*s));
end

%Bode diagram for the open-loop system
figure (5), bode(Go1_s(1), '.r',Go1_s(2), '-g', Go1_s(3), '--b'), legend('T = 0','T = 0.015','T = 0.03'), title('Bode diagram for the open-loop system, Example

%Bode diagram for the closed-loop system
figure (6), bode(feedback(Go1_s(1),1), '.r',feedback(Go1_s(2),1), '-g', feedback(Go1_s(3),1), '--b'), legend('T = 0','T = 0.015','T = 0.03'), title('Bode diagr
```

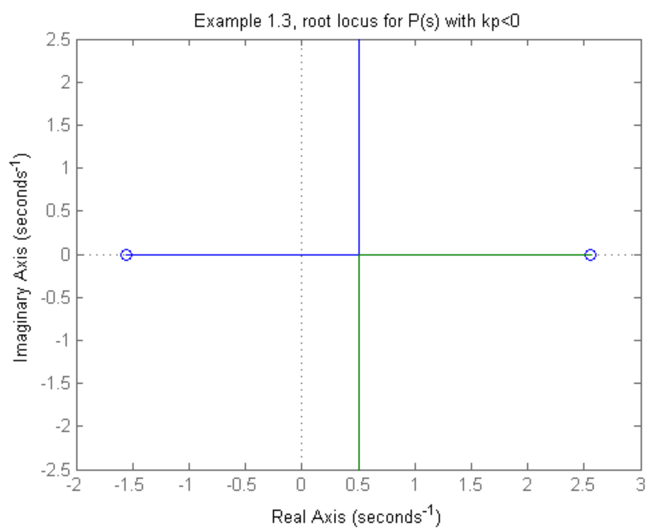


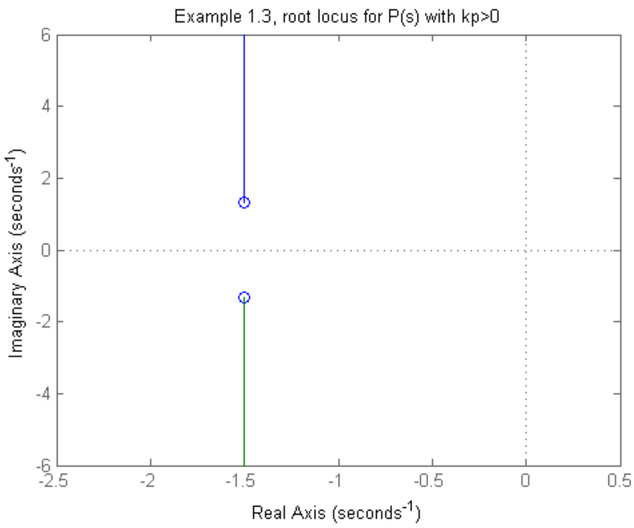


Example 1.3: Integrator with unknown sign

Consider a process whose dynamics obeys the following equation: $G_0(s) - \frac{k_p}{s}$ If kp would be either positive or negative, it's phase mary vary 180º Assume that the controller transfer function is a monic polynomial and it's characteristic equation will be $P(s) = sR(s) + k_p S(s)$ where the ractional function $\frac{R(s)}{S(s)}$ is the transfer funtion of the controller. When all coefficients of P(s) are positive, P(s) will have a zero on the left side of the plane s. Otherwise, the zero will move to the right side plane.

```
kp = [-2 2];

for i=1:2
  Go_2(i) = kp(i)/s;
  %R(s)>=S(s)
  R(i) = s+1;
  S(i) = s+2;
  P(i) = s*R(i)+kp(i)*S(i);
end
figure(7), rlocus(P(1)), title('Example 1.3, root locus for P(s) with kp<0')
figure(8), rlocus(P(2)), title('Example 1.3, root locus for P(s) with kp>0')
```

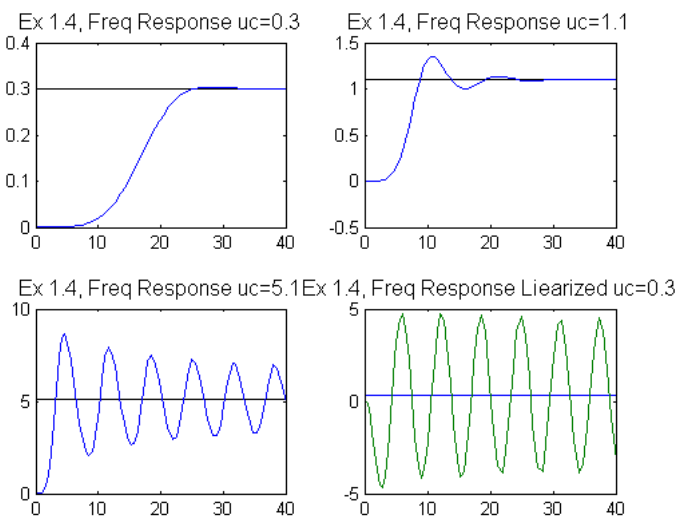


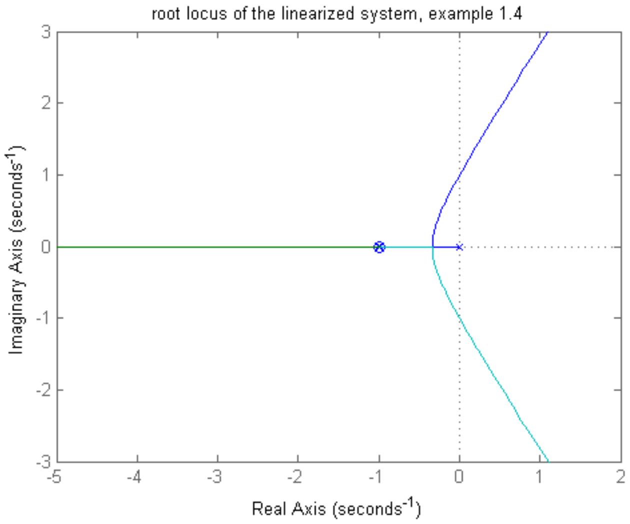


Example 1.4 and Question 1.5: Nonlinear valve

Consider a valve which dynamics is described by $v = f(u) = u^4$ The output of the controller PI is inserted into the valve. Once it's nonlinear, at some point uo, a little variation on the valve is described by $\Delta(v) = u_o^3\Delta(u)$

```
ex1_4_q5
t=output1(:,1);

ref1=output1(:,2);
ref2=output2(:,2);
ref3=output3(:,2);

s1=output1(:,3);
s2=output2(:,3);
s3=output3(:,3);
out_lin = output_lin(:,3);

figure(9)
subplot(2,2,1), plot(t,ref1,'k', t,s1,'b');
title('Ex 1.4, Freq Response uc=0.3', 'fontsize', 12);
subplot(2,2,2), plot(t,ref2,'k', t,s2,'b');
title('Ex 1.4, Freq Response uc=1.1', 'fontsize', 12);
subplot(2,2,3), plot(t,ref3,'k', t,s3,'b');
title('Ex 1.4, Freq Response uc=5.1', 'fontsize', 12);
subplot(2,2,4), plot(t,ref1, t,out_lin);
title('Ex 1.4, Freq Response Liearized uc=0.3', 'fontsize', 12);

uo = (10/3)^(1/3); %

s = tf('s');
%roots of the open-loop transfer function
figure(10), rlocus([1 1],[1 3 3 1 0]), title('root locus of the linearized system, example 1.4');
```



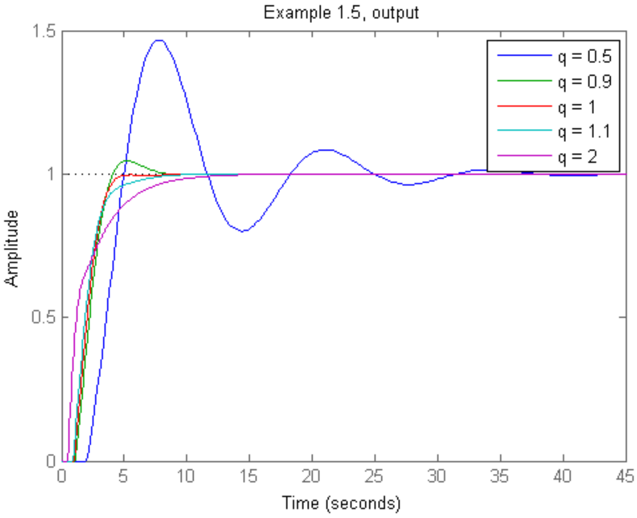


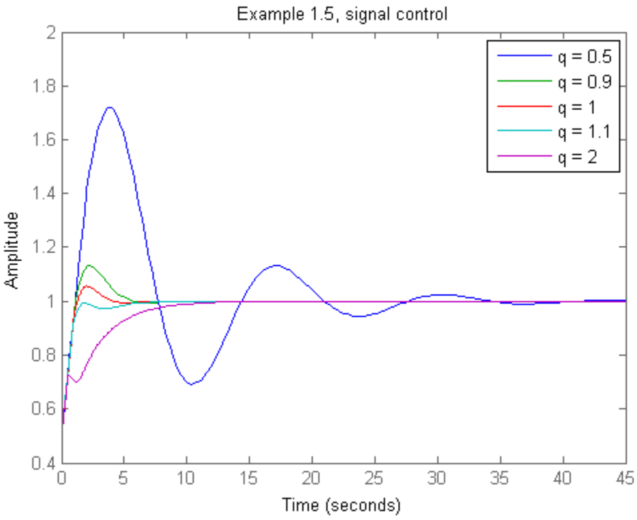
Example 1.5 and question 1.6: Concentration control

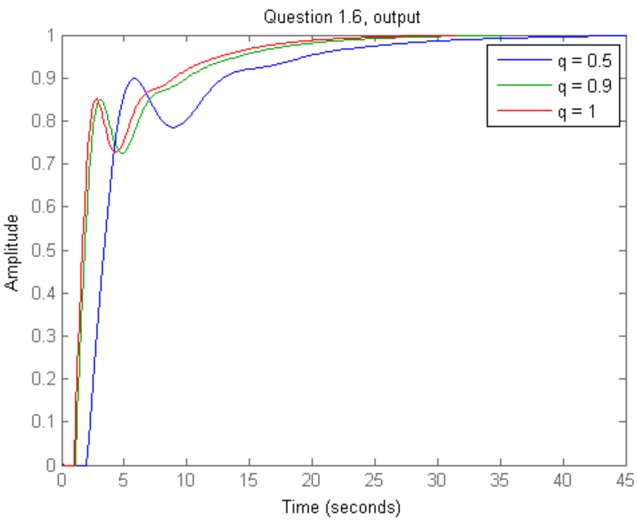
The concentration control of a fluid for a fluid that flows through a pipe, with no mixing, and through a tank, with perfect mixing. A mass balance gives: $V_m \frac{dc(t)}{dt} = q(t)(c_{in}(t-\tau) - c(t))$ For a fixed flow the transfer function of the process will be: $G_o(s) = \frac{e^{-sT}}{a + sT}$ The output and control signals of the process are depicted in example 1.5 and the controller was designed with Ziegler-Nichols method in question 1.6

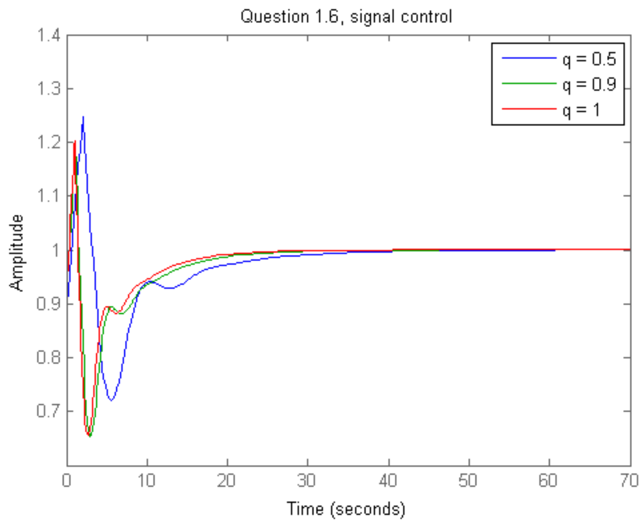
```
q = [0.5;0.9;1;1.1;2]; %flow
K = 0.5;
Ti = 1.1;
PI_s = K*(1 + 1/(1*Ti*s));

for i=1:length(q)
  T = 1/q(i);
  Go(i) = exp(-T*s)/(s*T+1);
  L(i) = PI_s*Go(i);
  TFcl(i) = feedback(L(i),1);
  Ccl(i) = PI_s/(1+L(i));
end

figure(11),step(TFcl(1),TFcl(2),TFcl(3),TFcl(4),TFcl(5)), title('Example 1.5, output'),legend('q = 0.5', 'q = 0.9','q = 1','q = 1.1','q = 2')
figure(12),step(Ccl(1),Ccl(2),Ccl(3),Ccl(4),Ccl(5)),title('Example 1.5, signal control'),legend('q = 0.5', 'q = 0.9','q = 1','q = 1.1','q = 2')

%question 1.6
% PI     K       Ti
%        0.4Ku   0.8Tu

Kq6 = [0.890 0.908 0.908 0.908 0.908]; %
Tiq6 = [4.98 3.45 3.09 2.83 2.80];
%figure(5), bode(Go(5))
for i=1:3
  PI_q6(i) = Kq6(i)*(1 + 1/(1*Tiq6(i)*s));
  L(i) = PI_q6(i)*Go(i);
  TFcl(i) = feedback(L(i),1);
  Ccl(i) = PI_q6(i)/(1+L(i));
end
figure(13),step(TFcl(1), TFcl(2), TFcl(3)), title('Question 1.6, output'),legend('q = 0.5', 'q = 0.9','q = 1','q = 1.1','q = 2')
figure(14),step(Ccl(1), Ccl(2), Ccl(3)), title('Question 1.6, signal control'),legend('q = 0.5', 'q = 0.9','q = 1','q = 1.1','q = 2')
```

```
Warning: Ignoring extra legend entries.
Warning: Ignoring extra legend entries.
```


Example 1.5, output
q = 0.5
q = 0.9
q = 1
q = 1.1
q = 2
Amplitude
Time (seconds)


Example 1.5, signal control
q = 0.5
q = 0.9
q = 1
q = 1.1
q = 2
Amplitude
Time (seconds)


Question 1.6, output
q = 0.5
q = 0.9
q = 1
Amplitude
Time (seconds)



Question 1.7

Consider the following system with two inputs, two outputs: $$\dot{x} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & -1 \end{bmatrix} x + \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{bmatrix} u$$

where $u_2 = -k_2 y_2$

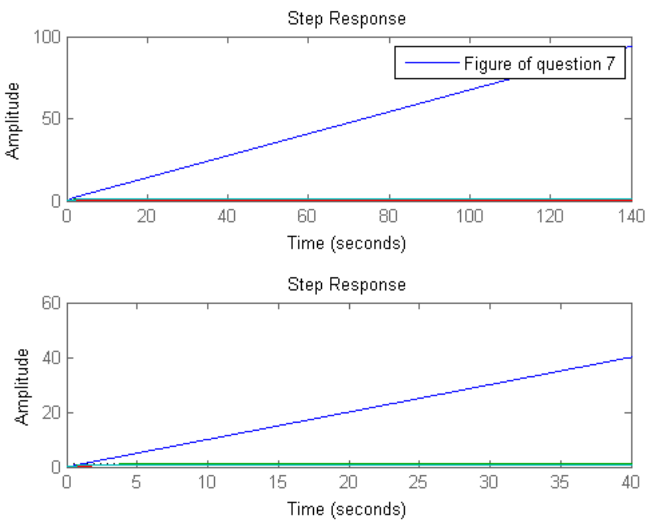
The relation Yi(s)/Ui(s) is given by

$Hi(s) = [C_i(sI - A)^{-1}]B_i, i \in \{1, 2\}$

C_i is the i-th row of C and B_i is the i-th column of B

The transfer functions will depend of k2

```
k2 = [-1 0 1 0.2];
for i=1:length(k2)
A = [-1 0 0; -2*k2(i) -3 -2*k2(i); -k2(i) 0 -1-k2(i)];
B = [1 0; 0 2; 0 1];
C = [1 1 0; 1 0 1];
H1(i) = C(1,:)*(inv((s*eye(3)-A)))*B(:,1);
H2(i) = C(2,:)*(inv((s*eye(3)-A)))*B(:,2);

end

figure(15), subplot(2,1,1), step(H1(1),H1(2),H1(3),H1(4)),legend('Figure of question 7'),subplot(2,1,2),step(H2(1),H2(2),H2(3),H2(4))
```



Example 1.8: Regulation of a quality variable in process control

Consider regulation of a quality variable of an industrial process in which there are disturbances whose characteristics are changing. In the experiment it is assumed that the process dynamics are first order with time constant T=1. It is assumed that the disturbance acts on the process input. The disturbance is simulated by sending white noise through a band-pass filter. The process dynamics are constant, but the frequency of the band-pass filter changes. Regulation can be done by a PI controller, but perfonnance can be improved significantly by using a more complex controller that is tuned to the disturbance character.

```
%Central frequency of the band-pass filter needed to generate disturbs
w = [0.1 0.05 0.05];
%Frequency to build the controller
```

```
we = [0.1 0.1 0.05];

%numerator controller parameters
b0_c = 1;
b1_c = 1;
b2_c = 1;
%denominator controller parameters
a0_c = 1;
a1_c = 0;
%denominator filter parameters
a0 = 1;
for i=1:3
  a2_c = we(i)^2;
  a1(i) = 1.4*w(i);
  a2(i) = w(i)^2;
end

%Uncomment here after ex1_8 is running
%plot(output_ex8.signals.values), xlabel('Time (sec)'), ylabel('Amplitude'),
%title('Controller output error vs Time, w = w_e = 0,1')
```

Example 1.9: Adjustment of gains in a state feedback

Consider a SISO system, described by $\dot{x} = Ax + Bu$

If the process is order n, it's dynamics is known and the controller`s laws is given by

$u = -Lx$ In this case the controller is parameterized in terms of the the matrix L

Example 1.10: A general linear controller

A general linear controller can be described by

$R(s)U(s) = -S(s)Y(s) + T(s)U_c(s)$ where R, S ant T are polynomials in s and U, Y and Uc are the Laplace transform of the control signal, output and reference value, respectively.

Example 1.11: Adjustment of a friction compensator

Friction is common in all mechanical systems. Consider a simple servo drive. Friction can to some extent be compensated for by adding the signal Ufc to a controller, where $u_{fc} = u_+$, if v>0, else $u_{fc} = -u_-$, if v <0

where v is the velocity. The signal attempts to compensate for Coulomb friction by adding a positive control signal u+ when the velocity is positive and subtracting u_ when the velocity 1S negative. The reason for having two parameters is that the friction forces are typically not symmetrical. Since there are so many factors that influence friction, it is natural to try to find a mechanism that can adjust the parameters u+ and u. automatically.

Example 1.12: An adaptive autopilot for ship steering

This is an example of a dedicated system for a special application. The adaptive autopilot is superior to a conventional autopilot for two reasons: It gives better performance, and it is easier to operate. A conventional autopilot has three dials, which have to be adjusted over a continuous scale. The adaptive autopilot has a performance-related switch with two positions (tight steering and economic propulsion). In the tight steering mode the autopilot gives good, fast response to commands with no consideration for propulsion efficiency. In the economic propulsion mode the autopilot attempts to minimize the steering loss. The control performance is significantly better than that of a well-adjusted conventional autopilot, as shown in Fig. 16

```
figure(16), imshow('fig.jpg')
```

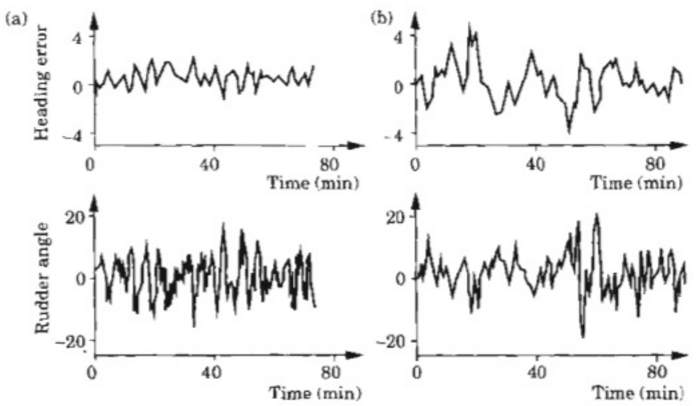


Figure 1.24 The figure shows the variations in heading and the corresponding rudder motions of a ship. (a) Adaptive autopilot. (b) Conventional autopilot based on a PID-like algorithm.

Example 1.13: Novatune

The first general-purpose adaptive system was Novatune, announced by the Swedish company Asea in 1982. The system can be regarded as a software configured toolbox for solving control problems. It broke with conventional process control by using a general-purpose discrete-time pulse transfer function as the building block. The system also has elements for conventional PI and PID control, lead-lag filter, logic, sequencing, and three modules for adaptive control. It has been used to implement control systems for a wide range of process control problems. The advantage of the system is that the control system designer has a simple means of introducing adaptation. The adaptive controller is now incorporated in ABB Master (see Chapter 12).

Example 1.6: Short-period aircraft dynamics

The following model is obtained if we assume that the aircraft is a rigid body: $\frac{dx}{dt} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & -a \end{bmatrix} x + \begin{bmatrix} b_1 \\ 0 \\ a \end{bmatrix} u$

where $x_t = (N_z, \dot{\theta}, \zeta)$ are the state variables.

This is the shor-period dynamics, where the parameters depend on the operating conditions. The table below shows the four flight conditions (FC):

	FC 1	FC 2	FC 3	FC 4
Match	0.5	0.85	0.9	1.5
Altitude(feet)	5000	5000	35000	35000
a_{ll}	-0.9896	-1.702	-0.667	-0.5162
a_{l2}	17.-41	50.72	18.11	26.96
a_{13}	96.15	263.5	84.34	178.9
a_{21}	0.2648	0.2201	0.08201	-0,6896
a_{22}	-0.8512	-1.418	-0.6587	-1.225
a_{23}	-11.39	-31.99	-10.81	-30.38
b_1	-97.78	-272.2	-85.09	-175.6
λ_1	-3.07	-4.90	-1.87	$-0.87 \pm 4.3i$
λ_2	1.23	1.78	0.56	

The system has three eigenvalues. One eigenvalue, -a = -14, which is due to the elevon servo, is constant. The other eigenvalues, λ_1 and λ_1, depend on the flight conditions. Let's assume that the output are the state variables.

```
C_root = eye(3);
for index=1:3
for FC=1:4
  switch FC
    case 1
        A = [-0.9896  17.41  96.15; 0.2648 -0.8512 -11.39; 0 0 -14];
        B = [-97.78; 0; 14];
        C = C_root(index,:);%[1 1 1];
        D = 0;
        MC = 0.5; %MC is the mach number
        [num,den] = ss2tf(A,B,C,D);
        G_aircraft_1 = tf(num, den);

    case 2
        A = [-1.702  50.72  263.5; 0.2201 -1.418 -31.99; 0 0 -14];
        B = [-272.2; 0; 14];
        C = C_root(index,:);%[1 1 1];
        D = 0;
        MC = 0.85;
        [num,den] = ss2tf(A,B,C,D);
        G_aircraft_2 = tf(num, den);

    case 3
        A = [-0.667  18.11 84.34; 0.08201 -0.6587 -10.81; 0 0 -14];
        B = [-85.09; 0; 14];
        C = C_root(index,:);%[1 1 1];
        D = 0;
        MC = 0.9;
        [num,den] = ss2tf(A,B,C,D);
        G_aircraft_3 = tf(num, den);

    case 4
        A = [-0.5162  26.96 178.9; -0.6896 -1.225 -30.38; 0 0 -14];
        B = [-175.6; 0; 14];
        C = C_root(index,:);%[1 1 1];
        D = 0;
        MC= 1.5;
        [num,den] = ss2tf(A,B,C,D);
        G_aircraft_4 = tf(num, den);
  end
end
hold on, figure(17), step(G_aircraft_1, 600),xlabel('Time'), ylabel('Amplitude'),
title('Example 1.6, step response for the aircraft models, FC=1'),legend('y = x1','y = x2','y = x3')

hold on, figure(18), step(G_aircraft_2, 400),xlabel('Time'), ylabel('Amplitude'),
title('Example 1.6, step response for the aircraft models, FC=2'),legend('y = x1','y = x2','y = x3')

hold on, figure(19), step(G_aircraft_3, 600),xlabel('Time'), ylabel('Amplitude'),
title('Example 1.6, step response for the aircraft models, FC=3'),legend('y = x1','y = x2','y = x3')

hold on, figure(20), step(G_aircraft_4),xlabel('Time'), ylabel('Amplitude'),
title('Example 1.6, step response for the aircraft models, FC=4'),legend('y = x1','y = x2','y = x3')
end
```

```
Warning: Ignoring extra legend entries.
Warning: Ignoring extra legend entries.
Warning: Ignoring extra legend entries.
Warning: Ignoring extra legend entries.
Warning: Ignoring extra legend entries.
Warning: Ignoring extra legend entries.
Warning: Ignoring extra legend entries.
Warning: Ignoring extra legend entries.
```

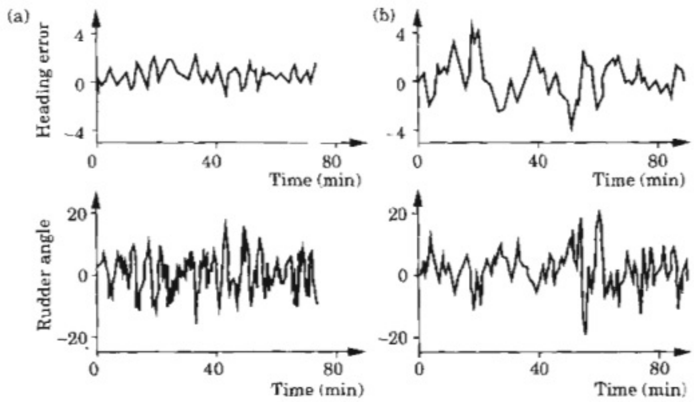
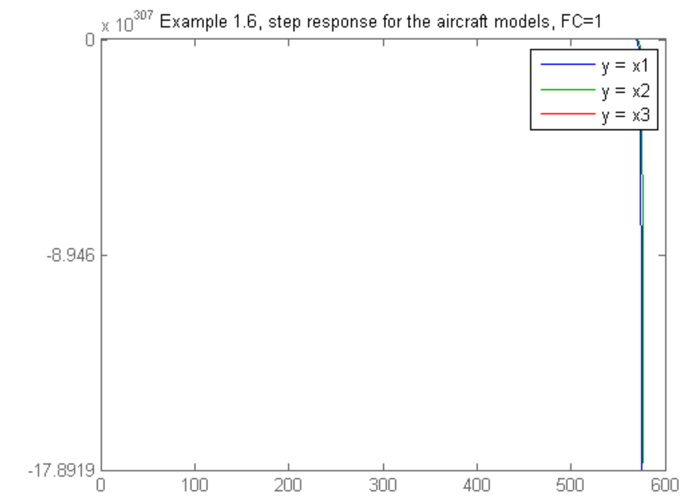
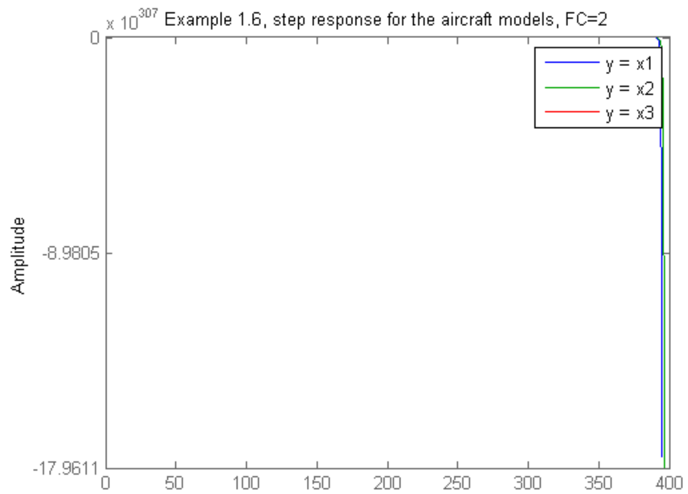
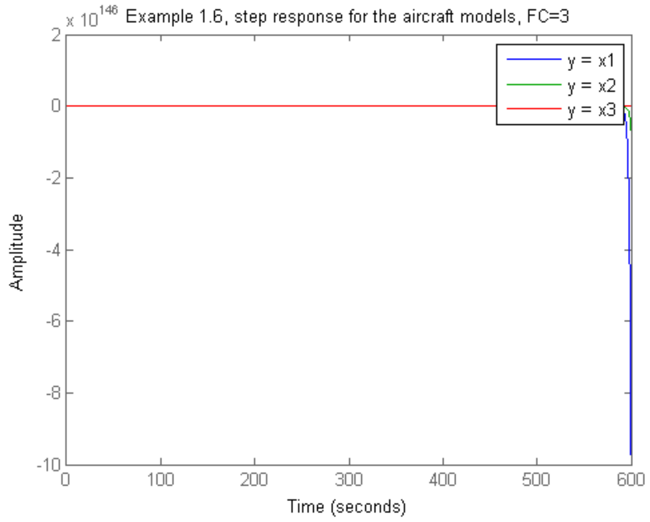
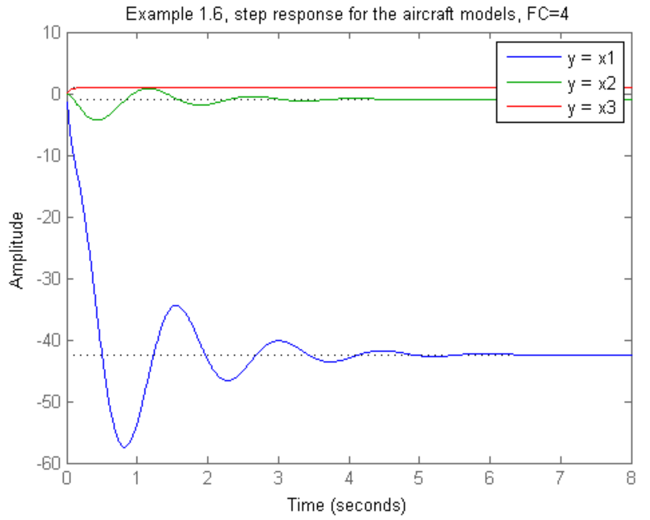


Figure 1.24 The figure shows the variations in heading and the corresponding rudder motions of a ship. (a) Adaptive autopilot. (b) Conventional autopilot based on a PID-like algorithm.





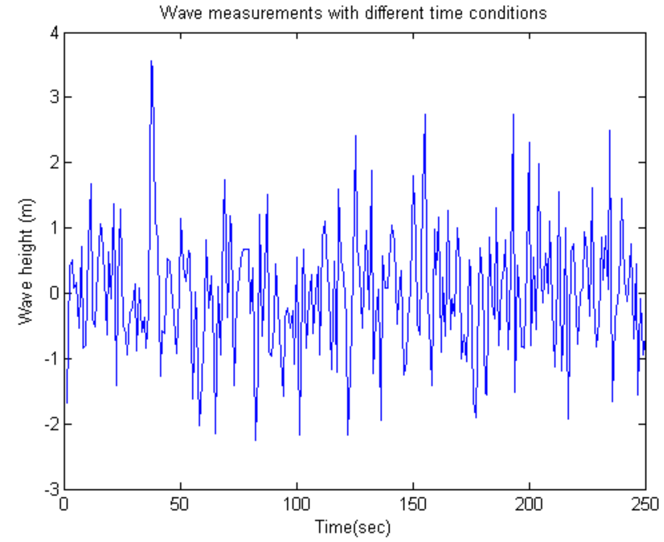


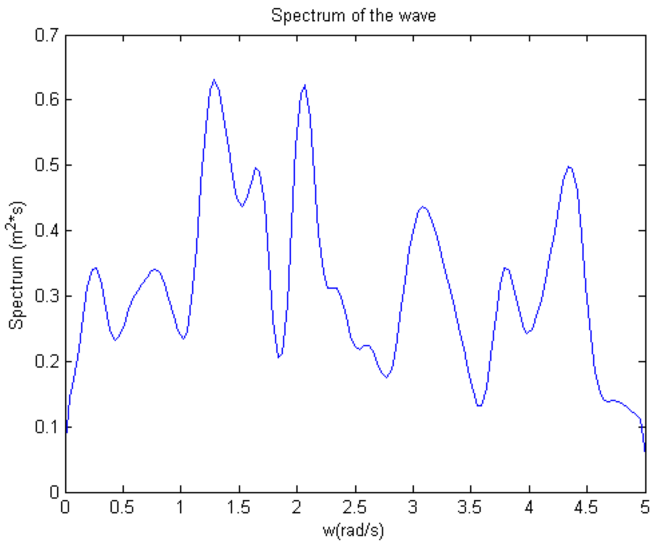


Example 1.7: Ship steering

A key problem in the design of an autopilot for ship steering is to compensate for the disturbing forces that act on the ship because of wind, waves, and current. The wave-generated forces are often the dominating forces. Waves have strong periodic components. The dominating wave frequency may change by a factor of 3 when the weather conditions change from light breeze to fresh gale. The frequency of the forces generated by the waves will change much more because it is also influenced by the velocity and heading of the ship. Examples of wave height and spectra for two weather conditions are shown in Fig. 21. It seems natural to take the nature of the wave disturbances into account in designing autopilots and roll dampers. Since the wave-induced forces change so much, it seems natural to adjust the controller parameters to cope with the disturbance characteristics. Positioning of ships and platforms is another example that is similar to ship steering. In this case the control system will typically have less control authority. This means that the platform to a greater extent has to "ride the waves" and can compensate only for a low-frequency component of the disturbances. This m.akes it even more critical to have a model for the disturbance pattern.

```
wind = randn(250,1);
figure(21),plot(wind),ylabel('Wave height (m)'), xlabel('Time(sec)'), title('Wave measurements with different time conditions')
windSpectrum = pwelch(wind);
w = linspace(0,5,size(windSpectrum,1));
figure(22), plot(w,windSpectrum), xlabel('w(rad/s)'), ylabel('Spectrum (m^2*s)'), title('Spectrum of the wave');
```





Question 1.8: Metal cutting machine

The machine is equipped with a force sensor, which measures the cutting force. A controller adjusts the feedback to maintain a constant cutting force. The cutting force is approximately given by $F = ka(\frac{v}{a})^{\alpha}$ The steady gain from the feed to force is $K = k\alpha v^{a-1}N^{-a}$ From the specification problem:

$K = 0.7N^{-a}$ On the other hand, K is the open loop gain in steady state, which implies

$K = \frac{1}{T_i}$ Run the q8.mdl file and set the parameters param = {a,N}

```
N=1;
a = 1;
Kq8 = 0.7*N^(-a);
```

Question 1.9 and 1.10: Processes with parameter variations

Consider a process $G(s) = \frac{K}{a_0s^2 + a_1s + a_2}$

where $K = K_0 + \Delta K$ and $a_i = a_{i0} + \Delta a_i$

And let the desired closed-loop response be given by:

$$Y_m = \frac{\omega_n^2}{s^2 + 2\zeta\omega s + \omega_n^2}$$

In order to simulate the process, run the file q910.mdl If you want to run the question 9, just make a0 = 0 and a1 = 1 The desired closed-loop for a first order will be $Y_m = \frac{K}{s+a}$

```
Ki = 1.5;
K0 = 1;
deltaK = -0.5;
Kq910 = K0+deltaK;
a0 = 1; %set to zero if the process is first order
a10 = 1.4;
delta_a1 = 2;
```

```
a1 = a10+delta_a1;
a20 = 1;
delta_a2 = 3;
a2 = a20+delta_a2;
```

Question 1.1: Definitions of Adaptive and Learning

Adaptive: serving or able to adapt; showing or contributing to adaptation : the adaptive coloring of a chameleon.

Learning: knowledge acquired by systematic study in any field of scholarly application. 2. the act or process of acquiring knowledge or skill. 3. Psychology. the modification of behavior through practice, training, or experience.

source: [5]

Question 1.2: Browsing manufacturers for adaptive controllers

Please, take a look on references [2], [3], [4]

Question 1.3 and 1.4: Why Adaptive Control?

Adaptive Control covers a set of techniques which provide a systematic approach for automatic adjustment of controllers in real time, in order to achieve or to maintain a desired level of control system performance when the parameters of the plant dynamic model are unknown and/or change in time. Consider first the case when the parameters of the dynamic model of the plant to be controlled are unknown but constant (at least in a certain region of operation). In such cases, although the structure of the controller will not depend in general upon the particular values of the plant model parameters, the correct tuning of the controller parameters cannot be done without knowledge of their values. Adaptive control techniques can provide an automatic tuning procedure in closed loop for the controller parameters. In such cases, the effect of the adaptation vanishes as time increases. Changes in the operation conditions may require a restart of the adaptation procedure. Now consider the case when the parameters of the dynamic model of the plant change unpredictably in time. These situations occur either because the environmental conditions change (ex: the dynamical characteristics of a robot arm or of a mechanical transmission depend upon the load; in a DC-DC converter the dynamic characteristics depend upon the load) or because we have considered simplified linear models for nonlinear systems (a change in operation condition will lead to a different linearized model). These situations may also occur simply because the parameters of the system are slowly time-varying (in a wiring machine the inertia of the spool is time-varying). In order to achieve and to maintain an acceptable level of control system performance when large and unknown changes in model parameters occur, an adaptive control approach has to be considered. In such cases, the adaptation will operate most of the time and the term non-vanishing adaptation fully characterizes this type of operation (also called continuous adaptation).

source: [6]

Conclusions

In this work were presented the concepts of Adaptive Control, and it's applications. A lot of examples and problems reinforced the theory and along them it could be seen that to control processes, adaptation is needed, once a large amount of dynamics systems are nonlinear.

References

[1] Åström, Karl J., and Björn Wittenmark. Adaptive control. Courier Corporation, 2013. APA

[2] http://www.omative.com/ACM.html

[3] http://www.transport-research.info/project/adaptive-control-manufacturing-processes-new-generation-jet-engine-components

[4] https://www.contemp.com.br/produtos/controladores-e-indicadores-de-temperatura-e-processos/controladores-de-temperatura-e-processos/controlador-de-processos-microprocessado-c709-pid-auto-adaptativo/

[5] www.dictionary.com

[6] Landau, Ioan Doré, et al. Adaptive control: algorithms, analysis and applications. Springer Science & Business Media, 2011.

Published with MATLAB® R2014a