

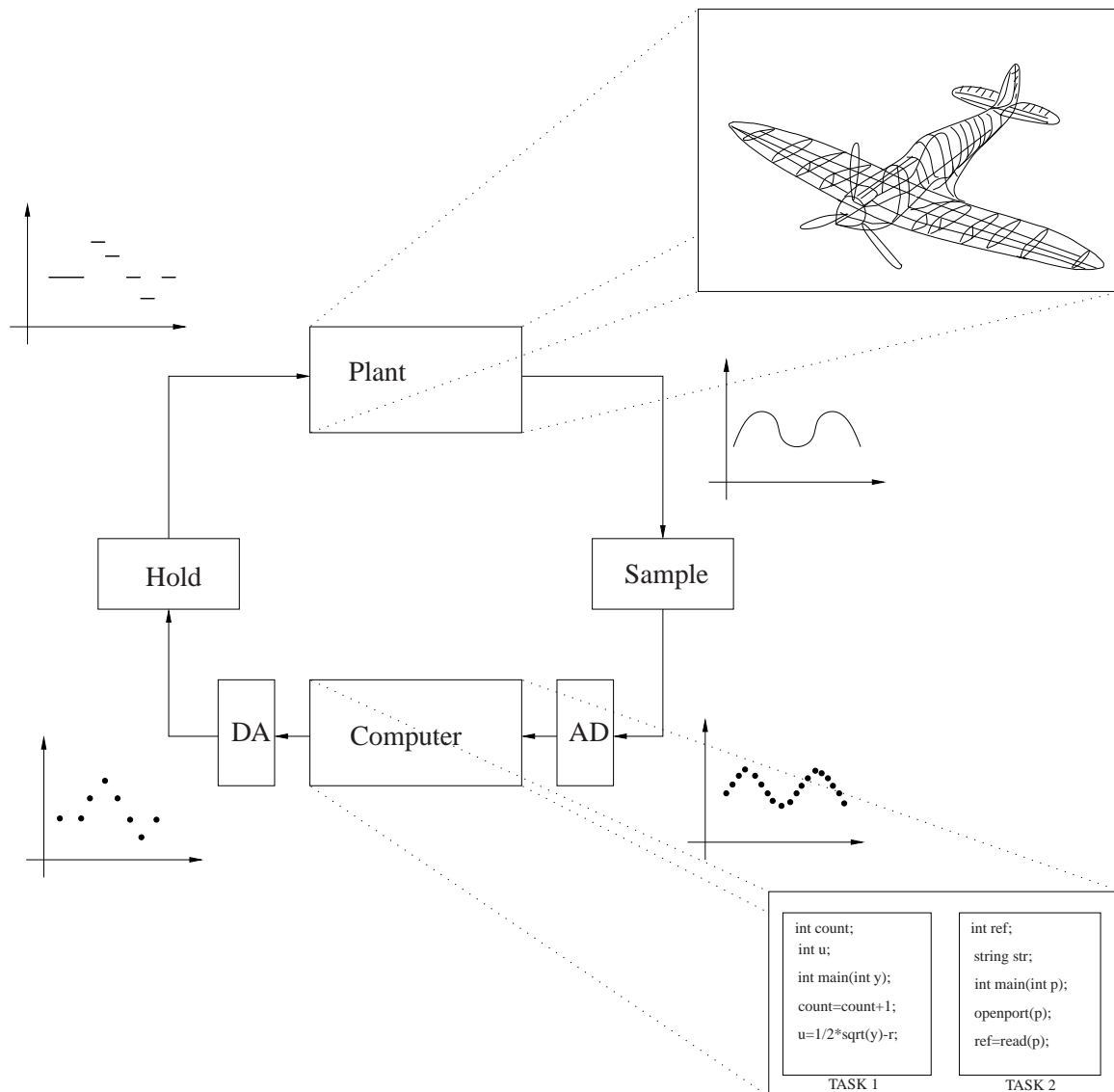


ROYAL INSTITUTE
OF TECHNOLOGY

EL2450 - Hybrid and Embedded Control Systems

Exercises

Alberto Speranzon, Oscar Flårdh, Magnus Lindhé,
Carlo Fischione and Karl Henrik Johansson
December 2008



Automatic Control Lab, School of Electrical Engineering
Royal Institute of Technology (KTH), Stockholm, Sweden

Preface

The present compendium has been developed by Alberto Speranzon, Oscar Flårdh and Karl Henrik Johansson in the beginning of 2005 for the course 2E1245 Hybrid and Embedded Control Systems, given at the Royal Institute of Technology, Stockholm. The material has been updated later in 2005 and in the beginning of 2006. Some of the exercises have been shamelessly borrowed (stolen) from other sources, and in that case a reference to the original source has been provided.

Alberto Speranzon, Oscar Flårdh and Karl Henrik Johansson, January 2006.

The material was edited and some problems and solutions were added in 2008, by Magnus Lindhé and Carlo Fischione. The course code also changed to EL2450.

CONTENTS

Exercises	7
I Time-triggered control	7
1 Review exercises: aliasing, z -transform, matrix exponential	8
2 Models of sampled systems	9
3 Analysis of sampled systems	12
4 Computer realization of controllers	17
5 Implementation aspects	20
II Event-triggered control	23
6 Real-time operating systems	24
7 Real-time scheduling	26
8 Models of computation I: Discrete-event systems	29
9 Models of computation II: Transition systems	31
III Hybrid control	34
10 Modeling of hybrid systems	35
11 Stability of hybrid systems	38
12 Verification of hybrid systems	42
13 Simulation and bisimulation	44
Solutions	46
I Time-triggered control	46
Review exercises	47
Models of sampled systems	54
Analysis of sampled systems	64
Computer realization of controllers	75
Implementation aspects	83
II Event-triggered control	88
Real-time operating systems	89

Real-time scheduling	93
Models of computation I: Discrete-event systems	101
Models of computation II: Transition systems	109
III Hybrid control	112
Modeling of hybrid systems	113
Stability of hybrid systems	116
Verification of hybrid systems	125
Simulation and bisimulation	129
Bibliography	129

Exercises

Part I

Time-triggered control

1 Review exercises: aliasing, z-transform, matrix exponential

EXERCISE 1.1 (Ex. 7.3 in [13])

Consider a sampling and reconstruction system as in Figure 1.1. The input signal is $x(t) = \cos(\omega_0 t)$. The Fourier transform of the signal is

$$X(j\omega) = \pi [\delta(\omega - \omega_0) + \delta(\omega + \omega_0)]$$

and the reconstruction (low-pass) filter has the transfer function

$$F(j\omega) = \begin{cases} h, & -\omega_s/2 < \omega < \omega_s/2 \\ 0, & \text{else} \end{cases}$$

where $\omega_s = \frac{2\pi}{h}$ is the sampling frequency. Find the reconstructed output signal $x_r(t)$ for the following input frequencies

- (a) $\omega_0 = \omega_s/6$
- (b) $\omega_0 = 2\omega_s/6$
- (c) $\omega_0 = 4\omega_s/6$
- (d) $\omega_0 = 5\omega_s/6$

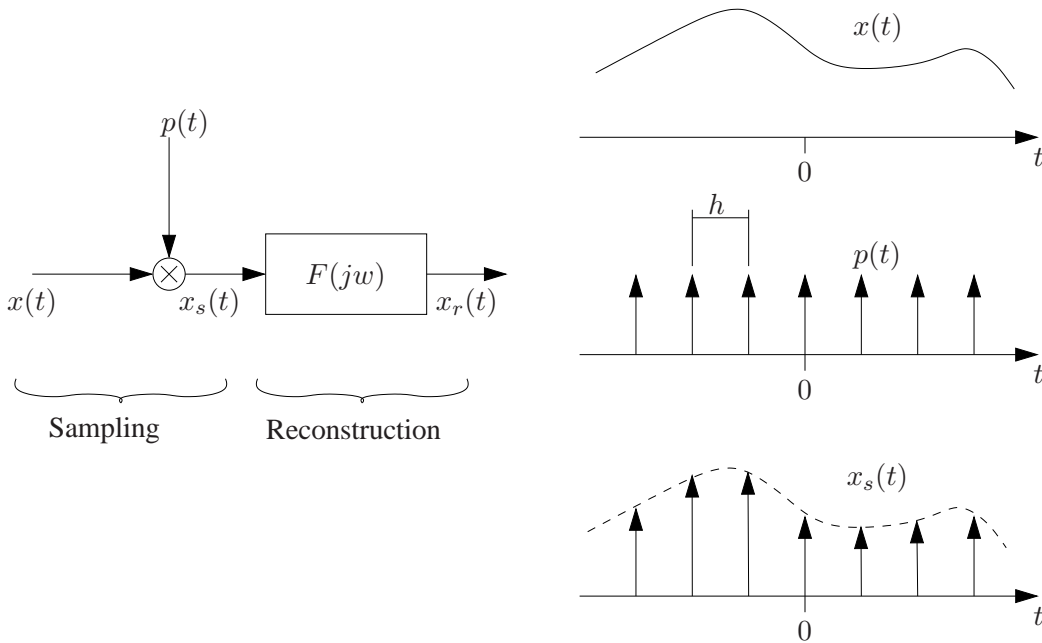


Figure 1.1: Sampling and reconstruction of a band-limited signal.

EXERCISE 1.2

Let the matrix A be

$$A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}.$$

Compute the matrix exponential e^A .

EXERCISE 1.3

Compute the z -transform of

$$x(kh) = e^{-kh/T} \quad T > 0.$$

EXERCISE 1.4

Compute the z -transform of

$$x(kh) = \sin(wh)$$

EXERCISE 1.5

Given the following system described by the following difference equation

$$y(k+2) - 1.5y(k+1) + 0.5y(k) = u(k+1)$$

with initial condition $y(0) = 0.5$ and $y(1) = 1.25$, determine the output when the input $u(k)$ is a unitary step.

2 Models of sampled systems**EXERCISE 2.1** (Ex. 2.1 in [2])

Consider the scalar system

$$\begin{aligned} \frac{dx}{dt} &= -ax + bu \\ y &= cx. \end{aligned}$$

Let the input be constant over periods of length h . Sample the system and discuss how the poles of the discrete-time system vary with the sampling frequency.

EXERCISE 2.2

Consider the following continuous-time transfer function

$$G(s) = \frac{1}{(s+1)(s+2)}.$$

The system is sampled with sampling period $h = 1$.

- (a) Derive a state-space representation of the sampled system.
- (b) Find the pulse-transfer function corresponding to the system in (a).

EXERCISE 2.3 (Ex. 2.2 in [2])

Derive the discrete-time system corresponding to the following continuous-time systems when a zero order-hold circuit is used

(a)

$$\dot{x} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u$$

$$y = (1 \ 0) x$$

(b)

$$\frac{d^2y}{dt^2} + 3\frac{dy}{dt} + 2y = \frac{du}{dt} + 3u$$

(c)

$$\frac{d^3y}{dt^3} = u$$

EXERCISE 2.4 (Ex. 2.3 in [2])

The following difference equations are assumed to describe continuous-time systems sampled using a zero-order-hold circuit and the sampling period h . Determine, if possible, the corresponding continuous-time systems.

(a)

$$y(kh) - 0.5y(kh - h) = 6u(kh - h)$$

(b)

$$x(kh + h) = \begin{pmatrix} -0.5 & 1 \\ 0 & -0.3 \end{pmatrix} x(kh) + \begin{pmatrix} 0.5 \\ 0.7 \end{pmatrix} u(kh)$$

$$y(kh) = (1 \ 1) x(kh)$$

(c)

$$y(kh) + 0.5y(kh - h) = 6u(kh - h)$$

EXERCISE 2.5 (Ex. 2.11 in [2])

The transfer function of a motor can be written as

$$G(s) = \frac{1}{s(s+1)}.$$

Determine:

- (a) the sampled system
- (b) the pulse-transfer function
- (c) the pulse response
- (d) a difference equation relating input and output
- (e) the variation of the poles and zeros of the pulse-transfer function with the sampling period

EXERCISE 2.6 (Ex. 2.12 in [2])

A continuous-time system with transfer function

$$G(s) = \frac{1}{s} e^{-s\tau}$$

is sampled with sampling period $h = 1$, where $\tau = 0.5$.

- Determine a state-space representation of the sampled system. What is the order of the sampled-system?
- Determine the pulse-transfer function and the pulse response of the sampled system
- Determine the poles and zeros of the sampled system.

EXERCISE 2.7 (Ex. 2.13 in [2])

Solve Problem 2.6 with

$$G(s) = \frac{1}{s+1} e^{-s\tau}$$

and $h = 1$ and $\tau = 1.5$.

EXERCISE 2.8 (Ex. 2.15 in [2])

Determine the polynomials $A(q)$, $B(q)$, $A^*(q^{-1})$ and $B^*(q^{-1})$ so that the systems

$$A(q)y(k) = B(q)u(k)$$

and

$$A^*(q^{-1})y(k) = B^*(q^{-1})u(k-d)$$

represent the system

$$y(k) - 0.5y(k-1) = u(k-9) + 0.2u(k-10).$$

What is d ? What is the order of the system?

EXERCISE 2.9 (Ex. 2.17 in [2])

Use the z-transform to determine the output sequence of the difference equation

$$y(k+2) - 1.5y(k+1) + 0.5y(k) = u(k+1)$$

when $u(k)$ is a step at $k = 0$ and when $y(0) = 0.5$ and $y(-1) = 1$.

EXERCISE 2.10

Consider the following continuous time controller

$$U(s) = -\frac{s_0s + s_1}{s + r_1}Y(s) + \frac{t_0s + t_1}{s + r_1}R(s)$$

where s_0, s_1, t_0, t_1 and r_1 are parameters that are chosen to obtain the desired closed-loop performance. Discretize the controller using exact sampling by means of sampled control theory. Assume that the sampling interval is h , and write the sampled controller on the form $u(kh) = -H_y(q)y(kh) + H_r(q)r(kh)$.

EXERCISE 2.11 (Ex. 2.21 in [2])

If $\beta < \alpha$, then

$$\frac{s + \beta}{s + \alpha}$$

is called a lead filter (i.e. it gives a phase advance). Consider the discrete-time system

$$\frac{z + b}{z + a}$$

- Determine when it is a lead filter
- Simulate the step response for different pole and zero locations

3 Analysis of sampled systems

EXERCISE 3.1 (Ex. 3.2 in [2])

Consider the system in Figure 3.1 and let

$$H(z) = \frac{K}{z(z - 0.2)(z - 0.4)} \quad K > 0$$

Determine the values of K for which the closed-loop system is stable.

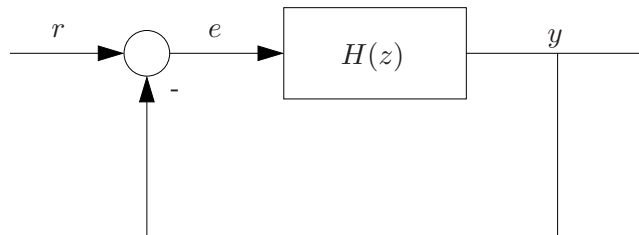


Figure 3.1: Closed-loop system for Problem 3.1.

EXERCISE 3.2 (Ex. 3.3 in [2])

Consider the system in Figure 3.2. Assume the sampling is periodic with period h , and that the D-A converter holds the control signal constant over a sampling interval. The control algorithm is assumed to be

$$u(kh) = K(r(kh - \tau) - y(kh - \tau))$$

where $K > 0$ and τ is the computation time. The transfer function of the process is

$$G(s) = \frac{1}{s}.$$

- How large are the values of the regulator gain, K , for which the closed-loop system is stable when $\tau = 0$ and $\tau = h$?
- Compare this system with the corresponding continuous-time systems, that is, when there is a continuous-time proportional controller and a time delay in the process.

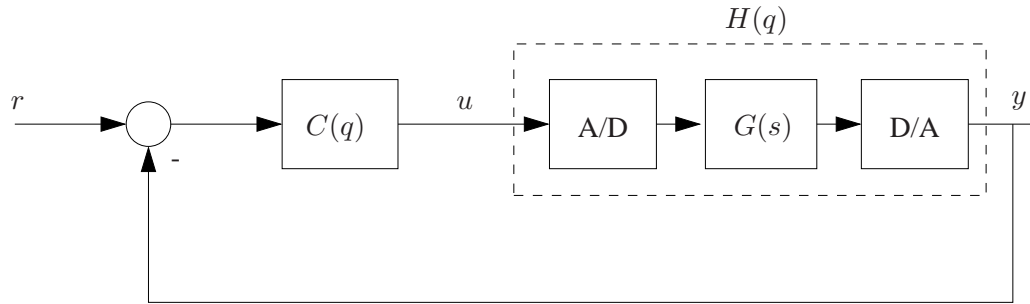


Figure 3.2: Closed-loop system for Problem 3.2.

EXERCISE 3.3 (Ex. 3.6 in [2])

Is the following system (a) observable, (b) reachable?

$$\begin{aligned} x(k+1) &= \begin{pmatrix} 0.5 & -0.5 \\ 0 & 0.25 \end{pmatrix} x(k) + \begin{pmatrix} 6 \\ 4 \end{pmatrix} u(k) \\ y(k) &= \begin{pmatrix} 2 & -4 \end{pmatrix} x(k) \end{aligned}$$

EXERCISE 3.4 (Ex. 3.7 in [2])

Is the following system reachable?

$$x(k+1) = \begin{pmatrix} 1 & 0 \\ 0 & 0.5 \end{pmatrix} x(k) + \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} u(k).$$

Assume that a scalar input $v(k)$ such that

$$u(k) = \begin{pmatrix} 1 \\ -1 \end{pmatrix} v(k)$$

is introduced. Is the system reachable from $v(k)$?

EXERCISE 3.5 (Ex. 3.11 in [2])

Determine the stability and the stationary value of the output for the system described by Figure 3.2 with

$$H(q) = \frac{1}{q(q - 0.5)}$$

where r is a step function and $C(q) = K$ (proportional controller), $K > 0$.

EXERCISE 3.6 (Ex. 3.12 in [2])

Consider the Problem 3.5. Determine the steady-state error between the reference signal r and the output y , when r is a unit ramp, that is $r(k) = k$. Assume $C(q)$ to be a proportional controller.

EXERCISE 3.7 (Ex. 3.18 in [2])

Consider a continuous-time (CT) system

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t).\end{aligned}$$

The zero-order hold sampling of CT gives the discrete-time (DT) system

$$\begin{aligned}x(kh + h) &= \Phi x(kh) + \Gamma u(kh) \\ y(kh) &= Cx(kh).\end{aligned}$$

Consider the following statements:

- (a) CT stable \Rightarrow DT stable
- (b) CT unstable \Rightarrow DT unstable
- (c) CT controllable \Rightarrow DT controllable
- (d) CT observable \Rightarrow DT observable.

Which statements are true and which are false (explain why) in the following cases:

- (i) For all sampling intervals $h > 0$
- (ii) For all $h > 0$ except for isolated values
- (iii) Neither (i) nor (ii).

EXERCISE 3.8 (Ex. 3.20 in [2])

Given the system

$$(q^2 + 0.4q)y(k) = u(k),$$

- (a) for which values of K in the proportional controller

$$u(k) = K(r(k) - y(k))$$

is the closed-loop system stable?

- (b) Determine the stationary error $r - y$ when r is a step and $K=0.5$ in the controller (a).

EXERCISE 3.9 (Ex. 4.1 in [2])

A general second-order discrete-time system can be written as

$$\begin{aligned}x(k+1) &= \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} x(k) + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} u(k) \\ y(k) &= (c_1 \quad c_2) x(k).\end{aligned}$$

Determine a state-feedback controller in the form

$$u(k) = -Lx(k)$$

such that the characteristic equation of the closed-loop system is

$$z^2 + p_1z + p_2 = 0.$$

Use the previous result to compute the deadbeat controller for the double integrator.

EXERCISE 3.10 (Ex. 4.2 in [2])

Given the system

$$\begin{aligned}x(k+1) &= \begin{pmatrix} 1 & 0.1 \\ 0.5 & 0.1 \end{pmatrix} x(k) + \begin{pmatrix} 1 \\ 0 \end{pmatrix} u(k) \\ y(k) &= (1 \quad 1) x(k).\end{aligned}$$

Determine a linear state-feedback controller

$$u(k) = -Lx(k)$$

such that the poles of the closed-loop system are placed in 0.1 and 0.25.

EXERCISE 3.11 (Ex. 4.5 in [2])

The system

$$\begin{aligned}x(k+1) &= \begin{pmatrix} 0.78 & 0 \\ 0.22 & 1 \end{pmatrix} x(k) + \begin{pmatrix} 0.22 \\ 0.03 \end{pmatrix} u(k) \\ y(k) &= (0 \quad 1) x(k).\end{aligned}$$

represents the normalized motor for the sampling interval of $h = 0.25$. Determine observers for the state based on the output by using each of the following:

- (a) Direct calculation.
- (b) An full-state observer.
- (c) The reduced-order observer.

EXERCISE 3.12 (Ex. 4.8 in [2])

Given the discrete-time system

$$x(k+1) = \begin{pmatrix} 0.5 & 1 \\ 0.5 & 0.7 \end{pmatrix} x(k) + \begin{pmatrix} 0.2 \\ 0.1 \end{pmatrix} u(k) + \begin{pmatrix} 1 \\ 0 \end{pmatrix} v(k)$$

$$y(k) = (1 \ 0) x(k).$$

where v is a constant disturbance. Determine controller such that the influence of v can be eliminated in steady state in each of the following cases:

- The state and v can be measured.
- The state can be measured.
- Only the output can be measured.

EXERCISE 3.13 (Ex. 4.6 in [2])

Figure 3.13 shows a system with two tanks, where the input signal is the flow to the first tank and the output is the level of water in the second tank. The continuous-time model of the system is

$$\dot{x} = \begin{pmatrix} -0.0197 & 1 \\ 0.0178 & -0.0129 \end{pmatrix} x + \begin{pmatrix} 0.0263 \\ 0 \end{pmatrix} u$$

$$y = (0 \ 1) x.$$

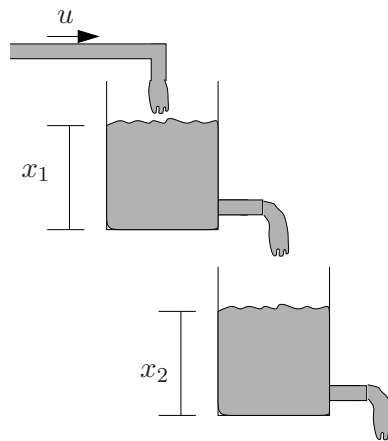


Figure 3.13: Closed-loop system for Problem 3.13.

- Sample the system with $h = 12$.
- Verify that the pulse-transfer operator for the system is

$$H(q) = \frac{0.030q + 0.026}{q^2 - 1.65q + 0.68}$$

- Determine a full-state observer. Choose the gain such that the observer is twice as fast as the open-loop system.

EXERCISE 3.14

Consider the following scalar linear system

$$\begin{aligned}\dot{x}(t) &= -5x(t) + u(t) \\ y(t) &= x(t).\end{aligned}$$

- (a) Sample the system with sampling period $h = 1$,
- (b) Show, using Lyapunov result, that the sampled system is stable when the input $u(kh) = 0$ for $k \geq 0$.

EXERCISE 3.15

Consider the following linear system

$$\begin{aligned}\dot{x}(t) &= \begin{pmatrix} -1 & 0 \\ 0 & -2 \end{pmatrix} x(t) + \begin{pmatrix} 1 \\ 1 \end{pmatrix} u(t) \\ y(t) &= x(t).\end{aligned}$$

- (a) Sample the system with sampling period $h = 1$
- (b) Design a controller that place the poles in 0.1 and 0.2.
- (c) Show, using Lyapunov result, that the closed loop sampled system is stable

4 Computer realization of controllers**EXERCISE 4.1**

Consider the following pulse-transfer

$$H(z) = \frac{z - 1}{(z - 0.5)(z - 2)}$$

- (a) Design a digital PI controller

$$H_c(z) = \frac{(K + K_i)z - K}{z - 1}$$

that places the poles of the closed-loop system in the origin.

- (b) Find a state-space representation of the digital controller in (a).

EXERCISE 4.2 (Ex. 8.2 in [2])

Use different methods to make an approximation of the transfer function

$$G(s) = \frac{a}{s + a}$$

- (a) Euler's method
- (b) Tustin's approximation
- (c) Tustin's approximation with pre-warping using $\omega_1 = a$ as warping frequency

EXERCISE 4.3 (Ex. 8.3 in [2])

The lead network with transfer function

$$G_\ell(s) = 4 \frac{s + 1}{s + 2}$$

Give a phase advance of about 20° at $\omega_c = 1.6 \text{ rad/s}$. Approximate the network for $h = 0.25$ using

- (a) Euler's method
- (b) Backward differences
- (c) Tustin's approximation
- (d) Tustin's approximation with pre-warping using $\omega_1 = \omega_c$ as warping frequency

EXERCISE 4.4 (Ex. 8.7 in [2])

Consider the tank system in Problem 2.13. Assume the following specifications:

1. The steady-state error after a step in the reference value is zero
 2. The crossover frequency of the compensated system is 0.025 rad/s
 3. The phase margin is about 50° .
- (a) Design a PI-controller such that the specifications are fulfilled.
 - (b) Determine the poles and the zeros of the closed-loop system. What is the damping corresponding to the complex poles?
 - (c) Choose a suitable sampling interval and approximate the continuous-time controller using Tustin's method with pre-warping. Use the crossover frequency as warping frequency.

EXERCISE 4.5 (Ex. 8.4 in [2])

The choice of sampling period depends on many factors. One way to determine the sampling frequency is to use continuous-time arguments. Approximate the sampled system as the hold circuit followed by the continuous-time system. Assuming that the phase margin can be decreased by 5° to 15° , verify that a rule of thumb in selecting the sampling frequency is

$$h\omega_c \approx 0.15 \text{ to } 0.5$$

where ω_c is the crossover frequency of the continuous-time system.

EXERCISE 4.6 (Ex. 8.12 in [2])

Consider the continuous-time double integrator described by

$$\begin{aligned}\dot{x} &= \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u \\ y &= \begin{pmatrix} 1 & 0 \end{pmatrix} x.\end{aligned}$$

Assume that a time-continuous design has been made giving the controller

$$\begin{aligned}u(t) &= 2r(t) - (12)\hat{x}(t) \\ \frac{d\hat{x}(t)}{dt} &= A\hat{x}(t) + Bu(t) + K(y(t) - C\hat{x}(t))\end{aligned}$$

with $K^T = (1, 1)$.

- Assume that the controller should be implemented using a computer. Modify the controller (not the observer part) for the sampling interval $h = 0.2$ using the approximation for state models.
- Approximate the observer using a backward-difference approximation

EXERCISE 4.7

Consider the following continuous time controller

$$U(s) = -\frac{s_0s + s_1}{s + r_1}Y(s) + \frac{t_0s + t_1}{s + r_1}R(s)$$

where s_0, s_1, t_0, t_1 and r_1 are parameters that are chosen to obtain the desired closed-loop performance.

- Discretize the controller using forward difference approximation. Assume that the sampling interval is h , and write the sampled controller on the form $u(kh) = -H_y(q)y(kh) + H_r(q)r(kh)$.
- Assume the following numerical values of the coefficients: $r_1 = 10$, $s_0 = 1$, $s_1 = 2$, $t_0 = 0.5$ and $t_1 = 10$. Compare the discretizations obtained in part (a) for the sampling intervals $h = 0.01$, $h = 0.1$ and $h = 1$. Which of those sampling intervals should be used for the forward difference approximation?

EXERCISE 4.8

Consider the following continuous-time controller in state-space form

$$\begin{aligned}\dot{x} &= Ax + Be \\ u &= Cx + De\end{aligned}$$

- Derive the backward-difference approximation in state-space form of the controller, i.e. derive Φ_c , Γ_c , H and J for a system

$$\begin{aligned}w(k+1) &= \Phi_c w(k) + \Gamma_c e(k) \\ u(k) &= H w(k) + J e(k)\end{aligned}$$

(b) Prove that the Tustin's approximation of the controller is given by

$$\begin{aligned}\Phi_c &= \left(I + \frac{A_c h}{2}\right) \left(I - \frac{A_c h}{2}\right)^{-1} \\ \Gamma_c &= \left(I - \frac{A_c h}{2}\right)^{-1} \frac{B_c h}{2} \\ H &= C_c \left(I - \frac{A_c h}{2}\right)^{-1} \\ J &= D_c + C_c \left(I - \frac{A_c h}{2}\right)^{-1} \frac{B_c h}{2}.\end{aligned}$$

5 Implementation aspects

EXERCISE 5.1

Consider the discrete-time controller characterized by the pulse-transfer function

$$H(z) = \frac{1}{(z-1)(z-1/2)(z^2 + 1/2z + 1/4)}.$$

Implement the controller in parallel form.

EXERCISE 5.2

(a) Given the system in Figure 5.2, find the controller $C_s(s)$ such that the closed loop transfer function from r to y becomes

$$H_{cl} = \frac{C(s)P(s)}{1 + C(s)P(s)} e^{-s\tau}$$

(b) Let

$$\begin{aligned}P(s) &= \frac{1}{s+1} \\ H_{cl}(s) &= \frac{8}{s^2 + 4s + 8} e^{-s\tau}\end{aligned}$$

find the expression for the Smith predictor $C_s(s)$.

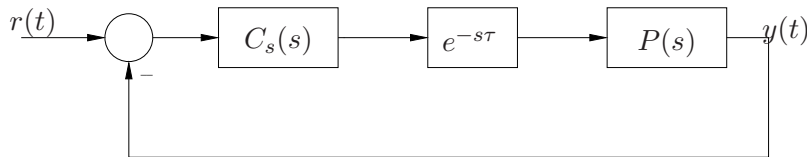


Figure 5.2: System of Problem 5.2.

EXERCISE 5.3

A process with transfer function

$$P(z) = \frac{z}{z - 0.5}$$

is controlled by the PI-controller

$$C(z) = K_p + K_i \frac{z}{z - 1}$$

where $K_p = 0.2$ and $K_i = 0.1$. The control is performed over a wireless network, as shown in Figure 5.3. Due to retransmission of dropped packets, the network induces time-varying delays. How large can the maximum delay be, so that the closed loop system is stable?

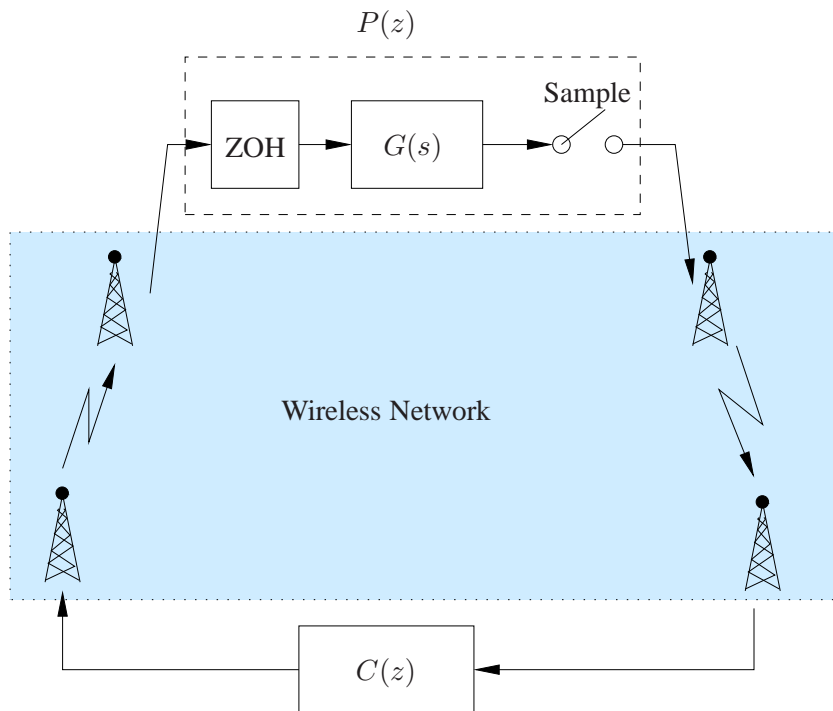


Figure 5.3: Closed loop system for Problem 5.3.

EXERCISE 5.4 (Inspired by Ex. 9.15 in [2])

Two different algorithms for a PI-controller are listed. Use the linear model for roundoff to analyze the sensitivity of the algorithms to roundoff in multiplications and divisions. Assume that the multiplications happen in the order as they appear in the formula and that they are executed before the division.

Algorithm 1:

```
repeat
  e:=r-y
  u:=k*(e+i)
  i:=i+e*h/ti
forever
```

Algorithm 2:

```
repeat
  e:=r-y
  u:=i+k*e
  i:=k*i+k*h*e/ti
forever
```

EXERCISE 5.5

Consider a first-order system with the discrete transfer function

$$H(z) = \frac{1}{1 - az^{-1}} \quad a = \frac{1}{8}.$$

Assume the controller is implemented using fixed point arithmetic with 8 bits word length and $h = 1$ second. Determine the system's unit step response for sufficient number of samples to reach steady-state. Assume that the data representation consists of

- 1 bit for sign
- 2 bits for the integer part
- 5 bits for the fraction part

and consider the cases of truncation and round-off.

Part II

Event-triggered control

6 Real-time operating systems

EXERCISE 6.1

In an embedded control system the control algorithm is implemented as a task in a CPU. The control task J_c can compute the new control action only after the acquisition task J_a has acquired new sensor measurements. The two tasks are independent and they share the same CPU. Suppose the sampling period is $h = 0.4$ seconds and the tasks have the following specifications

$$\begin{array}{c|c} & C_i \\ \hline J_c & 0.1 \\ J_a & 0.2 \end{array}$$

We assume that the period T_i and the deadline D_i are the same for the two tasks, and the release time is 0 for both tasks.

- Is possible to schedule the two tasks J_c and J_a ? Determine the schedule length and draw the schedule.
- Suppose that a third task is running in the CPU. The specifications for the task are

$$\begin{array}{c|ccc} & C_i & T_i = D_i & r_i \\ \hline J_x & 0.2 & 0.8 & 0.3 \end{array}$$

and we assume that the task J_x has higher priority than the tasks J_c and J_a . We also assume the CPU can handle preemption. Are the three tasks schedulable? Draw the schedule and determine the worst-case response time for the control task J_c .

EXERCISE 6.2

A digital PID controller is used to control the plant, which sampled with period $h = 2$ has the following transfer function

$$P(z) = \frac{1}{100} \frac{z - 0.1}{z - 0.5}.$$

The control law is

$$C(z) = 15 \left(1 + \frac{z}{z - 1} \right).$$

Assume that the control task J_c is implemented on a computer and has $C_c = 1$ as the worst case computation time. Assume that a higher priority interrupt occurs at time $t = 2$ which has a worst case computation time C_I . Determine the largest value of C_I such that the closed loop system is stable.

EXERCISE 6.3

A robot has been designed with three different tasks J_A , J_B , J_C , with increasing priority. The task J_A is a low priority thread which implements the DC-motor controller, the task J_B periodically send a "ping" through the wireless network card so that it is possible to know if the system is running. Finally the task J_C , with highest priority, is responsible to check the status of the data bus between two I/O ports, as shown in Figure 6.3. The control task is at low priority since the robot is moving very slowly in a cluttered environment. Since the data bus is a shared resource there is a semaphore that regulates the access to the bus. The tasks have the following characteristics

	T_i	C_i
J_A	8	4
J_B	5	2
J_C	1	0.1

Assuming the kernel can handle preemption, analyze the following possible working condition:

- at time $t = 0$, the task J_A is running and acquires the bus in order to send a new control input to the DC-motors,
- at time $t = 2$ the task J_C needs to access the bus meanwhile the control task J_A is setting the new control signal,
- at the same t J_B is ready to be executed to send the "ping" signal.

(a) Show graphically which tasks are running. What happens to the high priority task J_C ? Compute the response time of J_C in this situation.

(b) Suggest a possible way to overcome the problem in (a).

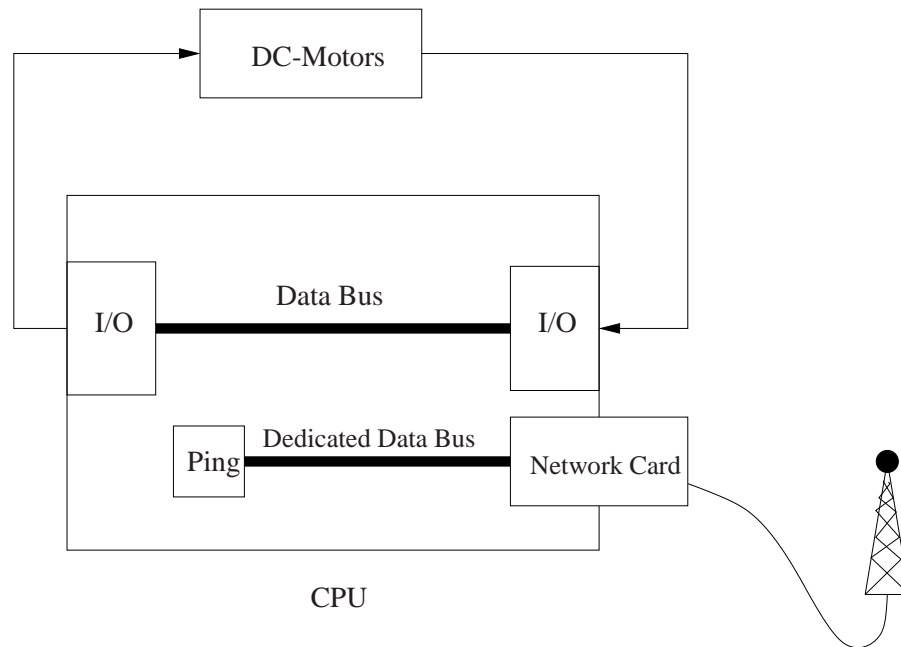


Figure 6.3: Schedule for the control task J_C and the task handling the interrupt, of Problem 6.2.

EXERCISE 6.4 (Jackson's algorithm, page 52 in [3])

We consider here the Jackson's algorithm to schedule a set \mathcal{J} of n aperiodic tasks minimizing a quantity called *maximum lateness* and defined as

$$L_{max} := \max_{i \in \mathcal{J}} (f_i - d_i).$$

All the tasks consist of a single job, have synchronous arrival times but have different computation times and deadlines. They are assumed to be independent. Each task can be characterized by two parameters, deadline d_i and computation time C_i

$$\mathcal{J} = \{J_i | J_i = J_i(C_i, d_i), \quad i = 1, \dots, n\}.$$

The algorithm, also called *Earliest Due Date* (EDD), can be expressed by the following rule

Theorem 1. Given a set of n independent tasks, any algorithm that executes the tasks in order of nondecreasing deadlines is optimal with respect to minimizing the maximum lateness.

- (a) Consider a set of 5 independent tasks simultaneously activated at time $t = 0$. The parameters are indicated in the following table

	J_1	J_2	J_3	J_4	J_5
C_i	1	1	1	3	2
d_i	3	10	7	8	5

Determine what is the maximum lateness using the scheduling algorithm EDD.

- (b) Prove the optimality of the algorithm.

7 Real-time scheduling

EXERCISE 7.1 (Ex. 4.3 in [3])

Verify the schedulability and construct the schedule according to the rate monotonic algorithm for the following set of periodic tasks

	C_i	T_i
J_1	1	4
J_2	2	6
J_3	3	10

EXERCISE 7.2 (Ex. 4.4 in [3])

Verify the schedulability under EDF of the task set given in Exercise 7.1 and then construct the corresponding schedule.

EXERCISE 7.3

Consider the following set of tasks

	C_i	T_i	D_i
J_1	1	3	3
J_2	2	4	4
J_3	1	7	7

Are the tasks schedulable with rate monotonic algorithm? Are the tasks schedulable with earliest deadline first algorithm?

EXERCISE 7.4

Consider the following set of tasks

	C_i	T_i	D_i
J_1	1	4	4
J_2	2	5	5
J_3	3	10	10

Assume that task J_1 is a control task. Every time that a measurement is acquired, task J_1 is released. When executing, it computes an updated control signal and outputs it.

- Which scheduling of RM or EDF is preferable if we want to minimize the delay between the acquisition and control output?
- Suppose that J_2 is also a control task and that we want its maximum delay between acquisition and control output to be two time steps. Suggest a schedule which guarantees a delay of maximally two time steps, and prove that all tasks will meet their deadlines.

EXERCISE 7.5

Consider the two tasks J_1 and J_2 with computation times, periods and deadlines as defined by the following table:

	C_i	T_i	D_i
J_1	1	3	3
J_2	1	4	4

- Suppose the tasks are scheduled using the rate monotonic algorithm. Will J_1 and J_2 meet their deadlines according to the schedulability condition on the utilization factor? What is the schedule length, i.e., the shortest time interval that is necessary to consider in order to describe the whole time evolution of the scheduler? Plot the time evolution of the scheduler when the release time for both tasks is at $t = 0$.
- If the two tasks implement a controller it is important to know what is the worst-case delay between the time the controller is ready to sample and the time a new input $u(kh)$ is ready to be released. Find the worst-case response time for J_1 and J_2 . Compare with the result in (a).

EXERCISE 7.6

Consider the set of periodic tasks given in the table below:

	C_i	T_i	O_i
J_1	1	3	1
J_2	2	5	1
J_3	1	6	0

where for task i , C_i the worst-case execution time, T_i denotes the period, and O_i the offset for the respective tasks. Assume that the deadlines coincide with the period. The offset denotes the relative release time of the first task instance for each task. Assume that all tasks are released at time 0 with their respective offset O_i .

- (a) Determine the schedule length.

Determine the worst-case response time for task J_2 for each of the following three scheduling policies:

- (b) Rate-monotonic scheduling
 (c) Deadline-monotonic scheduling
 (d) Earliest-deadline-first scheduling

EXERCISE 7.7

A control task J_c is scheduled in a computer together with two other tasks J_1 and J_2 . Assume that the three tasks are scheduled using a rate monotonic algorithm. Assume that the release time for all tasks are at zero and that the tasks have the following characteristics

	C_i	T_i	D_i
J_1	1	4	4
J_2	1	6	6
J_c	2	5	5

- (a) Is the set of tasks schedulable with rate monotonic scheduling? Determine the worst-case response time for the control task J_c .
 (b) Suppose the control task implements a sampled version of the continuous-time controller with delay

$$\begin{aligned} \dot{x}(t) &= Ax(t) + By(t - \tau) \\ u(t) &= Cx(t) \end{aligned}$$

where we let τ be the worst-case response time R_c of the task J_c . Suppose that the sampling period of the controller is $h = 2$ and $R_c = 3$. Derive a state-space representation for the sampled controller. Suggest also an implementation of the controller by specifying a few lines of computer code.

- (c) In order to improve performance the rate monotonic scheduling is substituted by a new scheduling algorithm that give highest priority to the control task and intermediate and lowest to the task J_1 and J_2 , respectively. Are the tasks schedulable in this case?

EXERCISE 7.8

Compute the maximum processor utilization that can be assigned to a polling server to guarantee the following periodic task will meet their deadlines

	C_i	T_i
J_1	1	5
J_2	2	8

EXERCISE 7.9

Together with the periodic tasks

	C_i	T_i
J_1	1	4
J_2	1	8

we want to schedule the following aperiodic tasks with a polling server having $T_s = 5$ and $C_s = 2$. The aperiodic tasks are

	a_i	C_i
a_1	2	3
a_2	7	2
a_3	9	1
a_3	29	4

EXERCISE 7.10

Consider the set of tasks in Problem 7.5, assuming that an aperiodic task could ask for CPU time. In order to handle the aperiodic task we run a polling server J_s with computation time $C_s = 3$ and period $T_s = 6$. Assume that the aperiodic task has computation time $C_a = 3$ and asks for the CPU at time $t = 3$. Plot the time evolution when a polling server is used together with the two tasks J_1 and J_2 scheduled as in Problem 7.5 part (a). Describe the scheduling activity illustrated in the plots.

8 Models of computation I: Discrete-event systems

EXERCISE 8.1

Consider the problem of controlling a gate which is lowered when a train is approaching and it is raised when the train has passed. We assume that the railway is unidirectional and that a train can be detected 1500m before the gate and 1000m after the gate. The sensors give binary outputs i.e., they give a '0' when the train is not over the sensor and a '1' when the train is over the sensor. The gate has a sensor which gives a binary information and in particular gives '0' if the gate is (fully) closed and '1' if the gate is (fully) opened. Figure 8.1 shows a schema of the system. The gate needs to be lowered as soon as a train is approaching, and raised when the train has passed. Model the system as a discrete-event system. Assume that trains, for safety reasons are distant from each other, so that no train approaches before the previous train has left.

EXERCISE 8.2

A vending machine dispenses soda for \$0.45. It accepts only dimes (\$0.10) and quarters (\$0.25). It does not give change in return if your money is not correct. The soda is dispensed only if the exact amount of money is inserted. Model the vending machine using a discrete-event system. Is it possible that the machine does not dispense soda? Prove it formally.

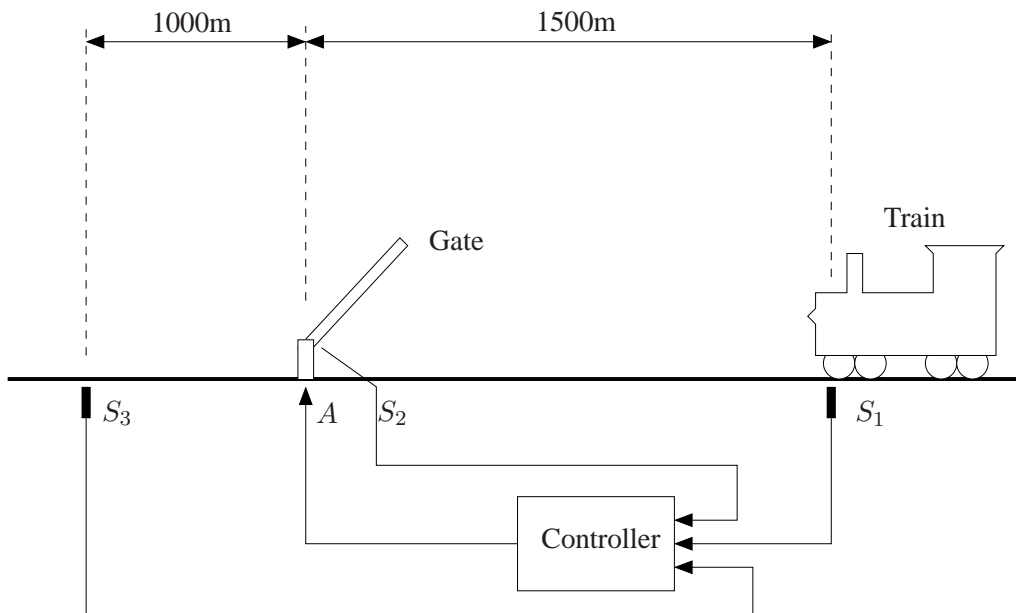


Figure 8.1: Control of a gate. Problem 8.1.

EXERCISE 8.3

Consider the automaton describing some discrete-event system shown in Figure 8.3. Describe formally the

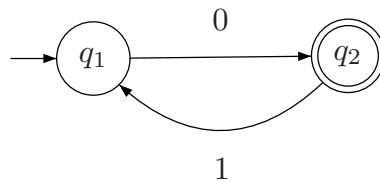


Figure 8.3: Automaton A of Problem 8.3.

DES. Compute the marked language L_m and the generated language L .

EXERCISE 8.4 (Example 2.13 in [5])

Consider the automaton A of Figure 8.4. Compute the language marked by the automaton A , $L_m(A)$ and the language generated by the automaton, $L(A)$.

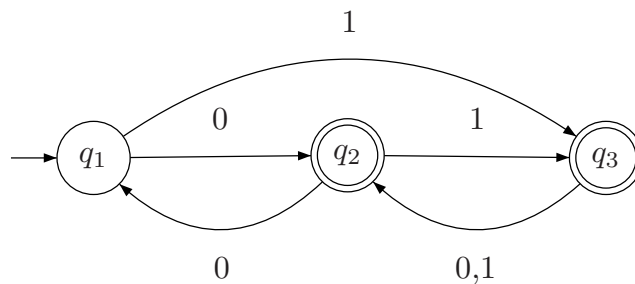


Figure 8.4: Automaton A of Problem 8.4.

EXERCISE 8.5 (Example 3.8 in [5])

Consider the automaton A of Figure 8.5. Determine the minimum state automaton.

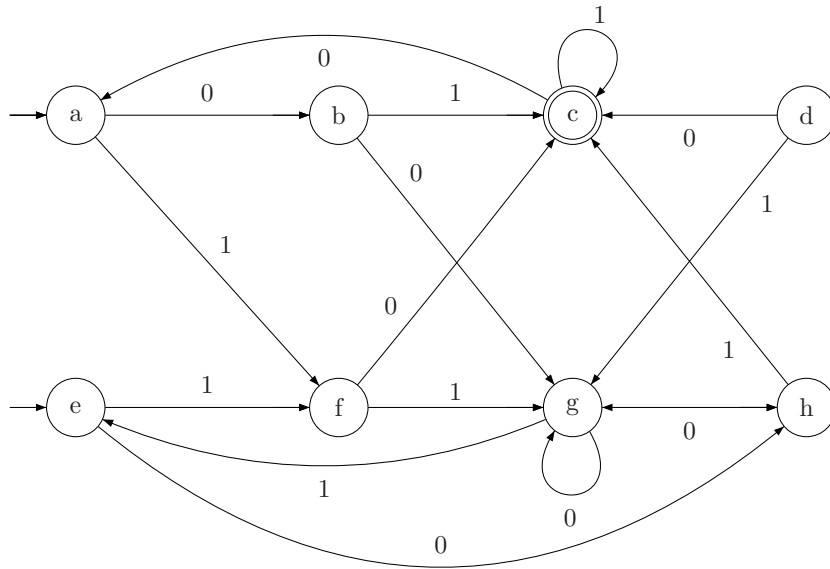


Figure 8.5: Automaton A of Problem 8.5.

EXERCISE 8.6 (Example 2.5 in [5])

Consider the automaton

$$A = (\{q_0, q_1\}, \{0, 1\}, \delta, q_0, \{q_1\})$$

be a nondeterministic automaton where

$$\delta(q_0, 0) = \{q_0, q_1\} \quad \delta(q_0, 1) = \{q_1\} \quad \delta(q_1, 0) = \delta(q_1, 1) = \{q_0, q_1\}.$$

Construct an deterministic automaton A' which accept the same L_m .

9 Models of computation II: Transition systems

EXERCISE 9.1

Consider a Discrete Event System described by an automaton and model it formally as a transition system.

EXERCISE 9.2

Try to model by a transition system the basic functionalities of the keypad of a mobile phone, including the states *mainmenu*, *contacts* and *lock*.

EXERCISE 9.3 [4]

Queuing systems arise in many application domains such as computer networks, manufacturing, logistics and transportation. A queuing system is composed of three basic elements: 1) the entities, generally referred to as *customers*, that do the waiting in their request for resources, 2) the resources for which the waiting is done, which are referred to as *servers*, and 3) the space where the waiting is done, which is defined as *queue*. Typical examples of servers are communication channels, which have a finite capacity to transmit information. In such a case, the customers are the unit of information and the queue is the amount of unit of information that is waiting to be transmitted over the channel.

A basic queue system is reported in figure 9.3. The circle represents a server, the open box is a queue preceding the server. The slots in the queue are waiting customers. The arrival rate of customers in the queue is denoted by a , whereas the departure rate of customers is denoted by b .

Model the queue system of figure 9.3 by a transition system. How many states has the system?

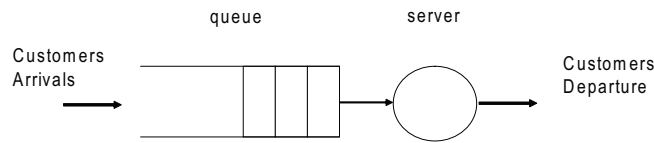


Figure 9.3: A basic queue system.

EXERCISE 9.4 [14]

Consider the transition system $T = \{S, \Sigma, \rightarrow, S_S\}$, where the cardinality of S is finite. The reachability algorithm is

```

Initialization : Reach1 = ∅;
                Reach0 = SS;
                i = 0;
Loop : While Reachi ≠ Reachi-1 do
        Reachi+1 = Reachi ∪ {s' ∈ S : ∃ s ∈ Reachi, σ ∈ Σ, s →σ s' ∈ →};
        i = i + 1;
    
```

Prove formally that

- the reachability algorithm finishes in a finite number of steps;
- upon exiting the algorithm, $Reach_i = Reach_T(S_S)$.

EXERCISE 9.5 [14]

Give the Transition System $T = \{S, \Sigma, \rightarrow, S_S\}$ reported in figure 9.5, describe the reach set when $S_S = \{3\}$ and $S_S = \{2\}$ by using the reachability algorithm.

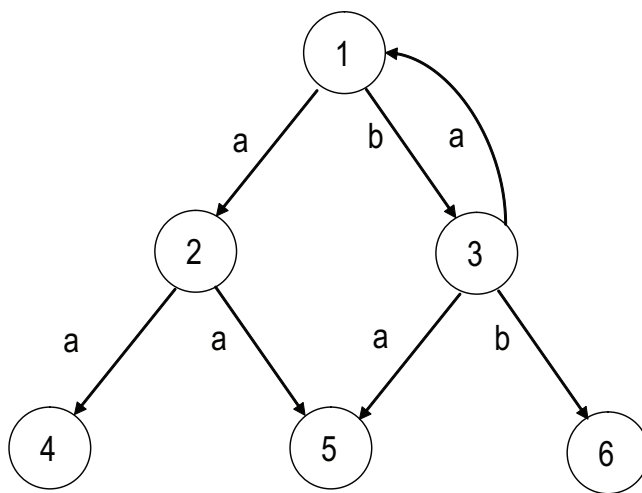


Figure 9.5: A Transition System.

Part III

Hybrid control

10 Modeling of hybrid systems

EXERCISE 10.1

A water level in a tank is controlled through a relay controller, which senses continuously the water level and turns a pump on or off. When the pump is off the water level decreases by 2 cm/s and when it is on, the water level increases by 1 cm/s. It takes 2 s for the control signal to reach the pump. It is required to keep the water level between 5 and 12 cm.

- Assuming that the controller starts the pump when the level reaches some threshold and turns it off when it reaches some other threshold, model the closed-loop system as a hybrid automaton.
- What thresholds should be used to fulfill the specifications?

EXERCISE 10.2

Consider the quantized control system in Figure 10.2. Such a system can be modeled as a hybrid automaton with continuous dynamics corresponding to $P(s)C(s)$ and discrete states corresponding to the levels of the quantizer. Suppose that each level of the quantizer can be encoded by a binary word of k bits. Then, how many discrete states N should the hybrid automaton have? Describe when discrete transitions in the hybrid automaton should take place.

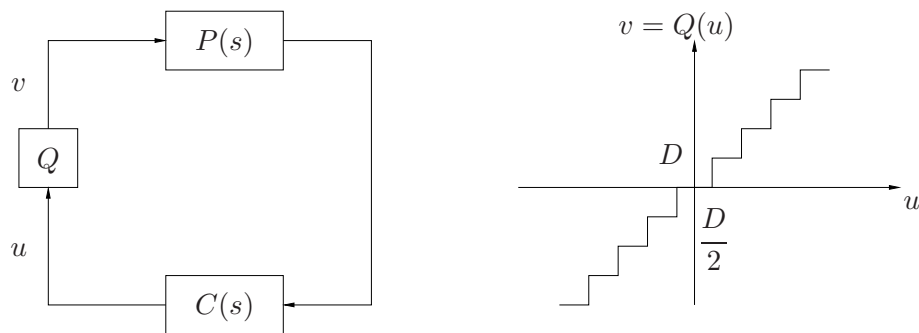


Figure 10.2: Quantized system in Problem 10.2.

EXERCISE 10.3

A system to cool a nuclear reactor is composed by two independently moving rods. Initially the coolant temperature x is 510 degrees and both rods are outside the reactor core. The temperature inside the reactor increases according to the following (linearized) system

$$\dot{x} = 0.1x - 50.$$

When the temperature reaches 550 degrees the reactor must be cooled down using the rods. Three things can happen

- the first rod is put into the reactor core
- the second rod is put into the reactor core
- none of the rods can be put into the reactor

For mechanical reasons a rod can be placed in the core if it has not been there for at least 20 seconds. If no rod is available the reactor should be shut down. The two rods can refrigerate the coolant according to the two following ODEs

$$\text{rod 1: } \dot{x} = 0.1x - 56$$

$$\text{rod 2: } \dot{x} = 0.1x - 60$$

When the temperature is decreased to 510 degrees the rods are removed from the reactor core. Model the system, including controller, as a hybrid system.

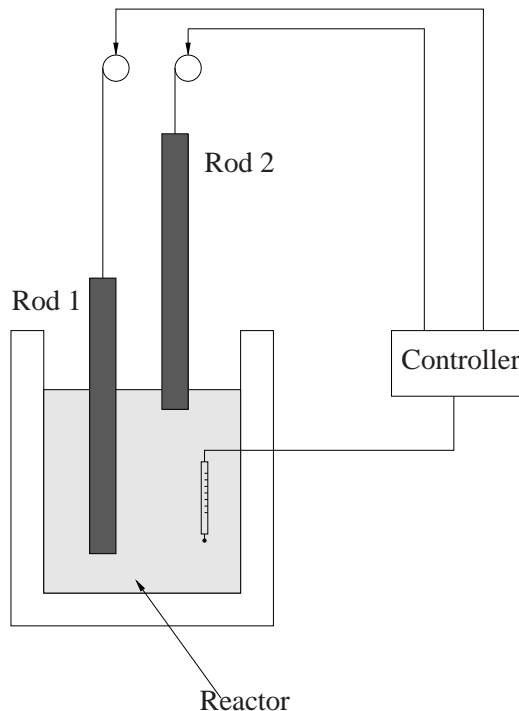


Figure 10.3: Nuclear reactor core with the two control rods

EXERCISE 10.4

Consider the classical sampled control system, shown in Figure 10.4. Model the system with a hybrid automaton. Suppose that the sampling period is k and that the hold circuit is a zero-order hold.

EXERCISE 10.5

Consider a hybrid system with two discrete states q_1 and q_2 . In state q_1 the dynamics are described by the linear system

$$\dot{x} = A_1 x = \begin{pmatrix} -1 & 0 \\ p & -1 \end{pmatrix}$$

and in state q_2 by

$$\dot{x} = A_2 x = \begin{pmatrix} -1 & p \\ 0 & -1 \end{pmatrix}.$$

Assume the system is

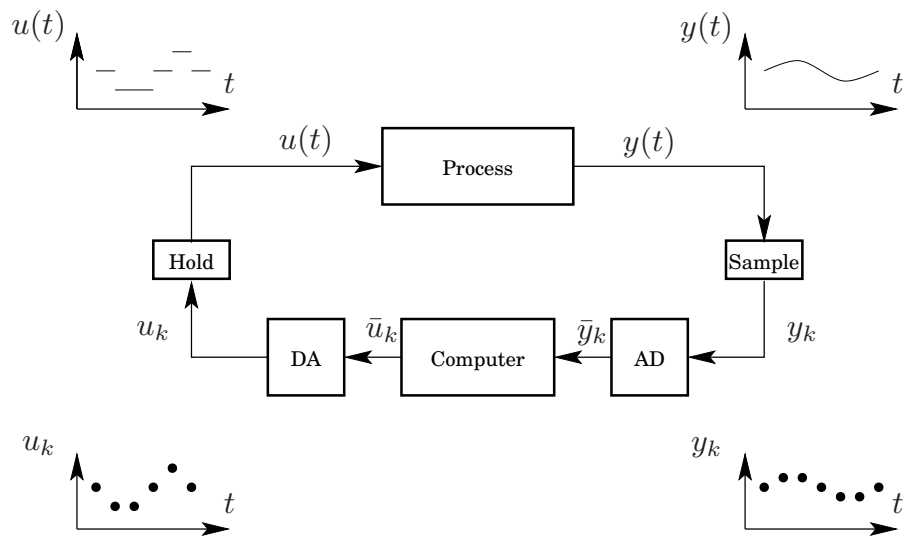


Figure 10.4: Sampled data control system of Problem 10.4.

in state q_1 if $2k \leq t < 2k + 1$ and

in state q_2 if $2k + 1 \leq t < 2k + 2$,

where $k = 0, 1, 2, \dots$

(a) Formally define a hybrid system with initial state q_1 , which operates in the way described above.

(b) Starting from $x(0) = x_0$, specify the evolution of the state $x(t)$ in the interval $t \in [0, 3)$ as a function of x_0 .

EXERCISE 10.6

Consider the hybrid system of Figure 10.6:

(a) Describe it as a hybrid automaton, $H = (Q, X, \text{Init}, f, D, E, G, R)$

(b) Find all the domains $D(q_3)$ so that the hybrid system is live?

(c) Plot the solution of the hybrid system.

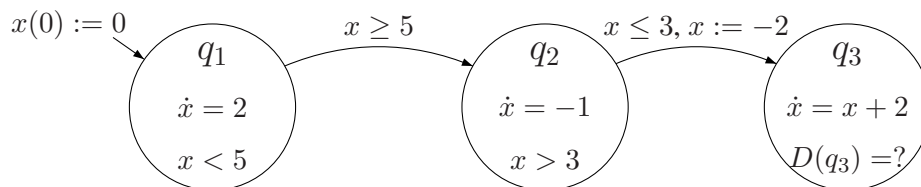


Figure 10.6: Hybrid system for Problem 10.6.

11 Stability of hybrid systems

EXERCISE 11.1

Consider three balls with unit mass, velocities v_1, v_2, v_3 , and suppose that they are touching at time $t = \tau_0$, see Figure 11.1. The initial velocity of Ball 1 is $v_1(\tau_0) = 1$ and Balls 2 and 3 are at rest, i.e., $v_2(\tau_0) = v_3(\tau_0) = 0$.

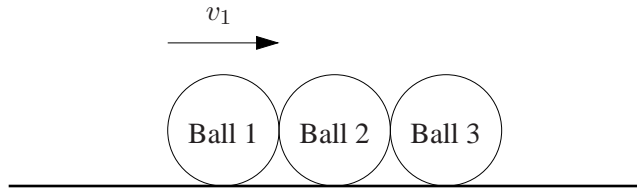


Figure 11.1: Three balls system. The Ball 1 has velocity v_1 at time $t = 0$.

Assume that the impact is a sequence of simple inelastic impacts occurring at $\tau'_0 = \tau'_1 = \tau'_2 = \dots$ (using notation from hybrid time trajectory). The first inelastic collision occurs at τ'_0 between balls 1 and 2, resulting in $v_1(\tau_1) = v_2(\tau_1) = 1/2$ and $v_3(\tau_1) = 0$. Since $v_2(\tau'_1) > v_3(\tau'_1)$, Ball 2 hits Ball 3 instantaneously giving $v_1(\tau_2) = 1/2$, and $v_2(\tau_2) = v_3(\tau_2) = 1/4$. Now $v_1(\tau'_2) > v_2(\tau'_2)$, so Ball 1 hits Ball 2 again resulting in a new inelastic collision. This leads to an infinite sequence of collisions.

- (a) Model the inelastic collisions of the three-ball system described above as a hybrid automaton $H = (Q, X, \text{Init}, f, D, E, G, R)$ with one discrete variable $Q = \{q\}$ and three continuous variables $X = \{v_1, v_2, v_3\}$.
- (b) Is the execution described above a Zeno execution? Motivate.
- (c) What is the accumulation point of the infinite series of hits described above? Make a physical interpretation.

EXERCISE 11.2

Consider the following system

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{pmatrix} -1 & 0 & 2 \\ 0 & -1 & 3 \\ -2 & -3 & -2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

Show, using a Lyapunov function, that the system is asymptotically stable.

EXERCISE 11.3

Consider the following theorem:

Theorem 2. A linear system

$$\dot{x} = Ax$$

is asymptotically stable if and only if for any positive definite symmetric matrix Q the equation

$$A^T P + PA = -Q$$

in the unknown $P \in \mathbb{R}^{n \times n}$ has a solution which is positive definite and symmetric.

Show the necessary part of the previous theorem (i.e., the *if* part).

EXERCISE 11.4

Consider the following system

$$\begin{aligned}\dot{x}_1 &= -x_1 + g(x_2) \\ \dot{x}_2 &= -x_2 + h(x_1)\end{aligned}$$

where the functions g and h are such that

$$|g(z)| \leq |z|/2 \quad |h(z)| \leq |z|/2$$

Show that the system is asymptotically stable.

EXERCISE 11.5

Consider the following discontinuous differential equations

$$\begin{aligned}\dot{x}_1 &= -\text{sgn}(x_1) + 2\text{sgn}(x_2) \\ \dot{x}_2 &= -2\text{sgn}(x_1) - \text{sgn}(x_2).\end{aligned}$$

where

$$\text{sgn}(z) = \begin{cases} +1 & \text{if } z \geq 0 \\ -1 & \text{if } z < 0. \end{cases}$$

Assume $x(0) \neq 0$,

- (a) define a hybrid automaton that models the discontinuous system
- (b) does the hybrid automaton exhibit Zeno executions for every initial state?

EXERCISE 11.6

Consider the following switching system

$$\dot{x} = a_q x, \quad a_q < 0 \quad \forall q$$

where $q \in \{1, 2\}$ and

$$\begin{aligned}\Omega_1 &= \{x \in \mathbb{R} \mid x \in [2k, 2k + 1), k = 0, 1, 2, \dots\} \\ \Omega_2 &= \{x \in \mathbb{R} \mid x \in [2k + 1, 2k + 2), k = 0, 1, 2, \dots\}\end{aligned}$$

Show that the system is asymptotically stable.

EXERCISE 11.7

Consider the following switching system

$$\dot{x} = A_q x$$

where $q \in \{1, 2\}$ and

$$A_1 = \begin{pmatrix} -1 & 0 \\ 0 & -2 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} -3 & 0 \\ 0 & -5 \end{pmatrix}.$$

Let Ω_q be such that

$$\Omega_1 = \{x \in \mathbb{R}^2 \mid x_1 \geq 0\}$$

$$\Omega_2 = \{x \in \mathbb{R}^2 \mid x_1 < 0\}$$

Show that the system is asymptotically stable.

EXERCISE 11.8

Consider the following switching system

$$\dot{x} = A_q x$$

where $q \in \{1, 2\}$ and

$$A_1 = \begin{pmatrix} -a_1 & b_1 \\ 0 & -c_1 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} -a_2 & b_2 \\ 0 & -c_2 \end{pmatrix}.$$

Assume that a_i, b_i and $c_i, i = 1, 2$ are real numbers and that $a_i, c_i > 0$. Show that the switched system is asymptotically stable.

EXERCISE 11.9

Consider a system that follows the dynamics

$$\dot{x} = A_1 x$$

for a time $\epsilon/2$ and then switches to the system

$$\dot{x} = A_2 x$$

for a time $\epsilon/2$. It then switches back to the first system, and so on.

- Model the system as a switched system
- Model the system as a hybrid automaton
- Let t_0 be a time instance at which the system begins a period in mode 1 (the first system) with initial condition x_0 . Determine the state at $t_0 + \epsilon/2$ and $t_0 + \epsilon$.
- Let ϵ tend to zero (very fast switching). Determine the solution the hybrid system will tend to.

EXERCISE 11.10

Consider the following hybrid system

$$\dot{x} = A_q x$$

where

$$A_1 = \begin{pmatrix} -1 & 0 \\ 0 & -2 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} -3 & 0 \\ 0 & -5 \end{pmatrix}.$$

Let Ω_q be such that

$$\Omega_1 = \{x \in \mathbb{R}^2 | x_1 \geq 0\}$$

$$\Omega_2 = \{x \in \mathbb{R}^2 | x_1 < 0\}$$

show that the switched system is asymptotically stable using a common Lyapunov function.

EXERCISE 11.11 (Example 2.1.5 page 18-19 in [12])

Consider the following switched system with $q \in \{1, 2\}$

$$\dot{x} = A_q x$$

where

$$A_1 = \begin{pmatrix} -1 & -1 \\ 1 & -1 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} -1 & -10 \\ 0.1 & -1 \end{pmatrix}.$$

- (a) Show that is impossible to find a quadratic common Lyapunov function.
- (b) Show that the origin is asymptotically stable for any switching sequence.

EXERCISE 11.12

Consider the following two-dimensional state-dependent switched system

$$\dot{x} = \begin{cases} A_1 x & \text{if } x_1 \leq 0 \\ A_2 x & \text{if } x_1 > 0 \end{cases}$$

where

$$A_1 = \begin{pmatrix} -5 & -4 \\ -1 & -2 \end{pmatrix} \quad \text{and} \quad A_2 = \begin{pmatrix} -2 & -4 \\ 20 & -2 \end{pmatrix}.$$

- (a) Prove that there is not a common quadratic Lyapunov function suitable to prove stability of the system
- (b) Prove that the switched system is asymptotically stable using the multiple Lyapunov approach.

12 Verification of hybrid systems

EXERCISE 12.1

Consider the following linear system

$$\dot{x} = \begin{pmatrix} -1 & 0 \\ 0 & -5 \end{pmatrix} x$$

Assume that the initial condition is defined in the following set

$$x_0 \in \{(x_1, x_2) \in \mathbb{R}^2 \mid x_1 = x_2, -10 \leq x_1 \leq 10\}$$

We want to verify that no trajectories enter in a *Bad* set defined as

$$Bad = \{(x_1, x_2) \in \mathbb{R}^2 \mid -8 \leq x_1 \leq 0 \wedge 2 \leq x_2 \leq 6\}.$$

EXERCISE 12.2

Consider the following linear system

$$\dot{x} = \begin{pmatrix} -5 & -5 \\ 0 & -1 \end{pmatrix} x.$$

Assume that the initial condition lies in the following set

$$x_0 \in \{(x_1, x_2) \in \mathbb{R}^2 \mid -2 \leq x_1 \leq 0 \wedge 2 \leq x_2 \leq 3\}.$$

Describe the system as a transition system and verify that no trajectories enter a *Bad* set defined as the triangle with vertices $v_1 = (-3, 2)$, $v_2 = (-3, -3)$ and $v_3 = (-1, 0)$.

EXERCISE 12.3

Consider the following controlled switched system

$$\begin{aligned} \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} &= \begin{pmatrix} 0 & 1 \\ \frac{1}{3} & 5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + B_1 u && \text{if } \|x\| < 1 \\ \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} &= \begin{pmatrix} 0 & 1 \\ \frac{1}{3} & -\frac{1}{3} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u && \text{if } 1 \leq \|x\| \leq 3 \\ \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} &= - \left(\begin{pmatrix} 0 & 1 \\ \frac{1}{3} & 5 \end{pmatrix} \begin{pmatrix} x_1 - 1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u \right) && \text{otherwise} \end{aligned}$$

Assume that the initial conditions are $x_0 \in \{x \in \mathbb{R}^2 \mid \|x\| > 3\}$,

- (a) Determine a control strategy such that $\text{Reach}_q \cup \Omega_1 \neq \emptyset$, i.e. Ω_1 can be reached from any initial condition when $B_1 = 0$.

Suppose in the following $B_1 = (0, 1)^T$,

- (b) Is it possible to determine a linear control input such that $(0, 0)$ is globally asymptotically stable?
- (c) Construct a piecewise linear system such that $(0, 0)$ is globally asymptotically stable.
- (d) Suppose now, that we do not want that the solution of the linear system would enter the *Bad* set Ω_1 . Determine a controller such that $\text{Reach}_q \cap \Omega_1 = \emptyset$.

EXERCISE 12.4

A system to cool a nuclear reactor is composed by two independently moving rods. Initially the coolant temperature x is 510 degrees and both rods are outside the reactor core. The temperature inside the reactor increases accordingly to the following (linearized) system

$$\dot{x} = 0.1x - 50.$$

When the temperature reaches 550 degrees the reactor must be cool down using the rods. Three things can happen

- the first rod is put into the reactor core
- the second rod is put into the reactor core
- none of the rods can be put into the reactor

For mechanical reasons the rods can be placed in the core if it has not been there for at least 20 seconds. The two rods can refrigerate the coolant accordingly to the two following ODEs

$$\text{rod 1: } \dot{x} = 0.1x - 56$$

$$\text{rod 2: } \dot{x} = 0.1x - 60$$

When the temperature is decreased to 510 degrees the rods are removed from the reactor core.

- Model the system as a hybrid system.
- If the temperature goes above 550 degrees, but there is no rod available to put down in the reactor, there will be a meltdown. Determine if this *Bad* state can be reached.

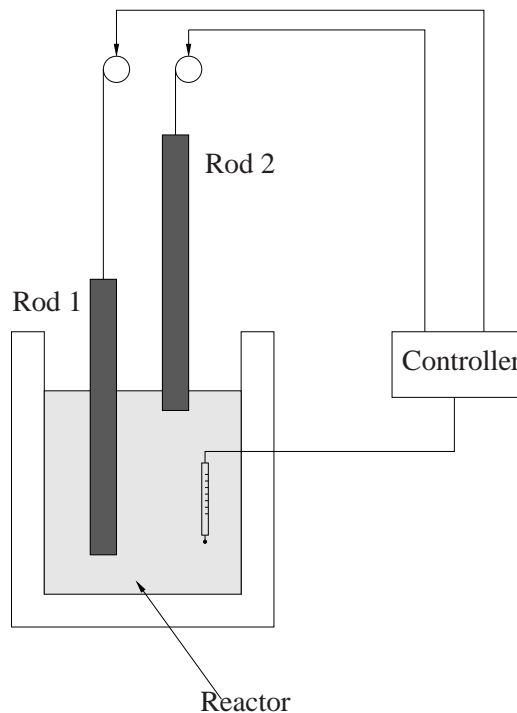


Figure 12.4: Nuclear reactor core with the two control rods

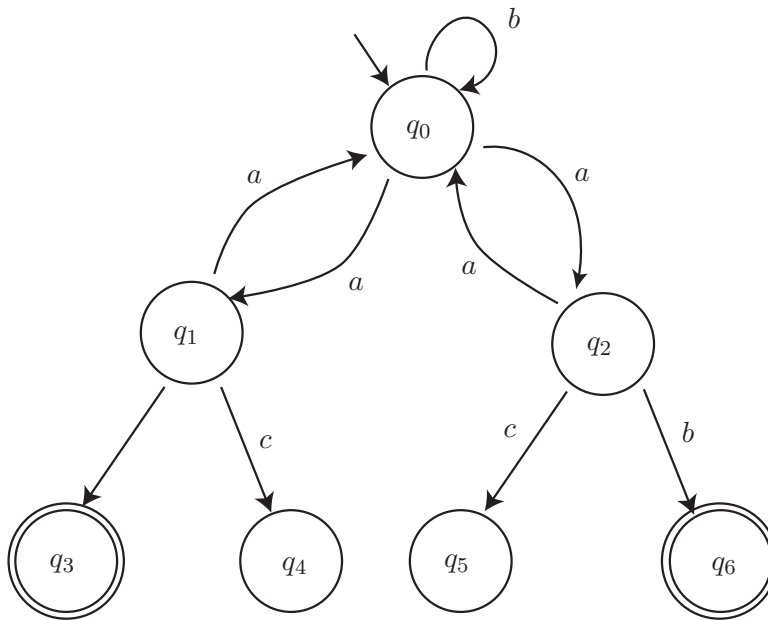


Figure 13.1: The transition system T .

13 Simulation and bisimulation

EXERCISE 13.1

Figure 13.1 shows a transition system $T = \{S, \Sigma, \rightarrow, S_0, S_F\}$, where

$$S = \{q_0, \dots, q_6\}$$

$$\Sigma = \{a, b, c\}$$

\rightarrow : According to the figure

$$S_0 = \{q_0\}$$

$$S_F = \{q_3, q_6\}.$$

Find the simplest quotient transition system \hat{T} that is bisimilar to T .

EXERCISE 13.2

Here we should insert a problem that the students do themselves: Show a system T and three candidates \hat{T}_i . Determine which systems are bisimilar to T .

Solutions

Part I

Time-triggered control

Solutions to review exercises

SOLUTION 1.1

Before solving the exercise we review some concepts on sampling and aliasing

Shannon sampling theorem

Let $x(t)$ be band-limited signal that is, $X(j\omega) = 0$ for $|\omega| > \omega_m$. Then $x(t)$ is uniquely determined by its samples $x(kh)$, $k = 0, \pm 1, \pm 2, \dots$ if

$$\omega_s > 2\omega_m$$

where $\omega_s = 2\pi/h$ is the sampling frequency, h the sampling period. The frequency $\omega_s/2$ is called the Nyquist frequency.

Reconstruction

Let $x(t)$ be the signal to be sampled. The sampled signal $x_s(t)$ is obtained multiplying the input signal $x(t)$ by a period impulse train signal $p(t)$, see Figure 1.1. We have that

$$\begin{aligned}x_s(t) &= x(t)p(t) \\ p(t) &= \sum_{k=-\infty}^{\infty} \delta(t - kh).\end{aligned}$$

Thus the sampled signal is

$$x_s(t) = \sum_{k=-\infty}^{\infty} x(kh)\delta(t - kh).$$

If we let the signal $x_s(t)$ pass through an ideal low-pass filter (see Figure 1.1.1) with impulse response

$$f(t) = \text{sinc}\left(\frac{\omega_s}{2}t\right)$$

and frequency response

$$F(j\omega) = \begin{cases} h, & -\omega_s/2 < \omega < \omega_s/2; \\ 0, & \text{otherwise.} \end{cases}$$

as shown in Figure 1.1.1. The output signal is

$$\begin{aligned}x_r(t) &= x_s(t) * f(t) = \int_{-\infty}^{\infty} x_s(t - \tau)f(\tau)d\tau \\ &= \int_{-\infty}^{\infty} \left(\sum_{k=-\infty}^{\infty} x(kh)\delta(t - \tau - kh) \right) f(\tau)d\tau \\ &= \sum_{k=-\infty}^{\infty} x(kh)\text{sinc}\left(\frac{\omega_s}{2}(t - kh)\right) \\ &= \sum_{k=-\infty}^{\infty} x(kh)\text{sinc}\left(\frac{\pi}{h}(t - kh)\right).\end{aligned}$$

Notice that perfect reconstruction requires an infinite number of samples.

Returning to the solution of the exercise we have that the Fourier transform of the sampled signal is given by

$$X_s(\omega) = \frac{1}{h} \sum_{k=-\infty}^{\infty} X(\omega + k\omega_s)$$

(a) The reconstructed signal is $x_r(t) = \cos(\omega_0 t)$ since $\omega_s = 6\omega_0 > 2\omega_0$.

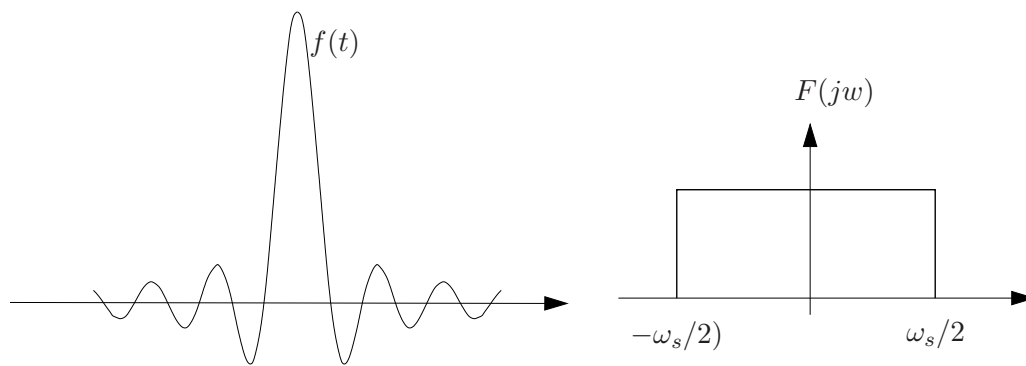


Figure 1.1.1: Impulse and frequency response of an ideal low-pass filter.

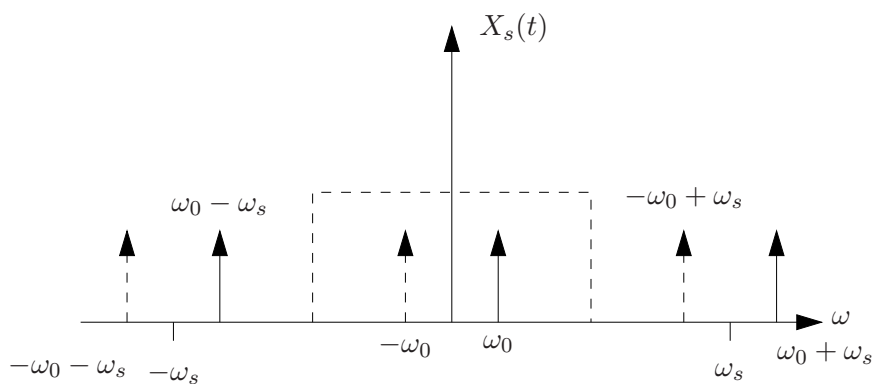


Figure 1.1.2: Frequency response of the signal with $\omega_0 = \omega_s/6$

(b) The reconstructed signal is $x_r(t) = \cos(\omega_0 t)$ since $\omega_s = 6\omega_0/2 > 2\omega_0$.

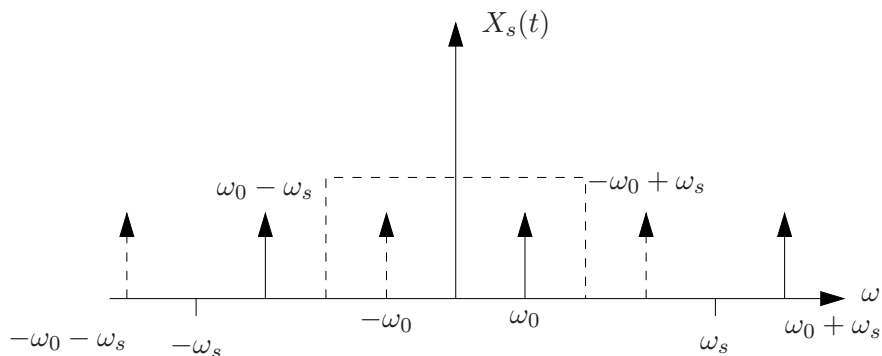


Figure 1.1.3: Frequency response of the signal with $\omega_0 = 6\omega_s/2$

(c) The reconstructed signal is $x_r(t) = \cos((-w_0 + \omega_s)t) = \cos(\omega_0/2t)$ since $\omega_s = 6\omega_0/4 < 2\omega_0$.

(d) The reconstructed signal is $x_r(t) = \cos((-w_0 + \omega_s)t) = \cos(\omega_0/5t)$ since $\omega_s = 6\omega_0/5 < 2\omega_0$.

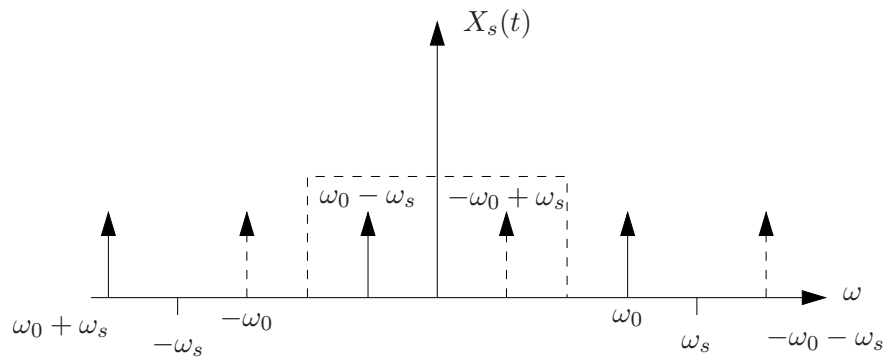


Figure 1.1.4: Frequency response of the signal with $\omega_0 = 6\omega_s/4$

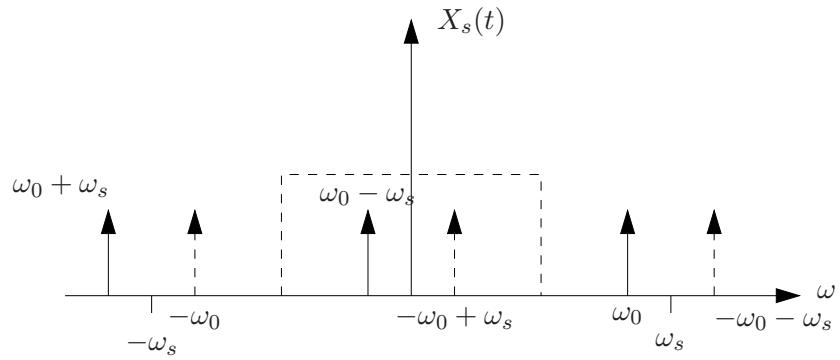


Figure 1.1.5: Frequency response of the signal with $\omega_0 = 6\omega_s/5$

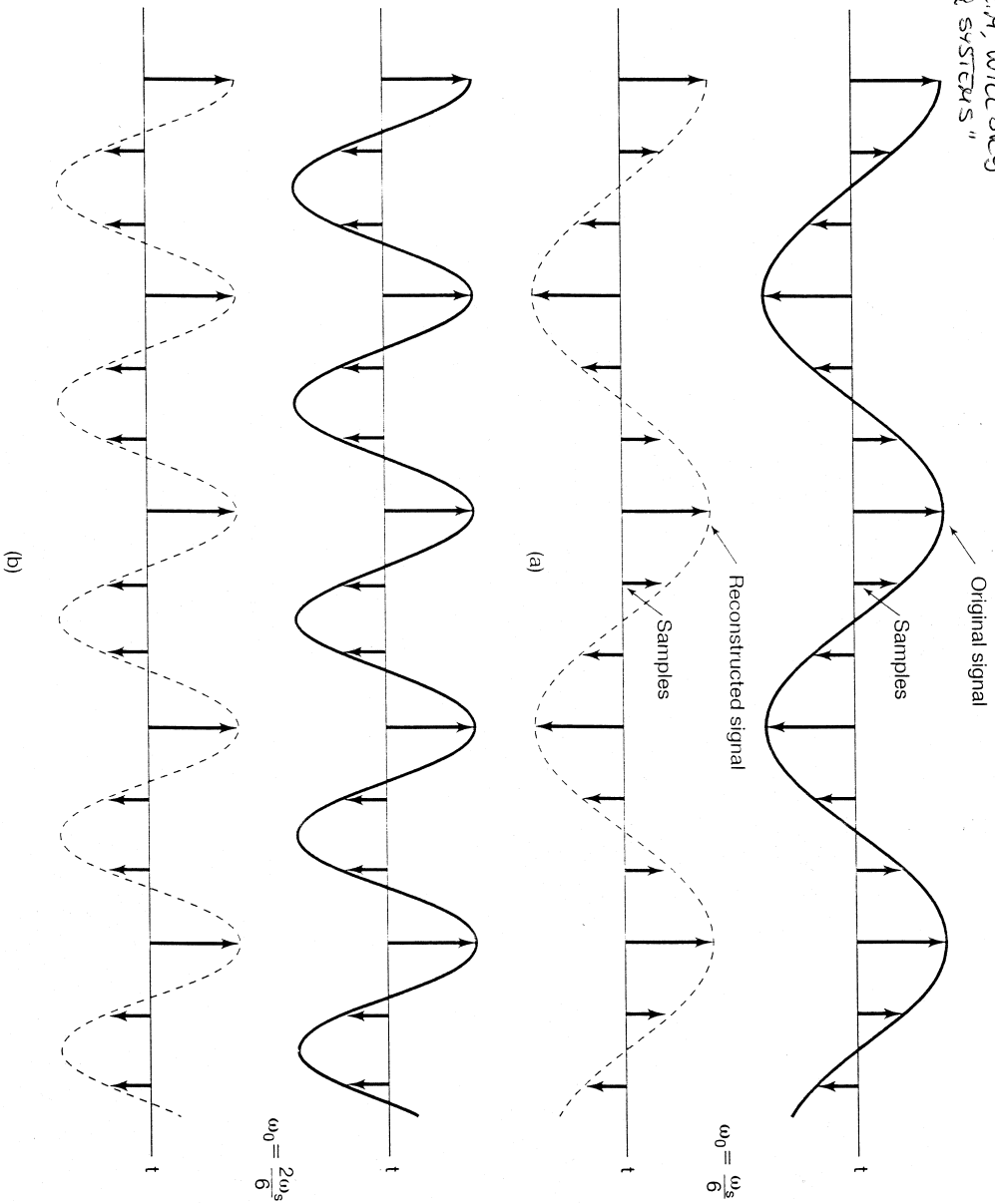


Figure 7.16 Effect of aliasing on a sinusoidal signal. For each of four values of ω_0 , the original sinusoidal signal (solid curve), its samples, and the reconstructed signal (dashed curve) are illustrated: (a) $\omega_0 = \omega_s/6$; (b) $\omega_0 = 2\omega_s/6$; (c) $\omega_0 = 4\omega_s/6$; (d) $\omega_0 = 5\omega_s/6$. In (a) and (b) no aliasing occurs, whereas in (c) and (d) there is aliasing.

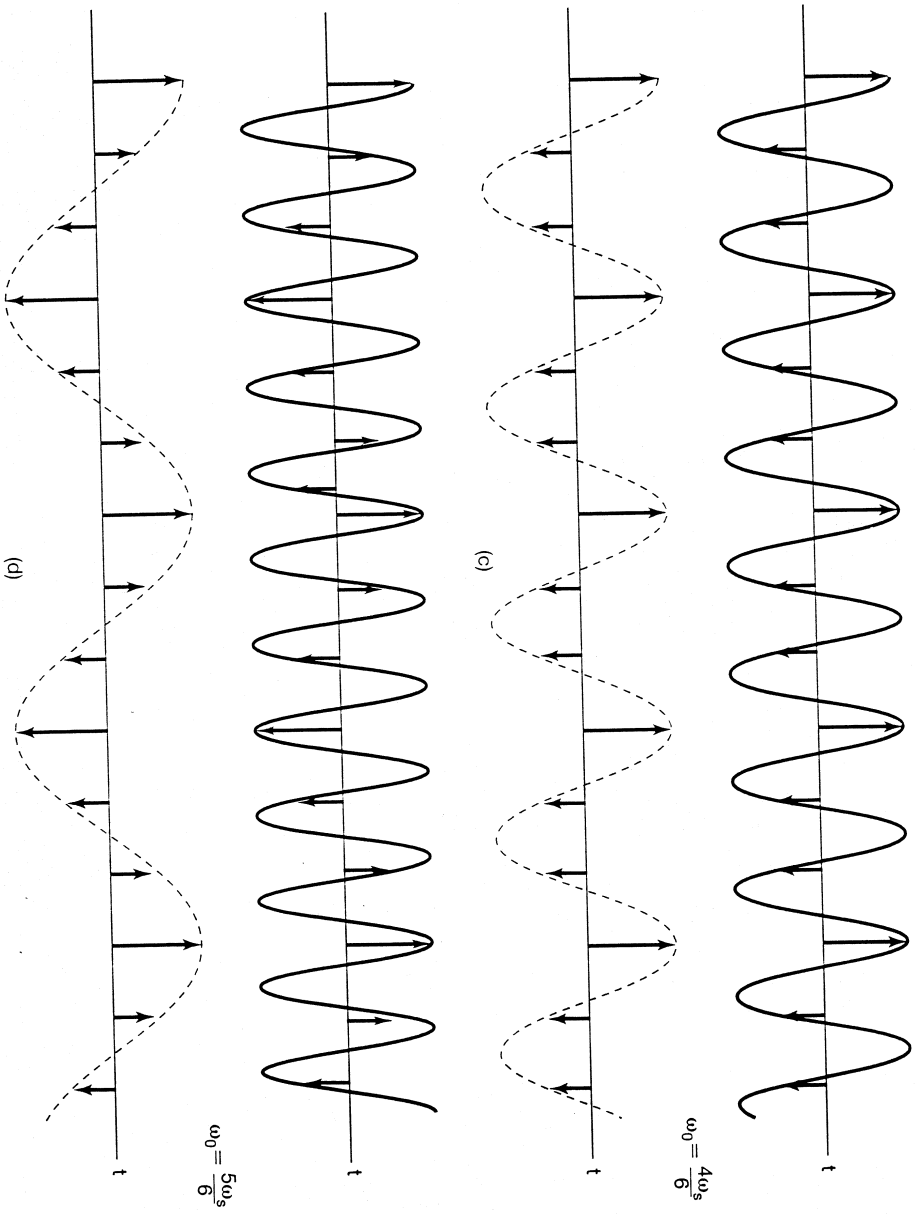


Figure 7.16 Continued

SOLUTION 1.2

We have that

$$e^{Ah} = \alpha_0 Ah + I\alpha_1.$$

The eigenvalues of Ah are $\pm ih$ thus we need to solve

$$\begin{aligned} e^{ih} &= \alpha_0 ih + \alpha_1 \\ e^{-ih} &= -\alpha_0 ih + \alpha_1. \end{aligned}$$

This gives

$$\begin{aligned} \alpha_0 &= \frac{e^{ih} - e^{-ih}}{2ih} = \frac{\sin h}{h} \\ \alpha_1 &= \frac{e^{ih} + e^{-ih}}{2} = \cos h. \end{aligned}$$

Thus

$$e^{Ah} = \frac{\sin h}{h} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} h + \cos h \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

We remind here some useful way of computing the matrix exponential of a matrix $A \in \mathbb{R}^{n \times n}$. Depending on the form of the matrix A we can compute the exponential in different ways

- If A is diagonal then

$$A = \begin{pmatrix} a_{11} & 0 & \dots & 0 \\ 0 & a_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & a_{nn} \end{pmatrix} \Rightarrow e^A = \begin{pmatrix} e^{a_{11}} & 0 & \dots & 0 \\ 0 & e^{a_{22}} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & e^{a_{nn}} \end{pmatrix}$$

- A is nilpotent of order m . Then $A^m = 0$ and $A^{m+i} = 0$ for $i = 1, 2, \dots$. Then it is possible to use the following series expansion to calculate the exponential

$$e^A = I + A + \frac{A^2}{2!} + \dots + \frac{A^{m-1}}{(m-1)!}$$

- Using the inverse Laplace transform we have

$$e^{At} = \mathcal{L}^{-1}((sI - A)^{-1})$$

- In general it is possible to compute the exponential of a matrix (or any continuous matrix function $f(A)$) using the Cayley-Hamilton Theorem. For every function f there is a polynomial p of degree less than n such that

$$f(A) = p(A) = \alpha_0 A^{n-1} + \alpha_1 A^{n-2} + \dots + \alpha_{n-1} I.$$

If the matrix A has distinct eigenvalues, the n coefficient $\alpha_0, \dots, \alpha_{n-1}$ are computed solving the system of n equations

$$f(\lambda_i) = p(\lambda_i) \quad i = 1, \dots, n.$$

If there is a multiple eigenvalue with multiplicity m , then the additional conditions

$$\begin{aligned} f^{(1)}(\lambda_i) &= p^{(i)}(\lambda_i) \\ &\vdots \\ f^{(m-1)}(\lambda_i) &= p^{(m-1)}(\lambda_i) \end{aligned}$$

hold, where $f^{(i)}$ is the i th derivative with respect to λ .

SOLUTION 1.3

We recall here what is the z -transform of a signal. Consider a *discrete-time signal* $x(kh)$, $k = 0, 1, \dots$. The z -transform of $x(kh)$ is defined as

$$\mathcal{Z}\{x(kh)\} = X(z) = \sum_{k=0}^{\infty} x(kh)z^{-k}$$

where z is a complex variable.

Using the definition

$$X(z) = \sum_{k=0}^{\infty} e^{-kh/T} z^{-k} = \sum_{k=0}^{\infty} \left\{ e^{-h/T} z^{-1} \right\}^k.$$

If $|e^{-h/T} z^{-1}| < 1$ then

$$X(z) = \frac{1}{1 - z^{-1}e^{-h/T}} = \frac{z}{z - e^{-h/T}}.$$

SOLUTION 1.4

Using the definition

$$X(z) = \sum_{k=0}^{\infty} \sin(wh)z^{-k}.$$

Since

$$\sin wh = \frac{e^{jwh} - e^{-jwh}}{2i}$$

then

$$X(z) = \frac{1}{2i} \sum_{k=0}^{\infty} \left\{ e^{iwh} z^{-1} \right\}^k - \frac{1}{2i} \sum_{k=0}^{\infty} \left\{ e^{-iwh} z^{-1} \right\}^k$$

If $|e^{\pm iwh} z^{-1}| < 1$ then

$$X(z) = \frac{1}{2i} \left(\frac{z}{z - e^{iwh}} - \frac{z}{z - e^{-iwh}} \right) = \dots = \frac{z \sin wh}{z^2 - 2z \cos wh + 1}.$$

SOLUTION 1.5

For a discrete-time signal $x(k)$, we have the following

$$\begin{aligned} \mathcal{Z}(x_k) &= X(z) = \sum_{k=0}^{\infty} x(k)z^{-k} = x(0) + x(1)z^{-1} + \dots \\ \mathcal{Z}(x_{k+1}) &= x(1) + x(2)z^{-1} + \dots \\ &= z x(0) - z x(0) + z (x(1)z^{-1} + x(2)z^{-2} + \dots) \\ &= z (x(0) + x(1)z^{-1} + x(2)z^{-2} + \dots) - z x(0) \\ &= z X(z) - z x(0) \end{aligned}$$

similarly,

$$\mathcal{Z}(x_{k+2}) = z^2 X(z) - z^2 x(0) - z x(1)$$

The discrete step is the following function

$$u(k) = \begin{cases} 0, & \text{if } k < 0; \\ 1, & \text{if } k \geq 0. \end{cases}$$

Thus

$$U(z) = \frac{z}{z-1}.$$

Using the previous z -transform we have

$$z^2 Y(z) - z^2 y(0) - z y(1) - 1.5z Y(z) + 1.5z y(0) + 0.5 Y(z) = z U(z) - z u(0).$$

Collecting $Y(z)$ and substituting the initial conditions we get

$$Y(z) = \frac{0.5z^2 - 0.5z}{z^2 - 1.5z + 0.5} + \frac{z}{z^2 - 1.5z + 0.5} U(z)$$

Since $U(z) = z/(z-1)$ then

$$Y(z) = \frac{0.5z}{z-0.5} + \frac{z^2}{(z-1)^2(z-0.5)}.$$

Inverse transform gives

$$\begin{aligned} y(k) &= 0.5^{k+1} + \left(\frac{0.5(k+1) - 1}{0.5^2} + \frac{0.5^{k+1}}{0.5^2} \right) u(k-1) \\ &= 0.5^{k+1} + \left(\frac{k-1}{0.5} + 0.5^{k-1} \right) u(k-1) \end{aligned}$$

Solutions to models of sampled systems

SOLUTION 2.1

The sampled system is given by

$$\begin{aligned} x(kh+h) &= \Phi x(kh) + \Gamma u(kh) \\ y(kh) &= Cx(kh) \end{aligned}$$

where

$$\begin{aligned} \Phi &= e^{-ah} \\ \Gamma &= \int_0^h e^{-as} ds b = \frac{b}{a} (1 - e^{-ah}). \end{aligned}$$

Thus the sampled system is

$$\begin{aligned} x(kh+h) &= e^{-ah} x(kh) + \frac{b}{a} (1 - e^{-ah}) u(kh) \\ y(kh) &= cx(kh). \end{aligned}$$

The poles of the sampled system are the eigenvalues of Φ . Thus there is a real pole at e^{-ah} . If h is small $e^{-ah} \approx 1$. If $a > 0$ then the pole moves towards the origin as h increases, if $a < 0$ it moves along the positive real axis, as shown in Figure 2.1.1.

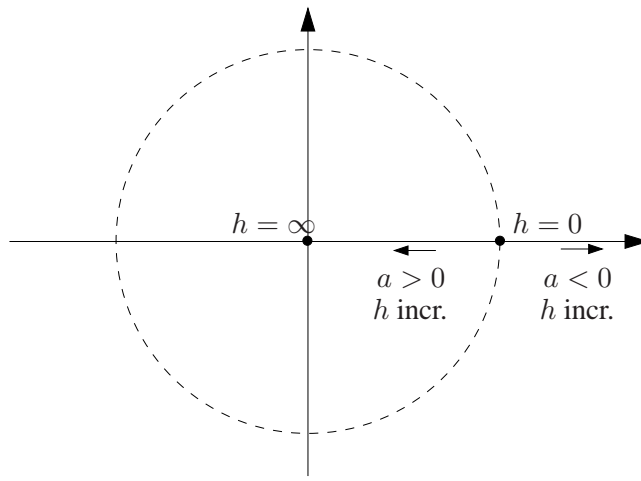


Figure 2.1.1: Closed-loop system for Problem 2.1.

SOLUTION 2.2

(a) The transfer function can be written as

$$G(s) = \frac{1}{(s+1)(s+2)} = \frac{\alpha}{s+1} + \frac{\beta}{s+2} = \frac{1}{s+1} - \frac{1}{s+2}.$$

A state-space representation (in diagonal form) is then

$$\begin{aligned} \dot{x} &= \underbrace{\begin{pmatrix} -1 & 0 \\ 0 & -2 \end{pmatrix}}_A x + \underbrace{\begin{pmatrix} 1 \\ 1 \end{pmatrix}}_B u \\ y &= \underbrace{\begin{pmatrix} 1 & -1 \end{pmatrix}}_C x. \end{aligned}$$

The state-space representation of the sampled system is

$$\begin{aligned} x(k+1) &= \Phi x(k) + \Gamma u(k) \\ y(k) &= Cx(k) \end{aligned}$$

where

$$\begin{aligned} \Phi &= e^{Ah} = \begin{pmatrix} e^{-1} & 0 \\ 0 & e^{-2} \end{pmatrix} \\ \Gamma &= \int_0^h e^{As} ds B = \int_0^1 \begin{pmatrix} e^{-s} \\ e^{-2s} \end{pmatrix} ds = \begin{pmatrix} 1 - e^{-1} \\ \frac{1 - e^{-2}}{2} \end{pmatrix} \end{aligned}$$

since A is diagonal.

(b) The pulse-transfer function is given by

$$\begin{aligned} H(z) &= C(zI - \Phi)^{-1}\Gamma = \begin{pmatrix} 1 & -1 \end{pmatrix} \begin{pmatrix} \frac{1}{z-e^{-1}} & 0 \\ 0 & \frac{1}{z-e^{-2}} \end{pmatrix} \begin{pmatrix} 1 - e^{-1} \\ \frac{1 - e^{-2}}{2} \end{pmatrix} \\ &= \frac{z(3/2 + e^{-1} - 1/2e^{-2}) + (3/2e^{-3} - e^{-2} - 1/2e^{-1})}{(z - e^{-1})(z - e^{-2})} \end{aligned}$$

SOLUTION 2.3

(a) The sampled system is

$$\begin{aligned}x(kh + h) &= \Phi x(kh) + \Gamma u(kh) \\y(kh) &= Cx(kh)\end{aligned}$$

where

$$\Phi = e^{Ah} \qquad \Gamma = \int_0^h e^{As} B ds.$$

To compute e^{Ah} we use the fact that

$$e^{Ah} = \mathcal{L}^{-1}((sI - A)^{-1}) = \mathcal{L}^{-1}\left(\frac{1}{s^2 + 1} \begin{pmatrix} s & 1 \\ -1 & s \end{pmatrix}\right).$$

Since

$$\begin{aligned}\mathcal{L}^{-1}\left(\frac{s}{s^2 + 1}\right) &= \cos h \\ \mathcal{L}^{-1}\left(\frac{1}{s^2 + 1}\right) &= \sin h\end{aligned}$$

then

$$e^{Ah} = \begin{pmatrix} \cos h & \sin h \\ -\sin h & \cos h \end{pmatrix}.$$

Equivalently we can compute e^{Ah} using Cayley-Hamilton's theorem. The matrix e^{Ah} can be written as

$$e^{Ah} = a_0 Ah + a_1 I$$

where the constants a_0 and a_1 are computed solving the characteristic equation

$$e^{\lambda_k} = a_0 \lambda_k + a_1 \quad k = 1, \dots, n$$

where n is the dimension of the matrix A and λ_k are distinct eigenvalues of the matrix Ah . In this example the eigenvalues of Ah are $\pm hi$. Thus we need to solve the following system of equations

$$\begin{aligned}e^{ih} &= a_0 ih + a_1 \\ e^{-ih} &= -a_0 ih + a_1\end{aligned}$$

which gives

$$\begin{aligned}a_0 &= \frac{1}{2hi} (e^{ih} - e^{-ih}) = \frac{\sin h}{h} \\ a_1 &= \frac{1}{2} (e^{ih} + e^{-ih}) = \cos h.\end{aligned}$$

Finally we have

$$e^{Ah} = \frac{\sin h}{h} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} h + \cos h \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos h & \sin h \\ -\sin h & \cos h \end{pmatrix}.$$

(b) Using Laplace transform we obtain

$$s^2Y(s) + 3sY(s) + Y(s) = sU(s) + 3U(s).$$

Thus the system has the transfer function

$$G(s) = \frac{s+3}{s^2+3s+2} = \frac{2}{s+1} - \frac{1}{s+2}.$$

One state-space realization of the system with transfer function $G(s)$ is

$$\begin{aligned} \dot{x} &= \underbrace{\begin{pmatrix} -1 & 0 \\ 0 & -2 \end{pmatrix}}_A x + \underbrace{\begin{pmatrix} 1 \\ 1 \end{pmatrix}}_B u \\ y &= \underbrace{\begin{pmatrix} 2 & -1 \end{pmatrix}}_C x. \end{aligned}$$

Thus the sampled system is

$$\begin{aligned} x(kh+h) &= \Phi x(kh) + \Gamma u(kh) \\ y(kh) &= Cx(kh) \end{aligned}$$

with

$$\begin{aligned} \Phi &= e^{Ah} = \begin{pmatrix} e^{-h} & 0 \\ 0 & e^{-2h} \end{pmatrix} \\ \Gamma &= \int_0^h e^{As} B ds = \int_0^h \begin{pmatrix} e^{-s} \\ e^{-2s} \end{pmatrix} ds = \begin{pmatrix} 1 - e^{-h} \\ \frac{1 - e^{-2h}}{2} \end{pmatrix} \end{aligned}$$

(c) One state-space realization of the system is

$$\begin{aligned} \dot{x} &= \underbrace{\begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}}_A x + \underbrace{\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}}_B u \\ y &= \underbrace{\begin{pmatrix} 0 & 0 & 1 \end{pmatrix}}_C x. \end{aligned}$$

We need to compute Φ and Γ . In this case we can use the series expansion of e^{Ah}

$$e^{Ah} = I + Ah + \frac{A^2 h^2}{2} + \dots$$

since $A^3 = 0$, and thus all the successive powers of A . Thus in this case

$$\begin{aligned} \Phi &= e^{Ah} = \begin{pmatrix} 1 & 0 & 0 \\ h & 1 & 0 \\ h^2/2 & h & 1 \end{pmatrix} \\ \Gamma &= \int_0^h e^{As} B ds = \int_0^h (1 \quad s \quad s^2/2)^T ds = (h \quad h^2/2 \quad h^3/6)^T \end{aligned}$$

SOLUTION 2.4

We will use in this exercise the following relation

$$\Phi = e^{Ah} \Rightarrow A = \frac{\ln \Phi}{h}$$

(a)

$$y(kh) - 0.5y(kh - h) = 6u(kh - h) \Rightarrow y(kh) - 0.5q^{-1}y(kh) = 6q^{-1}u(kh)$$

which can be transformed in state-space as

$$\begin{aligned} x(kh + h) &= \Phi x(kh) + \Gamma u(kh) = 0.5x(kh) + 6u(kh) \\ y(kh) &= x(kh). \end{aligned}$$

The continuous time system is then

$$\begin{aligned} \dot{x}(t) &= ax(t) + bu(t) \\ y(t) &= x(t), \end{aligned}$$

where, in this case since Φ and Γ are scalars, we have

$$\begin{aligned} a &= \frac{\ln \Phi}{h} = -\frac{\ln 2}{h} \\ b &= \Gamma / \int_0^h e^{as} ds = \frac{12 \ln 2}{h} \end{aligned}$$

(b)

$$\begin{aligned} x(kh + h) &= \underbrace{\begin{pmatrix} -0.5 & 1 \\ 0 & -0.3 \end{pmatrix}}_{\Phi} x(kh) + \begin{pmatrix} 0.5 \\ 0.7 \end{pmatrix} u(kh) \\ y(kh) &= (1 \quad 1) x(kh). \end{aligned}$$

We compute the eigenvalues of Φ

$$\begin{aligned} \det(sI - \Phi) &= \begin{vmatrix} s + 0.5 & -1 \\ 0 & s + 0.3 \end{vmatrix} = 0 \Leftrightarrow (s + 0.5)(s + 0.3) = 0 \\ \lambda_1 &= -0.5, \quad \lambda_2 = -0.3. \end{aligned}$$

Both eigenvalues of Φ are on the negative real axis, thus no corresponding continuous system exists.

(c) We can proceed as in (a). In this case $\Phi = -0.5$ which means that the sampled system has a pole on the negative real axis. Thus, as in (b), no corresponding continuous system exists.

SOLUTION 2.5

(Ex. 2.11 in [2])

(a) A state space representation of the transfer function is

$$\begin{aligned} \dot{x} &= \begin{pmatrix} 0 & 0 \\ 0 & -1 \end{pmatrix} x + \begin{pmatrix} 1 \\ 1 \end{pmatrix} u \\ y &= (1 \quad -1) x. \end{aligned}$$

In this case

$$\Phi = e^{Ah} = \begin{pmatrix} 1 & 0 \\ 0 & e^{-h} \end{pmatrix}$$

$$\Gamma = \int_0^h e^{As} B ds = \begin{pmatrix} h \\ 1 - e^{-h} \end{pmatrix}$$

(b) The pulse-transfer function is given by

$$H(z) = C(zI - \Phi)^{-1}\Gamma = (1 \quad -1) \begin{pmatrix} z-1 & 0 \\ 0 & z-e^{-h} \end{pmatrix}^{-1} \begin{pmatrix} h \\ 1-e^{-h} \end{pmatrix}$$

$$= \frac{(h + e^{-h} - 1)z + (1 - e^{-h} - he^{-h})}{(z-1)(z-e^{-h})}$$

(c) The pulse response is

$$h(k) = \begin{cases} 0, & k = 0; \\ C\Phi^{k-1}\Gamma, & k \geq 1. \end{cases}$$

Since

$$\Phi^k = (e^{Ah})^k = e^{Akh}$$

then we have

$$h(k) = C\Phi^{k-1}\Gamma = (1 \quad -1) \begin{pmatrix} 1 & 0 \\ 0 & e^{-(k-1)h} \end{pmatrix} \begin{pmatrix} h \\ 1 - e^{-h} \end{pmatrix}$$

$$= h - e^{-(kh-h)} + e^{-hk}$$

(d) A difference equation relating input and output is obtained from $H(q)$.

$$y(kh) = H(q)u(kh) = \frac{(h + e^{-h} - 1)q + (1 - e^{-h} - he^{-h})}{(q-1)(q-e^{-h})}u(kh)$$

which gives

$$y(kh + 2h) - (1 + e^{-h})y(kh + h) + e^{-h}y(kh) = (h + e^{-h} - 1)u(kh + h) + (1 - e^{-h} - he^{-h})u(kh)$$

(e) The poles are in $z = 1$ and $z = e^{-h}$. The second pole moves from 1 to 0 as h goes from 0 to ∞ . There is a zero in

$$z = -\frac{1 - e^{-h} - he^{-h}}{h + e^{-h} - 1}.$$

The zero moves from -1 to 0 as h increases, see Figure 2.5.1

SOLUTION 2.6

(a) We notice that $\tau < h$. The continuous-time system in state-space is

$$\dot{x} = \underbrace{0}_A \cdot x + \underbrace{1}_B \cdot u(t - \tau)$$

$$y = \underbrace{1}_C \cdot x$$

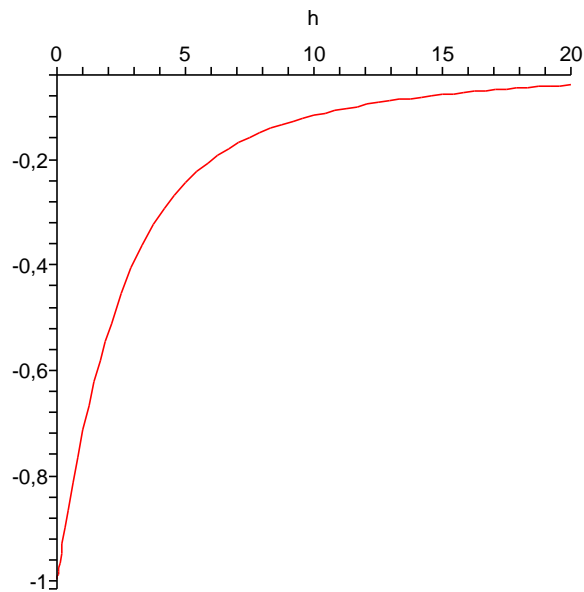


Figure 2.5.1: The zero of Problem 2.5 as function of h .

Sampling the continuous-time system with sampling period $h = 1$ we get

$$\begin{aligned} x(k+1) &= \Phi x(k) + \Gamma_0 u(k) + \Gamma_1 u(k-1) \\ y(k) &= x(k), \end{aligned}$$

where

$$\begin{aligned} \Phi &= e^{Ah} = e^0 = 1 \\ \Gamma_0 &= \int_0^{h-\tau} e^{As} ds B = 0.5 \\ \Gamma_1 &= e^{A(h-\tau)} \int_0^{\tau} e^{As} ds B = 0.5. \end{aligned}$$

The system in state space is

$$\begin{aligned} \begin{pmatrix} x(k+1) \\ u(k) \end{pmatrix} &= \begin{pmatrix} 1 & 0.5 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x(k) \\ u(k-1) \end{pmatrix} + \begin{pmatrix} 0.5 \\ 1 \end{pmatrix} u(k) \\ y(k) &= (1 \quad 0) \begin{pmatrix} x(k) \\ u(k-1) \end{pmatrix}. \end{aligned}$$

The system is of second order.

(b) The pulse-transfer function is

$$H(z) = C(zI - \Phi)^{-1}\Gamma = \dots = \frac{0.5(z+1)}{z(z-1)}.$$

To determine the pulse-response we inverse-transform $H(z)$.

$$H(z) = \frac{0.5(z+1)}{z(z-1)} = 0.5 \left(z^{-1} \frac{z}{z-1} + z^{-2} \frac{z}{z-1} \right).$$

The inverse transform of $z/(z-1)$ is a step. Thus we have the sum of two steps delayed of 1 and 2 time-steps, thus the pulse-response is

$$h(kh) = \begin{cases} 0, & k = 0; \\ 0.5, & k = 1; \\ 1, & k > 1. \end{cases}$$

(c) We can consider $H(z)$ computed in (b). There are two poles, one in $z = 0$ and another in $z = 1$. There is a zero in $z = -1$.

SOLUTION 2.7

In this case the time delay is longer than the sampling period, $\tau > h$. The sampled system with $h = 1$ is

$$\begin{aligned} x(k+1) &= \Phi x(k) + \Gamma_0 u(k - (d-1)) + \Gamma_1 u(k-d) \\ y(k) &= x(k), \end{aligned}$$

where we compute d as the integer such that

$$\tau = (d-1)h + \tau', \quad 0 < \tau' \leq h$$

and where Γ_0 and Γ_1 are computed as in the solution of exercise 2.6, where τ is replaced by τ' . In this example $d = 2$ and $\tau' = 0.5$, and where

$$\begin{aligned} \Phi &= e^{-1} \\ \Gamma_0 &= 1 - e^{-0.5} \\ \Gamma_1 &= e^{-0.5} - e^{-1}. \end{aligned}$$

A state representation is then

$$\begin{aligned} \begin{pmatrix} x(k+1) \\ u(k-1) \\ u(k) \end{pmatrix} &= \begin{pmatrix} \Phi & \Gamma_0 & \Gamma_1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x(k) \\ u(k-2) \\ u(k-1) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u(k) \\ y(k) &= (1 \quad 0 \quad 0) \begin{pmatrix} x(k) \\ u(k-2) \\ u(k-1) \end{pmatrix}. \end{aligned}$$

We still have a finite dimensional system (third order).

SOLUTION 2.8

The system

$$y(k) - 0.5y(k-1) = u(k-9) + 0.2u(k-10)$$

can be shifted in time so that

$$y(k+10) - 0.5y(k+9) = u(k+1) + 0.2u(k)$$

which can be written as

$$(q^{10} - 0.5q^9)y(k) = (q + 0.2)u(k).$$

Thus

$$\begin{aligned} A(q) &= q^{10} - 0.5q^9 \\ B(q) &= q + 0.2 \end{aligned}$$

and the system order is $\deg A(q) = 10$. We can rewrite the given system as

$$(1 - 0.5q^{-1})y(k) = (1 + 0.2q^{-1})u(k - 9).$$

where

$$\begin{aligned} A^*(q^{-1}) &= 1 - 0.5q^{-1} \\ B^*(q^{-1}) &= 1 + 0.2q^{-1} \end{aligned}$$

with $d = 9$. Notice that

$$\frac{B(q)}{A(q)} = \frac{q + 0.2}{q^{10} - 0.5q^9} = q^{-9} \frac{1 + 0.2q^{-1}}{1 - 0.5q^{-1}} = \frac{B^*(q^{-1})}{A^*(q^{-1})}$$

SOLUTION 2.9

We can rewrite the system

$$y(k + 2) - 1.5y(k + 1) + 0.5y(k) = u(k + 1)$$

as

$$q^2y(k) - 1.5qy(k) + 0.5y(k) = qu(k).$$

We use the z-transform to find the output sequence when the input is a step, namely

$$u(k) = \begin{cases} 0, & k < 0; \\ 1, & k \geq 0. \end{cases}$$

when $y(0) = 0.5$ and $y(-1) = 1$. We have

$$z^2(Y(z) - y(0) - y(1)z^{-1}) - 1.5z(Y(z) - y(0)) + 0.5Y(z) = z(U - u(0)).$$

We need to compute $y(1)$. From the given difference equation we have

$$y(1) = 1.5y(0) - 0.5y(-1) + u(0) = 1.25$$

Thus substituting in the z-transform and rearranging the terms, we get

$$(z^2 - 1.5z + 0.5)Y(z) - 0.5z^2 - 1.25z + 0.75z = zU(z) - z.$$

Thus we have

$$Y(z) = \frac{0.5z(z - 1)}{(z - 1)(z - 0.5)} + \frac{z}{(z - 1)(z - 0.5)}U(z).$$

Now $U(z) = z/(z - 1)$ this we obtain

$$Y(z) = \frac{0.5z}{z - 0.5} + \frac{2}{(z - 1)^2} + \frac{1}{z - 0.5}.$$

Using the following inverse z-transforms

$$\mathcal{Z}^{-1} \left(\frac{z}{z - e^{1/T}} \right) = e^{-k/T}, \quad e^{-1/T} = 0.5 \Rightarrow T = 1/\ln 2$$

$$\mathcal{Z}^{-1} \left(z^{-1} \frac{z}{z - 0.5} \right) = e^{-(k-1)\ln 2}$$

$$\mathcal{Z}^{-1} \left(\frac{1}{(z - 1)^2} \right) = \mathcal{Z}^{-1} \left(z^{-1} \frac{z}{(z - 1)^2} \right) = k - 1$$

we get

$$y(k) = 0.5e^{-k \ln 2} + 2(k - 1) + e^{-(k-1)\ln 2}$$

SOLUTION 2.10

The controller can be written as

$$U(s) = -U_y(s) + U_r(s) = -G_y(s)Y(s) + G_r(s)R(s)$$

where the transfer functions $G_y(s)$ and $G_r(s)$ are given

$$G_y(s) = \frac{s_0s + s_1}{s + r_1} = s_0 + \frac{s_1 - s_0r_1}{s + r_1}$$

$$G_r(s) = \frac{t_0s + t_1}{s + r_1} = t_0 + \frac{t_1 - t_0r_1}{s + r_1}.$$

We need to transform this two transfer functions in state-space form. We have

$$\dot{x}_y(t) = -r_1x_y(t) + (s_1 - s_0r_1)y(t)$$

$$u_y(t) = x_y(t) + s_0y(t)$$

and

$$\dot{x}_r(t) = -r_1x_r(t) + (t_1 - t_0r_1)r(t)$$

$$u_r(t) = x_r(t) + t_0r(t).$$

The sampled systems corresponding to the previous continuous time systems, when the sampling interval in h , are

$$x_y(kh + h) = \Phi x_y(kh) + \gamma_y y(kh)$$

$$u_y(kh) = x_y(kh) + s_0y(kh)$$

and

$$x_r(kh + h) = \Phi x_r(kh) + \gamma_r r(kh)$$

$$u_r(kh) = x_r(kh) + t_0r(kh)$$

where

$$\Phi = e^{-r_1h}$$

$$\gamma_y = \int_0^h e^{-r_1s} ds (s_1 - s_0r_1) = -(e^{-r_1h} - 1) \frac{s_1 - s_0r_1}{r_1}$$

$$\gamma_r = \int_0^h e^{-r_1s} ds (t_1 - t_0r_1) = -(e^{-r_1h} - 1) \frac{t_1 - t_0r_1}{r_1}.$$

From the state representation we can compute the pulse transfer function as

$$u_y(kh) = \left(\frac{\gamma_y}{q - \phi_y} + s_0 \right) y(kh)$$

$$u_r(kh) = \left(\frac{\gamma_r}{q - \phi_r} + t_0 \right) r(kh).$$

Thus the sampled controller in the form asked in the problem is

$$u(kh) = - \underbrace{\frac{s_0q + \gamma_y - s_0\phi_y}{q - \phi_y}}_{H_y(q)} y(kh) + \underbrace{\frac{t_0q + \gamma_r - t_0\phi_r}{q - \phi_r}}_{H_r(q)} r(kh).$$

SOLUTION 2.11

(Ex. 2.21 in [2]) Consider the discrete time filter

$$\frac{z + b}{z + a}$$

(a)

$$\begin{aligned} \arg \left(\frac{e^{i\omega h} + b}{e^{i\omega h} + a} \right) &= \arg \left(\frac{\cos \omega h + b + i \sin \omega h}{\cos \omega h + a + i \sin \omega h} \right) \\ &= \arctan \left(\frac{\sin \omega h}{b + \cos \omega h} \right) - \arctan \left(\frac{\sin \omega h}{a + \cos \omega h} \right). \end{aligned}$$

We have a phase lead if

$$\begin{aligned} \arctan \left(\frac{\sin \omega h}{b + \cos \omega h} \right) &> \arctan \left(\frac{\sin \omega h}{a + \cos \omega h} \right), \quad 0 < \omega h < \pi \\ \frac{\sin \omega h}{b + \cos \omega h} &> \frac{\sin \omega h}{a + \cos \omega h}. \end{aligned}$$

Thus we have lead if $b < a$.

Solutions to analysis of sampled systems

SOLUTION 3.1

The characteristic equation of the closed loop system is

$$z(z - 0.2)(z - 0.4) + K = 0 \quad K > 0.$$

The stability can be determined using the root locus. The starting points are $z = 0$, $z = 0.2$ and $z = 0.4$. The asymptotes have the directions $\pm\pi/3$ and $-\pi$. The crossing of the asymptotes is 0.2. To find the value of K such that the root locus intersects the unit circle we let $z = a + ib$ with $a^2 + b^2 = 1$. This gives

$$(a + ib)(a + ib - 0.2)(a + ib - 0.4) = -K.$$

Multiplying by $a - ib$ and since $a^2 + b^2 = 1$ we obtain

$$a^2 - 0.6a - b^2 + 0.08 + i(2ab - 0.6b) = -K(a - ib).$$

Equating real parts and imaginary parts we obtain

$$\begin{aligned} a^2 - 0.6a - b^2 + 0.08 &= -Ka \\ b(2a - 0.6) &= Kb. \end{aligned}$$

If $b \neq 0$ then

$$\begin{aligned} a^2 - 0.6a - (1 - a^2) + 0.08 &= -a(2a + 0.6) \\ 4a^2 - 1.2a - 0.92 &= 0. \end{aligned}$$

Solving with respect to a we get

$$a = 0.15 \pm \sqrt{0.0225 + 0.23} = \begin{cases} 0.652 \\ -0.352 \end{cases}$$

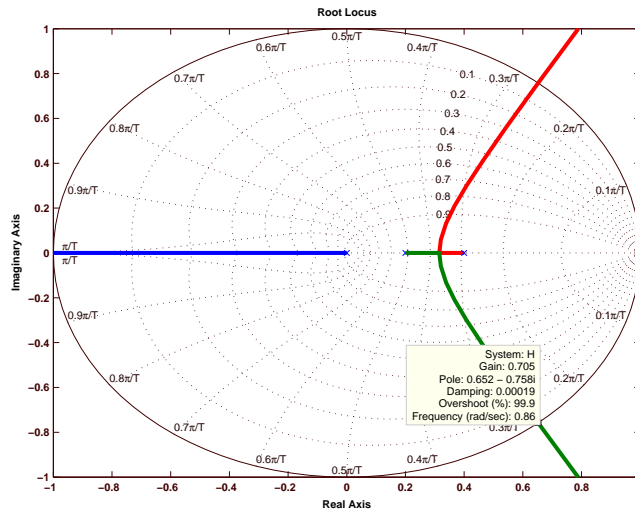


Figure 3.1.1: Root locus for the system in Problem 3.1

This gives $K = 0.70$ and $K = -1.30$. The root locus may also cross the unit circle if $b = 0$ for $a = \pm 1$. A root at $z = -1$ is obtained when

$$-1(-1 - 0.2)(-1 - 0.4) + K = 0$$

namely when $K = 1.68$. There is a root at $z = 1$ when $K = -0.48$. The closed loop system is stable for

$$K \leq 0.70$$

SOLUTION 3.2

We sample the system $G(s)$. In order to do this we derive a state-space realization of the given system

$$\begin{aligned}\dot{x} &= u \\ y &= x\end{aligned}$$

which gives the following matrices of the sampled system

$$\begin{aligned}\Phi &= e^{-h0} = 1 \\ \Gamma &= \int_0^h ds = h.\end{aligned}$$

The pulse transfer operator is

$$H(q) = C(qI - \Phi)^{-1}\Gamma = \frac{h}{q - 1}.$$

(a) When $\tau = 0$ the regulator is

$$u(kh) = Ke(kh)$$

and the characteristic equation of the closed loop system becomes

$$1 + C(z)H(z) = Kh + z - 1 = 0.$$

The system is stable if

$$|1 - Kh| < 1 \Rightarrow 0 < K < 2/h.$$

When there is a delay of one sample, $\tau = h$ then the characteristic equation becomes

$$z^2 - z + Kh = 0.$$

The roots of the characteristic equation (the poles of the system) must be inside the unit circle for guaranteeing stability and thus $|z_1| < 1$ and $|z_2| < 1$. Thus $|z_1||z_2| = |z_1z_2| < 1$. Since $z_1z_2 = Kh$ we have

$$K < 1/h.$$

- (b) Consider the continuous-time system $G(s)$ in series with a time delay of τ seconds. The transfer function is then

$$\bar{G}(s) = \frac{K}{s} e^{-s\tau}.$$

The phase of the system as function of the frequency is

$$\arg \bar{G}(j\omega) = -\frac{\pi}{2} - \omega\tau$$

and the gain is

$$|\bar{G}(j\omega)| = \frac{K}{\omega}.$$

The system is stable if the gain is less than 1 at the cross over frequency, which satisfies

$$-\frac{\pi}{2} - \omega_c\tau = \pi \Rightarrow \omega_c = \frac{\pi}{2\tau}$$

The system is stable if

$$|\bar{G}(j\omega_c)| = \frac{K}{\omega_c} < 1$$

which yields

$$K < \frac{\pi}{2\tau} = \begin{cases} \infty & \tau = 0 \\ \frac{\pi}{2h} & \tau = h \end{cases}$$

The continuous-time system will be stable for all values of K if $\tau = 0$ and for $K < \pi/2h$ when $\tau = h$. This value is about 50% larger than the value obtained for the sampled system in (a).

SOLUTION 3.3

- (a) The observability matrix is

$$W_o = \begin{pmatrix} C \\ C\Phi \end{pmatrix} = \begin{pmatrix} 2 & -4 \\ 1 & -2 \end{pmatrix}.$$

The system is not observable since $\text{rank}(W_o) = 1$, or $\det W_o = 0$.

- (b) The controllability matrix is

$$W_c = (\Gamma \quad \Phi\Gamma) = \begin{pmatrix} 6 & 1 \\ 4 & 1 \end{pmatrix}$$

which has full rank. Thus the system is reachable.

SOLUTION 3.4

The controllability matrix is

$$W_c = (\Gamma \quad \Phi\Gamma) = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 0.5 & 0 \end{pmatrix}$$

which has full rank (check the first two rows of W_c), thus the system is reachable. From the input u we get the system

$$x(k+1) = \begin{pmatrix} 1 & 0 \\ 0 & 0.5 \end{pmatrix} x(k) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} v(k).$$

In this case

$$W_c = \begin{pmatrix} 0 & 0 \\ 1 & 0.5 \end{pmatrix}$$

which has rank one, and thus the system is not reachable from v .

SOLUTION 3.5

The closed loop system is

$$y(k) = H_{cl}(q)r(k) = \frac{C(q)H(q)}{1 + C(q)H(q)}r(k).$$

(a) With $C(q) = K$, $K > 0$ we get

$$y(k) = \frac{K}{q^2 - 0.5q + K}r(k).$$

The characteristic polynomial of the closed loop system is

$$z^2 - 0.5z + K = 0,$$

and stability is guaranteed if the roots of the characteristic polynomial are inside the unit circle. In general for a second order polynomial

$$z^2 + a_1z + a_2 = 0$$

all the roots are inside the unit circle if¹

$$\begin{aligned} a_2 &< 1 \\ a_2 &> -1 + a_1 \\ a_2 &> -1 - a_1. \end{aligned}$$

If we apply this result to the characteristic polynomial of the given system H_{cl} we get

$$\begin{aligned} K &< 1 \\ K &> -1.5 \\ K &> -0.5 \end{aligned}$$

which together with the hypothesis that $K > 0$ gives $0 < K < 1$. The steady state gain is given by

$$\lim_{z \rightarrow 1} H_{cl}(z) = \frac{K}{K + 0.5}.$$

¹The conditions come from the Jury's stability criterion applied to a second order polynomial. For details see pag. 81 in [2].

SOLUTION 3.6

The z -transform of a ramp is given in Table 2, pag. 22 in [1] and we get

$$R(z) = \frac{z}{(z-1)^2}.$$

Using the pulse transfer function from Problem 3.5 and the final value theorem we obtain

$$\lim_{k \rightarrow \infty} e(k) = \lim_{k \rightarrow \infty} (r(k) - y(k)) = \lim_{z \rightarrow 1} \left(\frac{z-1}{z} \underbrace{H_{cl}(z)R(z)}_{F(z)} \right)$$

if $(1-z^{-1})F(z)$ does not have any root on or outside the unit circle. For the $H_{cl}(z)$ as in this case the condition is not fulfilled. Let us consider the steady state of the first derivative of the error signal $e(k)$

$$\lim_{k \rightarrow \infty} \dot{e}(k) = \lim_{z \rightarrow 1} \frac{z-1}{z} \frac{z^2 - 0.5z}{z^2 - 0.5z + K} \frac{z}{(z-1)^2} = \frac{0.5}{K+0.5}$$

which is positive, meaning that in steady state the reference and the output diverge.

SOLUTION 3.7

- (a) (i) - Poles are mapped as $z = e^{sh}$. This mapping maps the left half plane on the unit circle
- (b) (i) - The right half plane is mapped outside the unit circle
- (c) (ii) - Consider the harmonic oscillator:

$$\begin{aligned} \dot{x} &= \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u \\ y &= \begin{pmatrix} 1 & 0 \end{pmatrix} x \end{aligned}$$

which is controllable since

$$W_c = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

has full rank. If we sampled with sampling period h we get

$$\begin{aligned} x(kh+h) &= \begin{pmatrix} \cos \omega h & \sin \omega h \\ -\sin \omega h & \cos \omega h \end{pmatrix} x(kh) + \underbrace{\begin{pmatrix} 1 - \cos \omega h \\ \sin \omega h \end{pmatrix}}_{\Gamma} u(kh) \\ y(kh) &= \begin{pmatrix} 1 & 0 \end{pmatrix} x(kh) \end{aligned}$$

If we choose $h = 2\pi/\omega$ then Γ is the 0 vector and clearly the system is not controllable.

- (d) (ii) - as in (c) we can find example sampling periods that make the system not observable.

SOLUTION 3.8

The open loop system has pulse transfer operator

$$H_0 = \frac{1}{q^2 + 0.4q}$$

and the controller is proportional, thus $C(q) = K$.

(a) The closed loop system has pulse transfer operator

$$H_{cl} = \frac{KH_0}{1 + KH_0} = \frac{K}{q^2 + 0.4q + K}.$$

From the solution of Problem 3.5 we know that the poles are inside the unit circle if

$$K < 1$$

$$K > -1 + 0.4$$

$$K > -1 - 0.4 \Rightarrow -0.6 < K < 1$$

(b) Let $e(k) = r(k) - y(k)$ then

$$E(z) = (1 - H_{cl})R(z).$$

If K is chosen such that the closed loop system is stable, the final-value theorem can be used and

$$\lim_{k \rightarrow \infty} e(k) = \lim_{z \rightarrow 1} \frac{z-1}{z} \frac{1}{1 + KH_0} R(z) = \frac{z-1}{z} \frac{z^2 + 0.4z}{z^2 + 0.4z + K} \frac{z}{z-1} = \frac{1.4}{1.4 + K}$$

If for example we choose $K = 0.5$ then $\lim_{k \rightarrow \infty} e(k) = 0.74$.

SOLUTION 3.9

The closed loop system has the following characteristic equation

$$\det(zI - (\Phi - \Gamma L)) = z^2 - (a_{11} + a_{22} - b_2 \ell_2 - b_1 \ell_1)z + a_{11}a_{22} - a_{12}a_{21} + (a_{12}b_2 - a_{22}b_1)\ell_1 + (a_{21}b_1 - a_{11}b_2)\ell_2.$$

This must be equal to the given characteristic equations, thus

$$\begin{pmatrix} b_1 & b_2 \\ a_{12}b_2 - a_{22}b_1 & a_{21}b_1 - a_{11}b_2 \end{pmatrix} \begin{pmatrix} \ell_1 \\ \ell_2 \end{pmatrix} = \begin{pmatrix} p_1 + \text{tr } \Phi \\ p_2 - \det \Phi \end{pmatrix}$$

where $\text{tr } \Phi = a_{11} + a_{22}$ and $\det \Phi = a_{11}a_{22} - a_{12}a_{21}$. The solution is

$$\begin{pmatrix} \ell_1 \\ \ell_2 \end{pmatrix} = \frac{1}{\Delta} \begin{pmatrix} a_{21}b_1 - a_{11}b_2 & -b_2 \\ -a_{12}b_2 + a_{22}b_1 & b_1 \end{pmatrix} \begin{pmatrix} p_1 + \text{tr } \Phi \\ p_2 - \det \Phi \end{pmatrix}$$

where $\Delta = a_{21}b_1^2 - a_{12}b_2^2 + b_1b_2(a_{22} - a_{11})$. To check when $\Delta = 0$ we consider the controllability matrix of the system

$$W_c = (\Gamma \quad \Phi\Gamma) = \begin{pmatrix} b_1 & a_{11}b_1 + a_{12}b_2 \\ b_2 & a_{21}b_1 + a_{22}b_2 \end{pmatrix}$$

and we find that $\Delta = \det W_c$. There exists a solution to the system of equation above if the system is controllable, since then the rank of the controllability matrix is full and $\det W_c \neq 0$.

For the double integrator $a_{11} = a_{12} = a_{22} = b_2 = 1$ and $a_{21} = 0$ and $b_1 = 0.5$. In order to design an dead beat controller we need to place the poles in zero, thus $p_1 = p_2 = 0$. This yields

$$\begin{pmatrix} \ell_1 \\ \ell_2 \end{pmatrix} = -1 \begin{pmatrix} -1 & -1 \\ -0.5 & -0.5 \end{pmatrix} \begin{pmatrix} 2 \\ -1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1.5 \end{pmatrix}.$$

SOLUTION 3.10

In this case the characteristic equation is

$$(z - 0.1)(z - 0.25) = z^2 - 0.35z + 0.025.$$

Using the result from Problem 3.9 we find that $\Delta = 0.5$ and L is obtained from

$$L^T = \begin{pmatrix} \ell_1 \\ \ell_2 \end{pmatrix} = \frac{1}{\Delta} \begin{pmatrix} 0.5 & 0 \\ 0.1 & 1 \end{pmatrix} \begin{pmatrix} 0.75 \\ -0.025 \end{pmatrix} = \begin{pmatrix} 0.75 \\ 0.1 \end{pmatrix}.$$

SOLUTION 3.11

(a) The static observer gives

$$\hat{x}(k) = \Phi^{n-1}W_o^{-1} \begin{pmatrix} y(k-n+1) \\ \vdots \\ y(k) \end{pmatrix} + \underbrace{(\Phi^{n-2}\Gamma \quad \Phi^{n-3}\Gamma \quad \dots \quad \Gamma) - \Phi^{n-1}W_o^{-1}}_{\Psi} \underbrace{\begin{pmatrix} 0 & 0 & \dots & 0 \\ C\Gamma & 0 & \dots & 0 \\ C\Phi\Gamma & C\Gamma & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C\Phi^{n-2}\Gamma & C\Phi^{n-3}\Gamma & \dots & C\Gamma \end{pmatrix}}_{W_u} \begin{pmatrix} u(k-n+1) \\ \vdots \\ u(k-1) \end{pmatrix}.$$

In this case we have

$$\begin{aligned} W_o &= \begin{pmatrix} C \\ C\Phi \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0.22 & 1 \end{pmatrix} & W_o^{-1} &= \begin{pmatrix} -4.55 & 4.55 \\ 1 & 0 \end{pmatrix} \\ W_u &= \begin{pmatrix} 0 \\ C\Gamma \end{pmatrix} = \begin{pmatrix} 0 \\ 0.03 \end{pmatrix} \\ \Psi &= \Gamma - \Phi W_o^{-1} W_u \\ &= \begin{pmatrix} 0.22 \\ 0.03 \end{pmatrix} - \begin{pmatrix} 0.78 & 0 \\ 0.22 & 1 \end{pmatrix} \begin{pmatrix} -4.55 & 4.55 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0.03 \end{pmatrix} = \begin{pmatrix} 0.114 \\ 0 \end{pmatrix}. \end{aligned}$$

Thus we get

$$\begin{aligned} \hat{x}(k) &= \Phi W_o^{-1} \begin{pmatrix} y(k-1) \\ y(k) \end{pmatrix} + \Psi u(k-1) \\ &= \begin{pmatrix} -3.55 & 3.55 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} y(k-1) \\ y(k) \end{pmatrix} + \begin{pmatrix} 0.114 \\ 0 \end{pmatrix} u(k-1). \end{aligned}$$

(b) The dynamic observer has the form

$$\hat{x}(k+1|k) = (\Phi - KC)\hat{x}(k|k-1) + \Gamma u(k) + Ky(k).$$

We choose K such that the the eigenvalues of $\Phi - KC$ are in the origin (very fast observer). Using the results in Problem 3.9 where we use Φ^T and C^T instead of Φ and Γ , we obtain

$$K = \begin{pmatrix} 2.77 \\ 1.78 \end{pmatrix}.$$

(c) The reduced observer has the form

$$\hat{x}(k|k) = (I - KC) [\Phi \hat{x}(k-1|k-1) + \Gamma u(k-1)] + Ky(k).$$

In this case we want to find K such that

- $CK = 1$
- $(I - KC)\Phi$ has the eigenvalues in the origin.

The first condition imposes $k_2 = 1$. Since

$$(I - KC)\Phi = \begin{pmatrix} 0.78 - 0.22k_1 & -k_1 \\ 0 & 0 \end{pmatrix}$$

in order to have eigenvalues in the origin we need to choose $k_1 = 0.78/0.22 = 3.55$. The reduced observer is then

$$\hat{x}(k|k) = \begin{pmatrix} 0 & -3.55 \\ 0 & 0 \end{pmatrix} \hat{x}(k-1|k-1) + \begin{pmatrix} 0.114 \\ 0 \end{pmatrix} u(k-1) + \begin{pmatrix} 3.55 \\ 1 \end{pmatrix} y(k).$$

Since $\hat{x}_2(k|k) = y(k)$ the we get

$$\hat{x}(k|k) = \begin{pmatrix} -3.55 & 3.55 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} y(k-1) \\ y(k) \end{pmatrix} + \begin{pmatrix} 0.114 \\ 0 \end{pmatrix} u(k-1)$$

which is the same as the static observer computed in (a).

SOLUTION 3.12

The constant disturbance $v(k)$, which typically has high energy at low frequencies, can be described by the dynamical system

$$\begin{aligned} w(k+1) &= A_w w(k) \\ v(k) &= C_w w(k), \end{aligned}$$

where the matrix A_w typically has eigenvalues at the origin or on the imaginary axis. We consider the augmented state vector

$$z = \begin{pmatrix} x \\ w \end{pmatrix}.$$

The augmented system can be described by

$$\begin{aligned} \begin{pmatrix} \dot{x} \\ \dot{w} \end{pmatrix} &= \begin{pmatrix} A & C_w \\ 0 & A_w \end{pmatrix} \begin{pmatrix} x \\ w \end{pmatrix} + \begin{pmatrix} B \\ 0 \end{pmatrix} u \\ y &= (C \ 0) \begin{pmatrix} x \\ w \end{pmatrix} \end{aligned}$$

Sampling the system gives the following discrete-time system

$$\begin{aligned} \begin{pmatrix} x(k+1) \\ w(k+1) \end{pmatrix} &= \begin{pmatrix} \Phi & \Phi_{xw} \\ 0 & \Phi_w \end{pmatrix} \begin{pmatrix} x(k) \\ w(k) \end{pmatrix} + \begin{pmatrix} \Gamma \\ 0 \end{pmatrix} u(k) \\ y(k) &= (C \ 0) \begin{pmatrix} x(k) \\ w(k) \end{pmatrix} \end{aligned}$$

where

$$\begin{aligned} w(k+1) &= \Phi_w w(k) \\ v(k) &= C_w w(k), \end{aligned}$$

and Φ_{xw} relates $x(k+1)$ and $w(k)$. In this exercise the disturbance can be modeled by the system

$$\begin{aligned} w(k+1) &= w(k) \\ v(k) &= w(k), \end{aligned}$$

and the process is described by

$$\begin{aligned} \Phi_w &= 1 \\ \Phi_{xw} &= \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \end{aligned}$$

(a) If the state x and the disturbance v can be measured then we can use the controller

$$u(k) = -Lx(k) - L_w w(k).$$

This gives the closed loop system

$$\begin{aligned} x(k+1) &= \Phi x(k) + \Phi_{xw} w(k) - \Gamma L x(k) - \Gamma L_w w(k) \\ y(k) &= C x(k) \end{aligned}$$

In general it is not possible to eliminate completely the influence of $w(k)$. This is possible only if $\Phi_{xw} - \Gamma L_w = 0$. We will therefore consider the situation at the output in steady state

$$y(\infty) = C (I - (\Phi - \Gamma L))^{-1} (\Phi_{xw} - \Gamma L_w) w(\infty) = H_w(1) w(\infty).$$

The influence of w (or v) can be zero in steady state if

$$H_w(1) = 0.$$

Let ϕ_{ij} the (i, j) element of $\Phi - \Gamma L$ and γ_i the i th element of Γ . Then

$$C (I - (\Phi - \Gamma L))^{-1} (\Phi_{xw} - \Gamma L_w) = -\frac{1 - L_w \gamma_1 - \phi_{22} + \phi_{22} L_w \gamma_1 - \phi_{12} L_w \gamma_2}{-1 + \phi_{22} + \phi_{11} - \phi_{11} \phi_{22} + \phi_{12} \phi_{21}} = 0$$

yields

$$L_w = \frac{-1 + \phi_{22}}{-\gamma_1 + \phi_{22} \gamma_1 - \phi_{12} \gamma_2}.$$

If L is the feedback matrix that gives a dead beat controller, that is

$$L = (3.21 \quad 5.57)$$

such that

$$\Phi - \Gamma L = \begin{pmatrix} -0.142 & -0.114 \\ 0.179 & 0.142 \end{pmatrix}$$

then we have $L_w = 5.356$.

(b) In this case the state is measurable, but not the disturbance. The disturbance can be calculated from the state equation

$$\Phi_{xw} w(k-1) = x(k) - \Phi x(k-1) - \Gamma u(k-1).$$

The first element of this vector gives

$$w(k-1) = (1 \ 0) (x(k) - \Phi x(k-1) - \Gamma u(k-1)).$$

Since $w(k)$ is constant and $x(k)$ is measurable, it is possible to calculate $\hat{w}(k) = w(k-1)$. The following control law can then be used

$$u(k) = -Lx(k) - L_w \hat{w}(k)$$

where L and L_w are the same as in (a). Compared with the controller in (a) there is a delay in the detection of the disturbance.

- (c) If only the output is measurable then the state and the disturbance can be estimated by using the following observer:

$$\begin{pmatrix} \hat{x}(k+1) \\ \hat{w}(k+1) \end{pmatrix} = \begin{pmatrix} \Phi & \Phi_{xw} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \hat{x}(k) \\ \hat{w}(k) \end{pmatrix} + \begin{pmatrix} \Gamma \\ 0 \end{pmatrix} u(k) + \begin{pmatrix} K \\ K_w \end{pmatrix} \epsilon(k)$$

$$\epsilon(k) = y(k) - C\hat{x}(k).$$

The gains K and K_w can be determined so that the error goes to zero, provided that the augmented system is observable. Let $\tilde{x}(k) = \hat{x}(k) - x(k)$ and similarly $\tilde{w}(k) = \hat{w}(k) - w(k)$. Then

$$\begin{pmatrix} \tilde{x}(k+1) \\ \tilde{w}(k+1) \end{pmatrix} = \begin{pmatrix} \Phi - KC & \Phi_{xw} \\ -K_w C & 1 \end{pmatrix} \begin{pmatrix} \tilde{x}(k) \\ \tilde{w}(k) \end{pmatrix}.$$

The characteristic equation of the system matrix for the error is

$$z^3 + (k_1 - 2.2)z^2 + (1.05 - 1.7k_1 + k_2 + k_w)z + 0.7k_1 + 0.15 - 0.7k_w - k_2 = 0.$$

The eigenvalues can be placed at the origin if

$$K = \begin{pmatrix} 2.2 \\ -0.64 \end{pmatrix} K_w = 3.33.$$

The controller is

$$u(k) = -L\hat{x}(k) - L_w\hat{w}(k)$$

where L and L_w are the same as (a).

SOLUTION 3.13

- (a) The eigenvalues of the matrix A are

$$\begin{aligned} \lambda_1 &= -0.0197 \\ \lambda_2 &= -0.0129. \end{aligned}$$

The matrices of the sampled system are

$$\Phi = e^{Ah} = \begin{pmatrix} 0.790 & 0 \\ 0.176 & 0.857 \end{pmatrix}$$

$$\Gamma = \begin{pmatrix} 0.281 \\ 0.0296 \end{pmatrix}$$

- (b) The pulse transfer operator is given by

$$H(q) = C(qI - \Phi)^{-1}\Gamma = (0 \ 1) \begin{pmatrix} q - 0.790 & 0 \\ -0.176 & q - 0.857 \end{pmatrix}^{-1} \begin{pmatrix} 0.281 & 0.0296 \end{pmatrix} = \frac{0.030q + 0.026}{q^2 - 1.65q + 0.68}$$

- (c) The poles of the continuous-time system are at -0.0197 and -0.0129. The observer should be twice as fast as the fastest mode of the open-loop system, thus we choose the poles of the observer in

$$z = e^{-0.0192 \cdot 2 \cdot 12} = 0.62.$$

The desired characteristic equation of $\Phi - KC$ is then

$$z^2 - 1.24z + 0.38 = 0.$$

Using the results from Problem 3.9 we obtain

$$K = (0.139 \ 0.407).$$

SOLUTION 3.14

(a) The sampled system is

$$\begin{aligned}x(k+1) &= \Phi x(k) + \Gamma u(k) \\ y(k) &= x(k)\end{aligned}$$

where

$$\begin{aligned}\Phi &= e^{-5} \\ \Gamma &= \frac{1 - e^{-5}}{5}.\end{aligned}$$

(b) In order to prove stability with Lyapunov argument we need to choose a Lyapunov function V . Let $V = |x|$. The increment of the Lyapunov function is then

$$\Delta V = |x^+| - |x| = |e^{-5}x| - |x| = (e^{-5} - 1)|x| \leq 0.$$

since $u(k) = 0$. Since $\Delta V \leq 0$, then the system is stable. We actually know that the system is asymptotically stable, since the pole is inside the unit circle. We can conclude on *asymptotical* stability using the Lyapunov argument, noticing that $\Delta V < 0$ if $x \neq 0$. Thus the system is asymptotically stable.

SOLUTION 3.15

(a) The sampled system is

$$\begin{aligned}x(k+1) &= \Phi x(k) + \Gamma u(k) \\ y(k) &= x(k)\end{aligned}$$

where

$$\begin{aligned}\Phi &= \begin{pmatrix} e^{-1} & 0 \\ 0 & e^{-2} \end{pmatrix} \\ \Gamma &= \begin{pmatrix} 1 - e^{-1} \\ \frac{1 - e^{-2}}{2} \end{pmatrix}.\end{aligned}$$

(b) The characteristic polynomial is

$$(z - 0.1)(z - 0.2) = z^2 - 0.3z + 0.02.$$

Using the result from Problem 3.9 we obtain

$$\begin{pmatrix} \ell_1 \\ \ell_2 \end{pmatrix} = \frac{1}{-0.0636} \begin{pmatrix} -e^{-1} \frac{1 - e^{-2}}{2} & -\frac{1 - e^{-2}}{2} \\ -e^{-2}(1 - e^{-1}) & 1 - e^{-1} \end{pmatrix} \begin{pmatrix} -0.3 + e^{-1} + e^{-2} \\ 0.02 - e^{-1}e^{-2} \end{pmatrix} = \begin{pmatrix} 0.3059 \\ 0.0227 \end{pmatrix}$$

(c) The closed loop system is

$$x(k+1) = \underbrace{\begin{pmatrix} 0.1745 & -0.0143 \\ -0.1323 & 0.1255 \end{pmatrix}}_{\Phi_c} x(k).$$

We consider the following Lyapunov function

$$V(x) = x^T x$$

then

$$\Delta V(x) = x^T \Phi_c^T \Phi_c x - x^T x = x^T \underbrace{(\Phi_c^T \Phi_c - I)}_{\Omega, \text{ symmetric}} x.$$

Since the eigenvalues of the symmetric matrix Ω are negative then $\Delta V \leq 0$. Thus the closed loop system is stable. Notice that $\Delta V < 0$ if $x \neq 0$ thus the closed loop system is asymptotically stable (the eigenvalues are placed in 0.1 and 0.2).

Solutions to computer realization of controllers

SOLUTION 4.1

(a) The characteristic polynomial of the closed loop system is equal to the numerator of $1 + HH_c$, that is,

$$z^2 + (K + K_i - 2.5)z - K + 1.$$

The poles of the closed-loop system are in the origin if

$$\begin{aligned} K + K_i - 2.5 &= 0 \\ -K + 1 &= 0 \end{aligned}$$

which yields $K = 1$ and $K_i = 1.5$.

(b) We can use the following partial fraction expansion for $H_c(z)$

$$H_c(z) = M + \frac{N}{z - 1}.$$

With simple calculations we obtain $M = 2.5$ and $N = 1.5$. Thus the state-space representation of the controller is

$$\begin{aligned} x(k+1) &= x(k) + e(k) \\ u(k) &= 1.5x(k) + 2.5e(k) \end{aligned}$$

SOLUTION 4.2

(a) Using Euler's method we get

$$H(z) = \frac{a}{(z-1)/h + a} = \frac{ah}{z-1+ah}.$$

This corresponds to the difference equation

$$y(kh+h) + (ah-1)y(kh) = ah u(kh).$$

The difference equation is stable if

$$|ah-1| < 1 \Rightarrow 0 < h < 2/a.$$

The approximation may be poor even if the difference equation is stable.

(b) Tustin's approximation gives

$$H(z) = \frac{a}{\frac{2z-1}{hz+1} + a} = \frac{(z+1)ah/2}{(1+ah/2)z + (ah/2-1)}$$

$$= \frac{ah/2}{1+ah/2} \frac{z+1}{z + \frac{ah/2-1}{ah/2+1}}$$

The pole of the discrete-time system will vary from 1 to -1 when h varies from 0 to ∞ . The discrete-time approximation is always stable if $a > 0$.

(c) Tustin's approximation with prewarping gives

$$H(z) = \frac{a}{\alpha \frac{z-1}{z+1} + a} = \frac{a/\alpha}{1+a/\alpha} \frac{z+1}{z + \frac{a/\alpha-1}{a/\alpha+1}}$$

where

$$\alpha = \frac{a}{\tan(ah/2)}$$

SOLUTION 4.3

(a) Euler's method gives

$$H(z) = 4 \frac{(z-1)/h + 1}{(z-1)/h + 2} = 4 \frac{z-1+h}{z-1+2h} = 4 \frac{z-0.75}{z-0.5}$$

(b) Backward differences give

$$H(z) = 4 \frac{(z-1)/(zh) + 1}{(z-1)/(zh) + 2} = 4 \frac{z(1+h) - 1}{z(1+2h) + 1} = 3.33 \frac{z-0.80}{z-0.667}$$

(c) Tustin's approximation gives

$$H(z) = 4 \frac{\frac{2z-1}{hz+1} + 1}{\frac{2z-1}{hz+1} + 2} = 4 \frac{z(1+h/2) - (1-h/2)}{z(1+h) - (1-h)} = 3.6 \frac{z-0.778}{z-0.6}$$

(d) Tustin's approximation with pre-warping

$$H(z) = 4 \frac{\alpha \frac{z-1}{z+1} + 1}{\alpha \frac{z-1}{z+1} + 2} = 4 \frac{z(1+1/\alpha) - (1-1/\alpha)}{z(1+2/\alpha) - (1-2/\alpha)} = 3.596 \frac{z-0.775}{z-0.596}$$

All the four approximation has the form

$$H(z) = K \frac{z+a}{z+b}$$

The gain and phase at $\omega = 1.6\text{rad/s}$ are obtained from

$$H(e^{i\omega h}) = K \frac{e^{i\omega h} + a}{e^{i\omega h} + b} = K \frac{(e^{i\omega h} + a)(e^{-i\omega h} + b)}{(e^{i\omega h} + b)(e^{-i\omega h} + b)}$$

$$= K \frac{1 + ab + (a + b) \cos(\omega h) + i(b - a) \sin(\omega h)}{1 + b^2 + 2b \cos(\omega h)}$$

$$\arg H(e^{i\omega h}) = \arctan \frac{(b - a) \sin \omega h}{1 + ab + (a + b) \cos(\omega h)}$$

$$|H(e^{i\omega h})| = K \sqrt{\frac{1 + a^2 + 2a \cos(\omega h)}{1 + b^2 + 2b \cos(\omega h)}}$$

The four different approximations give at $\omega = 1.6\text{rad/s}$ the following results

	$ H(\cdot) $	$\arg H(\cdot)$	Rel. err. $ \cdot $	Rel. err. $\arg(\cdot)$
Continuous-time (no approx.)	2.946680646	0.3374560692		
Euler	2.966414263	0.4105225474	0.67%	21.65%
Backward	2.922378065	0.2772636846	-0.82%	-17.83%
Tustin	2.959732059	0.3369122161	0.44%	-0.16%
Tustin with prewarping	2.946680646	0.3374560692	0.0%	0.0%

SOLUTION 4.4

The tank process in Problem 2.13 has the transfer function

$$G(s) = \frac{0.000468}{(s + 0.0197)(s + 0.0129)}$$

(a) At the desired cross-over frequency we have

$$|G(i\omega_c)| = 0.525$$

$$\arg G(i\omega_c) = -115^\circ.$$

We use a PI controller in the form

$$F(s) = \frac{K(Ts + 1)}{Ts}$$

and we have the following specifications at ω_c

- gain $1/0.525$
- phase -15° .

This gives $K = 1.85$ and $T = 149$.

(b) The characteristic equation of the closed loop system is

$$s^2 + 0.0326s^2 + 0.00112s + 5.91 \cdot 10^{-6} = 0$$

which has roots $s_{1,2} = -0.0135 \pm i0.0281$ and $s_3 = -0.006$. The complex poles have a damping of $\zeta = 0.43$. The zero of the closed loop system is -0.0062 .

(c) Tustin's approximation with given warping is

$$H_c(z) = \frac{1.85 \left(\alpha \frac{z-1}{z+1} + 0.0067 \right)}{\alpha \frac{z-1}{z+1}} = \frac{1.85(\alpha + 0.0067)}{\alpha} \left(1 + \frac{0.0134}{(\alpha + 0.0067)(z-1)} \right).$$

The rule of thumb for the selection of the sampling period gives

$$h \approx 6 - 20 \text{sec.}$$

The choice of $h = 12$ seems to be reasonable. This gives $\alpha = 0.165$ and

$$H_c(z) = 1.925 \left(1 + \frac{0.0778}{z-1} \right).$$

SOLUTION 4.5

The sample and zero-order hold circuit can be approximate by a delay of $h/2$ seconds. Indeed the output of the zero-order hold is

$$u_h(t) = u(kh), \quad kh < t < kh + h$$

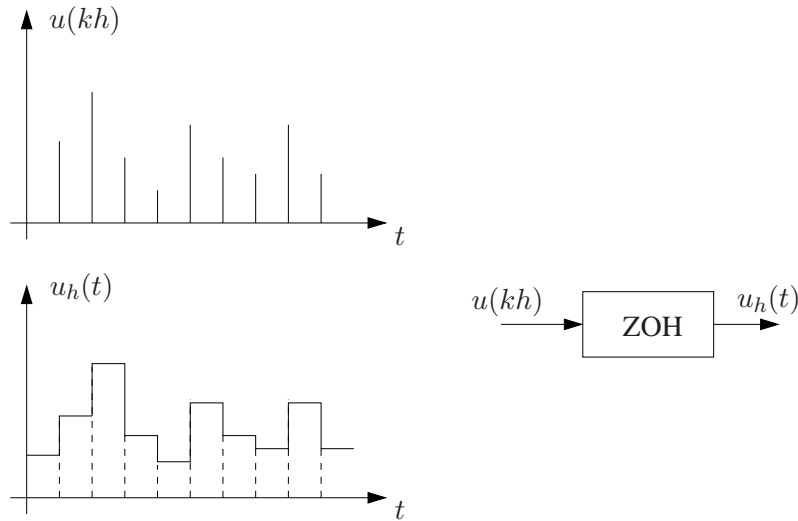


Figure 4.5.1: The zero-order hold.

If $u(kh) = \delta(kh)$ then $u_h(t)$ is the impulse response of the ZOH filter. In this case the output, let us call it, $u_h^\delta(t)$ is a pulse of height 1 and duration h i.e.,

$$u_h^\delta(t) = (1(t) - 1(t-h)).$$

The Laplace transform of the impulse response is the transfer function of the ZOH filter, which is

$$ZOH(s) = \mathcal{L}\{u_h^\delta(t)\} = \int_0^\infty \frac{1}{h} (1(t) - 1(t-h)) e^{-st} dt = (1 - e^{-sh})/s.$$

When we sample a signal we have a scaling factor of $1/h$, as

$$X_s(j\omega) = \frac{1}{h} \sum_{k=-\infty}^{\infty} X(j(\omega + k\omega_s)).$$

Thus for small h we have

$$SAMPLE(s) \cdot ZOH(s) = \frac{1}{h} \frac{(1 - e^{-sh})}{s} = \frac{1 - 1 + sh - (sh)^2/2 + \dots}{sh} = 1 - \frac{sh}{2} + \dots \approx e^{-sh/2}$$

which is approximately a delay of $h/2$. If we assume a decrease of the phase margin of $5^\circ - 15^\circ$, then

$$\Delta\phi_{ZOH} = \omega_c h/2 = \frac{180^\circ \omega_c h}{2\pi} = \frac{\omega_c h}{0.035} = 5^\circ - 15^\circ,$$

which then gives

$$\omega_c h = 0.17 - 0.52$$

or

$$\omega_c h \approx 0.15 - 0.5.$$

SOLUTION 4.6

Consider the general problem of controlling a continuous-time system

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t)\end{aligned}$$

with the continuous-time controller

$$u(t) = Mr(t) - Lx(t).$$

The closed-loop system is

$$\begin{aligned}\dot{x}(t) &= (A - BL)x(t) + BMr(t) = A_c x(t) + BMr(t) \\ y(t) &= Cx(t).\end{aligned}$$

If $r(t)$ is constant over one sampling period, then the previous equation can be sampled, giving

$$\begin{aligned}x(kh + h) &= \Phi_c x(kh) + \Gamma_c Mr(kh) \\ y(kh) &= Cx(kh),\end{aligned}\tag{0.0}$$

where

$$\begin{aligned}\Phi_c &= e^{A_c h} \\ \Gamma_c &= \int_0^h e^{A_c s} ds B.\end{aligned}$$

Let us assume that the controller

$$u(kh) = \tilde{M}r(kh) - \tilde{L}x(kh)$$

is used to control the sampled system

$$\begin{aligned}x(kh + h) &= \Phi x(kh) + \Gamma u(kh) \\ y(kh) &= Cx(kh),\end{aligned}$$

where

$$\begin{aligned}\Phi &= e^{Ah} \\ \Gamma &= \int_0^h e^{As} ds B.\end{aligned}$$

In this case the closed-loop system is

$$\begin{aligned}x(kh + h) &= (\Phi - \Gamma\tilde{L})x(kh) + \Gamma\tilde{M}r(kh) \\y(kh) &= Cx(kh).\end{aligned}\tag{0.0}$$

It is in general not possible to choose \tilde{L} such that

$$\Phi_c = \Phi - \Gamma\tilde{L}.$$

However, we can make a series expansion and equate terms. Assume

$$\tilde{L} = L_0 + L_1h/2$$

then

$$\Phi_c \approx I - (A - BL)h + (A^2 - BLA - ABL - (BL)^2)h^2/2 + \dots$$

and

$$\Phi - \Gamma\tilde{L} \approx I + (A - BL_0)h + (A^2 - ABL_0 - (BL_1)^2)h^2/2 + \dots$$

The systems (0.0) and (0.0) have the same poles up to and including order h^2 if

$$\tilde{L} = L(I + (A - BL)h/2).$$

The modification on \tilde{M} is determined by assuming the steady-state values of (0.0) and (0.0) are the same. Let the reference be constant and assume that the steady-state value of the state is x^0 . This gives the relations

$$(I - \Phi_c)x^0 = \Gamma_cMr$$

and

$$(I - (\Phi - \Gamma\tilde{L}))x^0 = \Gamma\tilde{M}r$$

The series expansion of the left-hand sides of these two relations are equal for power of h up to and including h^2 . Then, we need to determine \tilde{M} so that the expansions of the right-hands are the same for h and h^2 . Assuming

$$\tilde{M} = M_0 + M_1h/2$$

then

$$\Gamma_cM \approx BMh + (A - BL)BMh^2/2 + \dots$$

and

$$\Gamma\tilde{M} \approx BM_0h + (BM_1 - ABM_0)h^2/2 + \dots,$$

which gives

$$\tilde{M} = (I - LBh/2)M.$$

(a) We need to compute \tilde{L} and \tilde{M} , which are

$$\begin{aligned}\tilde{L} &= L \begin{pmatrix} 1 & h/2 \\ -h/2 & 1-h \end{pmatrix} = \begin{pmatrix} 1 & 2 \\ -h/2 & 1-h \end{pmatrix} = \begin{pmatrix} 0.8 & 1.7 \\ -0.5 & 0.8 \end{pmatrix} \\ \tilde{M} &= 2 - 2h = 1.6\end{aligned}$$

(b) The backward difference approximation gives

$$\begin{aligned}\frac{1 - q^{-1}}{h}\hat{x}(kh) &= (A - KC)\hat{x}(kh) + Bu(kh) + Ky(kh) \\ (I - Ah - KCh)\hat{x}(kh) &= q^{-1}\hat{x}(kh) + Bu(kh) + Ky(kh).\end{aligned}$$

Let

$$\Phi_0 = (I - Ah - KCh)^{-1} = \frac{1}{h^2 + h + 1} \begin{pmatrix} 1 & h \\ -h & 1 + h \end{pmatrix}.$$

This gives,

$$\begin{aligned} \hat{x}(kh) &= (\Phi_0 \hat{x}(kh - h) + \Phi_0 Bu(kh) + \Phi_0 Ky(kh)) \\ &= \begin{pmatrix} 0.81 & 0.16 \\ -0.16 & 0.97 \end{pmatrix} \hat{x}(kh - h) + \begin{pmatrix} 0.03 \\ 0.19 \end{pmatrix} u(kh) + \begin{pmatrix} 0.19 \\ 0.16 \end{pmatrix} y(kh). \end{aligned}$$

- (c) Simulate the continuous-time controller and the discrete-time approximation. Let $x(0) = (1, 1)^T$ and $\hat{x}(t) = (0, 0)^T$.

SOLUTION 4.7

- (a) To obtain the forward difference approximation we substitute $s = \frac{q-1}{h}$ in the continuous-time expression. We obtain

$$u(kh) = -\frac{s_0q - s_0 + hs_1}{q + r_1h - 1} y(kh) + \frac{t_0q - t_0 + ht_1}{q + r_1h - 1} r(kh)$$

- (b) In order to compare the discretizations we examine the location of the controller pole. We use the graph available in order to compute the poles for the exact discretization. We have the following results:

Discretization	$h = 0.01$	$h = 0.1$	$h = 1$
Exact	0.9048	0.3679	0.00004540
Forward difference	0.9	0	-9

We notice that the forward difference approximation yields an unstable system for $h = 1$ (pole in -9). For $h = 0.1$, the forward difference approximation gives a pole at the origin, which corresponds to a one step delay. Thus this approximations of the continuous-time controller cannot be considered satisfactory. For $h = 0.01$ the pole of the forward difference approximation is very closed to the exact discretization which means that this is a better choice.

We could also consider how the sampling interval relates to the dominating time constant of the system. We know that the relation between this two parameters is given by the following rule-of-thumb

$$N_r = \frac{T_r}{h} \approx 4 - 10.$$

In this case only the controller dynamics are available, but using the time constant of the controller that is $T_c = 1/r_1 = 0.1$ we see that only for $h = 0.01$ the rule-of-thumb is fulfilled.

SOLUTION 4.8

- (a) The backward differences approximation substitutes s with $(z - 1)/zh$. We can consider the Laplace transform of the controller's equation yields

$$\begin{aligned} sX(s) &= AX(s) + BE(s) \\ U(s) &= CX(s) + DE(s) \end{aligned}$$

Substituting s with $(z - 1)/zh$ we get the following equations

$$\begin{aligned} x(k + 1) - x(k) &= hAx(k + 1) + hBe(k + 1) \\ u(k) &= Cx(k) + De(k). \end{aligned}$$

We consider first the state update equation. Dividing the signals at time $k + 1$ and k we get the following

$$x(k + 1) - hAx(k + 1) - hBe(k + 1) = x(k) =: w(k + 1).$$

Solving this equation for x in terms of w and e we get

$$x(k + 1) = (I - hA)^{-1}w(k + 1) + (I - hA)^{-1}Bhe(k + 1)$$

thus, since $w(k + 1) = x(k)$ we get

$$w(k + 1) = (I - hA)^{-1}w(k) + (I - hA)^{-1}Bhe(k).$$

This gives that

$$\begin{aligned}\Phi_c &= (I - hA)^{-1} \\ \Gamma_c &= (I - hA)^{-1}Bh.\end{aligned}$$

From the output equation substituting $x(k)$ we get directly

$$u(k) = C(I - hA)^{-1} + \{D + C(I - hA)^{-1}Bh\}e(k).$$

Thus

$$\begin{aligned}H &= C(I - hA)^{-1} \\ J &= D + C(I - hA)^{-1}Bh.\end{aligned}$$

- (b) To compute the Tustin's approximation we proceed in the same way, and we substitute s with $2(z - 1)/(h(z + 1))$. The state equation then becomes

$$x(k + 1) - x(k) = \frac{Ah}{2}(x(k + 1) - x(k)) + \frac{Bch}{2}(e(k + 1) + e(k)).$$

Again collecting the $k + 1$ terms on one side we get

$$\begin{aligned}x(k + 1) - \frac{Ah}{2}x(k) - \frac{Bh}{2}e(k + 1) &= x(k) + \frac{Ah}{2}x(k) + \frac{Bh}{2}e(k) \\ &=: w(k + 1).\end{aligned}$$

Thus we can derive $x(k + 1)$ from the previous equation as function of $w(k + 1)$ and $e(k + 1)$

$$x(k + 1) = (I - \frac{Ah}{2})^{-1}w(k + 1) - (I - \frac{Ah}{2})^{-1}\frac{Bh}{2}e(k + 1).$$

The state equation becomes

$$w(k + 1) = \underbrace{(I + \frac{Ah}{2})(I - \frac{Ah}{2})^{-1}}_{\Phi_c} w(k) + \underbrace{((I + \frac{Ah}{2})(I - \frac{Ah}{2})^{-1} + I)\frac{Bh}{2}}_{\Gamma_c} e(k).$$

The output equation becomes

$$u(k) = \underbrace{C(I - \frac{Ah}{2})^{-1}}_{H_c} w(k) + \underbrace{\{D + C(I - \frac{Ah}{2})^{-1}\frac{Bh}{2}\}}_{J_c} e(k).$$

Notice that is possible to write Γ_c in a more compact way as follows

$$\begin{aligned}((I + \frac{Ah}{2})(I - \frac{Ah}{2})^{-1} + I)\frac{Bh}{2} &= ((I + \frac{Ah}{2}) + (I - \frac{Ah}{2}))(I - \frac{Ah}{2})^{-1}\frac{Bh}{2} \\ &= (I - \frac{Ah}{2})^{-1}\frac{Bh}{2}.\end{aligned}$$

Solutions to implementation aspects

SOLUTION 5.1

The poles of the controller are

$$z_1 = 1; \quad z_2 = \frac{1}{2}; \quad z_{3,4} = -1/4 \pm i \frac{\sqrt{3}}{4}$$

We can represent the controller in Jordan form as

$$\begin{aligned} \Sigma_1 : s_1(k+1) &= 1 s_1(k) + 2.286y(k) \\ \Sigma_2 : s_2(k+1) &= \frac{1}{2} s_2(k) - 3.073y(k) \\ \Sigma_3 : \begin{pmatrix} s_3(k+1) \\ s_4(k+1) \end{pmatrix} &= \begin{pmatrix} \frac{-1}{4} & \frac{\sqrt{3}}{4} \\ -\frac{\sqrt{3}}{4} & \frac{-1}{4} \end{pmatrix} \begin{pmatrix} s_3(k) \\ s_4(k) \end{pmatrix} + \begin{pmatrix} 1.756 \\ 1.521 \end{pmatrix} y(k) \end{aligned}$$

The control input $u(k)$ is then

$$u(k) = 0.5s_1(k) + 0.8677s_2(k) + 0.8677s_3(k)$$

Another way to get the parallel form is the following. We write the given transfer function as

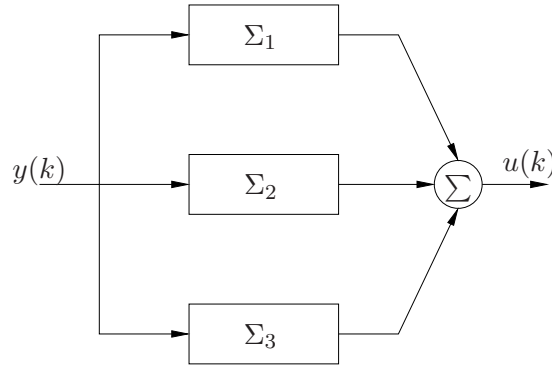


Figure 5.1.1: Parallel form for the controller of Problem 5.1

$$H(z) = \frac{1}{(z-1)(z-1/2)(z^2+1/2z+1/4)} = \frac{A}{z-1} + \frac{B}{z-1/2} + \frac{Cz+D}{z^2+1/2z+1/4}$$

where the constants A, B, C, D are to be determined. It is easy to find that the two polynomials in z are equal if and only if

$$A = 1.14, \quad B = -2.67, \quad D = 0.95, \quad C = 1.52.$$

Then the system is the parallel of

$$\begin{aligned} \Sigma_1 : H_{\Sigma_1} &= \frac{1.14}{z-1} \\ \Sigma_2 : H_{\Sigma_2} &= \frac{-2.67}{z-1/2} \\ \Sigma_3 : H_{\Sigma_3} &= \frac{1.52z+0.95}{z^2+1/2z+1/4} \end{aligned}$$

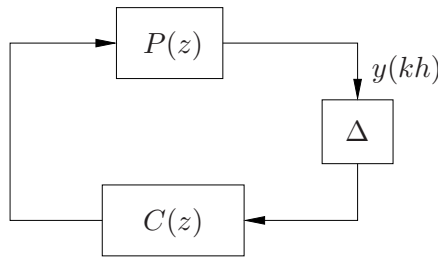


Figure 5.3.1: Closed loop system for Problem 5.3.

SOLUTION 5.2

(a) The closed loop system from r to y shown in Figure has the following transfer function

$$H(s) = \frac{C_s(s)P(s)e^{-s\tau}}{1 + C_s(s)P(s)e^{-s\tau}}.$$

In order to find $C_s(s)$ we need to solve $H(s) = H_{cl}(s)$ which gives

$$\begin{aligned} \frac{C_s(s)P(s)e^{-s\tau}}{1 + C_s(s)P(s)e^{-s\tau}} &= \frac{C(s)P(s)}{1 + C(s)P(s)}e^{-s\tau} \\ C_s(s)(1 + C(s)P(s)) &= C(s)(1 + C_s(s)P(s)e^{-s\tau}) \\ C_s(s) &= \frac{C(s)}{1 + C(s)P(s) - C(s)P(s)e^{-s\tau}} \end{aligned}$$

(b) We first find $C(s)$:

$$\frac{C(s)\frac{1}{s+1}}{1 + C(s)\frac{1}{s+1}}e^{-s\tau} = \frac{8}{s^2 + 4s + 8} \quad \Rightarrow \quad C(s) = \frac{8s + 8}{s^2 + 4s}.$$

Using the result in (a) we get

$$C_s(s) = \frac{8(s+1)}{s^2 + 4s + 8(1 - e^{-s\tau})}$$

SOLUTION 5.3

The control problem over wireless network can be represented as shown Figure 5.3.1. The delay Δ is such that

$$\Delta(y(kh)) = y(kh - d(k)) \quad d(k) \in \{0, \dots, N\}$$

The closed loop system is stable if

$$\left| \frac{P(e^{i\omega})C(e^{i\omega})}{1 + P(e^{i\omega})C(e^{i\omega})} \right| < \frac{1}{N|e^{i\omega} - 1|} \quad \omega \in [0, 2\pi]$$

where N is the number of samples that the control signal is delayed. Notice that the previous result is valid if the closed loop transfer function is stable. In this case the closed loop transfer function is stable with poles

$$z_1 = 1, \quad z_2 = 0.861, \quad z_3 = 0.5, \quad z_4 = 0.447.$$

If we plot the bode diagram of the closed loop system without delay versus the function $1/(N|e^{i\omega} - 1|)$ for different values of N we obtain the results shown in Figure 5.3.2. It can be seen that the closed loop system is stable if $N \leq 3$. Thus the maximum delay is 3 samples. Notice that the result is only a sufficient condition. This means that it might be possible that the system is stable for larger delays than 3 samples.

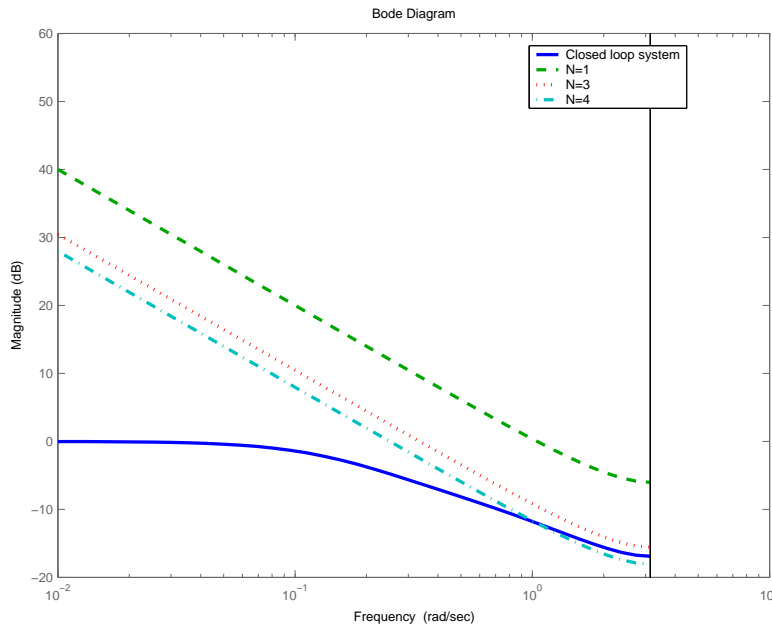


Figure 5.3.2: Bode diagram of the closed loop system of Problem 5.3 and the function $1/(N|e^{i\omega} - 1|)$ for different values of N .

SOLUTION 5.4

The linear model for multiplication with roundoff is shown in Figure 5.4.1. The signal ϵ is a stochastic variable representing the roundoff, and it is uniformly distributed in the interval $(-\delta/2, \delta/2)$. It thus has variance $\delta^2/12$. Before computing the variance of u as function of the variance of i we show that the two algorithms compute

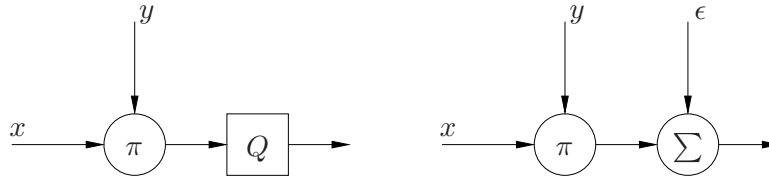


Figure 5.4.1: Linear model for multiplication with roundoff.

the same u . For Algorithm 1 we have

$$u = k(e + i) \xrightarrow{i^+ = i + eh/t_i} u = ke + k(i - 1) + keh/t_i$$

whereas for algorithm 2 we have

$$u = i + ke \xrightarrow{i^+ = ki + keh/t_i} u = ke + k(i - 1) + keh/t_i.$$

Alg 1: We have that

$$\text{Var}\{i^+\} = \text{Var}\{i\} + \text{Var}\left\{\left(eh + \epsilon\right) \frac{1}{t_i} + \epsilon\right\} = \text{Var}\{i\} + \left(1 + \frac{1}{t_i}\right) \frac{\delta^2}{12}.$$

Using the approximation of roundoff we have that $u \approx k(e + i) + \epsilon$ and thus

$$\text{Var}\{u\} = k\text{Var}\{i\} + \frac{\delta^2}{12}.$$

Alg 2: We have that

$$\text{Var}\{i^+\} = k\text{Var}\{i\} + \frac{\delta^2}{12} + \text{Var}\left\{\left((ke + \epsilon)h + \epsilon\right)\frac{1}{t_i} + \epsilon\right\} = k\text{Var}\{i\} + \left(2 + \frac{h}{t_i} + \frac{1}{t_i}\right)\frac{\delta^2}{12}.$$

Using the approximation of roundoff we have that $u \approx i + ke + \epsilon$ and thus

$$\text{Var}\{u\} = \text{Var}\{i\} + \frac{\delta^2}{12}.$$

If we consider

$$\text{Var}\{i^+\} = \alpha\text{Var}\{i\} + f(\delta^2)$$

then after n iteration we have that

$$\text{Var}\{i\}_n = \left(\sum_{i=0}^{n-1} \alpha^i\right) f(\delta^2),$$

where we have considered $\text{Var}\{i\}_0 = 0$. Thus we have that

Alg 1: In this case

$$\text{Var}\{i\}_n = n\left(1 + \frac{1}{t_i}\right)\frac{\delta^2}{12}$$

and thus

$$\text{Var}\{u\}_n = \left(kn + \frac{kn}{t_i} + 1\right)\frac{\delta^2}{12}$$

Alg 2: In this case

$$\text{Var}\{i\}_n = \left(\sum_{i=0}^{n-1} k^i\right)\left(\frac{h}{t_i} + \frac{1}{t_i} + 2\right)\frac{\delta^2}{12}$$

and thus

$$\text{Var}\{u\}_n = \left(\frac{k^n - 1}{k - 1}\right)\left(\frac{h}{t_i} + \frac{1}{t_i} + 2\right)\frac{\delta^2}{12} + \frac{\delta^2}{12}$$

In order to compare the two algorithms, we assume $h = t_i = 1$. (This is just for comparison, since multiplication with 1 normally does not add any roundoff error.) We then have for the two algorithms, that

$$\begin{aligned}\text{Var}\{u\}_n^{(1)} &= (2kn + 1)\frac{\delta^2}{12} \\ \text{Var}\{u\}_n^{(2)} &= \left(4\frac{k^n - 1}{k - 1} + 1\right)\frac{\delta^2}{12}.\end{aligned}$$

If we assume $k < 1$ and n large then

$$4\frac{k^n - 1}{k - 1} + 1 \approx \frac{4}{1 - k} + 1$$

and thus

$$2kn + 1 > \frac{4}{1 - k} + 1$$

which allows to conclude that the first algorithm is worst than the second. If $k > 1$ and n large then

$$4\frac{k^n - 1}{k - 1} + 1 \approx \frac{k^n}{k - 1}$$

and thus

$$2kn(k - 1) + 1 < k^n$$

which allows to conclude that in this case the second algorithm is worst than the first.

SOLUTION 5.5

For $a = 1/8$ the transfer function is

$$H(z) = \frac{1}{1 - \frac{1}{8}z^{-1}} = \frac{y(z)}{r(z)}$$

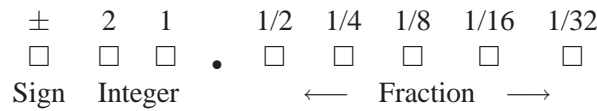
thus in time we have

$$y(k) = \frac{1}{8}y(k-1) + r(k).$$

For a step $r(k) = 1$ for $k \geq 0$ thus

$$y(k) = \frac{1}{8}y(k-1) + 1 \quad k \geq 0.$$

The data representation is the following



The steady state value is

$$\lim_{z \rightarrow 1} \frac{z-1}{z} \frac{1}{1 - \frac{1}{8}z^{-1}} \frac{z}{z-1} = \frac{1}{1 - 0.125} = 1.14285714.$$

If we compute the steady-state value when truncation and roundoff are considered, and we have limited word-length we obtain

iteration	$y(k)$ with truncation	$y(k)$ with roundoff	exact
0	1	1	1.0
1	001.00100b	001.00100b	$1 + \frac{1}{8}$
2	001.00100b	001.00101b	$1 + \frac{1}{8}(1 + \frac{1}{8}) = 1 + \frac{1}{8} + \frac{1}{64} = 1 + \frac{9}{64}$
3	001.00100b	001.00101b	$1 + \frac{1}{8} + \frac{1}{64} + \frac{1}{512} = 1 + \frac{73}{512}$
⋮	⋮	⋮	⋮
∞	001.00100b = 1.125	001.00101b = 1.15625	1.14285714

which then gives the following steady-state values

$$y_{ss}^T = 1.125 \quad \text{Truncation}$$

$$y_{ss}^R = 1.15625 \quad \text{Roundoff}$$

$$y_{ss}^E = 1.14285714 \quad \text{Exact.}$$

If we compute the relative error for the truncation an round-off we have

- Truncation: $\frac{y_{ss}^T - y_{ss}^E}{y_{ss}^E} \approx -1.6\%$
- Roundoff: $\frac{y_{ss}^R - y_{ss}^E}{y_{ss}^E} \approx +1.2\%$

Part II

Event-triggered control

Solutions to real-time operating systems

SOLUTION 6.1

(a) The two tasks are not independent in this particular case however. We first compute the CPU utilization

$$U = \sum_{i=1}^n \frac{C_i}{T_i} = \frac{C_c}{T_c} + \frac{C_a}{T_a}$$

where the period $T_c = T_a = h$. The two tasks needs to be finished within the new sampling interval. Thus we have

$$U = \frac{0.1}{0.4} + \frac{0.2}{0.4} = 0.75$$

thus the CPU is utilized 75% of the time. We still do not know if it is possible to schedule the tasks so that they meet their deadlines. In order to do this we calculate the schedule length and we draw the schedule. The schedule length is

$$\text{lcm}\{T_a, T_c\} = 0.4$$

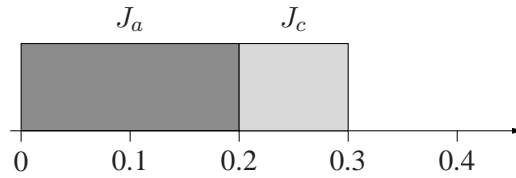


Figure 6.1.1: Schedule for the two tasks J_a and J_c of Problem 6.1.

(b) The utilization factor in this case is

$$U = \frac{0.1}{0.4} + \frac{0.2}{0.4} + \frac{0.2}{0.8} = 1$$

Thus the CPU will be fully occupied. In order to see if the tasks are schedulable we draw the schedule. In this case the schedule length is

$$\text{lcm}\{T_a, T_c, T_x\} = 0.8.$$

Notice that J_x starts after 3 time steps since the release time is $r_x = 0.3$. The worst case response time for the control task J_c is 0.4 as is can be seen from Figure 6.1.2.

SOLUTION 6.2

In a closed loop with a delay we know that the stability is guaranteed if

$$\left| \frac{P(e^{i\omega})C(e^{i\omega})}{1 + P(e^{i\omega})C(e^{i\omega})} \right| < \frac{1}{N|e^{i\omega} - 1|} \quad \omega \in [0, \pi]$$

where $N = \lceil R_c/h \rceil$, with R_c the worst case response time of the control task J_c caused by the high priority task.

From the previous equation we have immediately that

$$N < \frac{|1 + P(e^{i\omega})C(e^{i\omega})|}{|e^{i\omega} - 1||P(e^{i\omega})C(e^{i\omega})|}$$

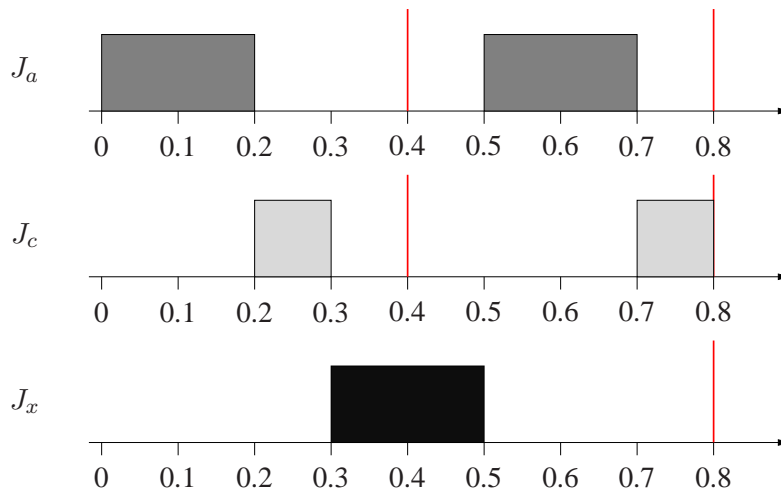


Figure 6.1.2: Schedule for the two tasks J_a , J_c and J_x of Problem 6.1.

and this should hold for all $\omega \in [0, \pi]$. We have that the magnitude of the closed loop transfer function is

$$\left| \frac{P(e^{i\omega})C(e^{i\omega})}{1 + P(e^{i\omega})C(e^{i\omega})} \right| = 3 \frac{|10e^{i\omega} - 1|}{|130e^{i\omega} - 103|}$$

and thus we have that

$$N < \frac{1}{3} \frac{|130e^{i\omega} - 103|}{|(e^{i\omega} - 1)(10e^{i\omega} - 1)|}$$

for all $\omega \in [0, \pi]$. This means that

$$N < \min_{\omega \in [0, \pi]} \frac{1}{3} \frac{|130e^{i\omega} - 103|}{|(e^{i\omega} - 1)(10e^{i\omega} - 1)|}$$

Since the right hand side of the previous inequality can be regarded as the magnitude of a transfer function with one zero and two poles on the positive real axis, it is clear the minimum is at $\omega = \pi$.

Thus we have that

$$N < \frac{1}{3} \frac{|130e^{i\omega} - 103|}{|(e^{i\omega} - 1)(10e^{i\omega} - 1)|} \Bigg|_{\omega=\pi} \approx 3.53.$$

Since N is an integer then we have that $N = 3$. Thus we have that $R_c = 2N = 6$.

We know that the worst case computation time for the control task is $C_c = 1$ with period $T_c = h = 2$. and that the high priority task is not released until time $t = 2$. The worst case response time for the control task is given by

$$R_c = C_c + C_I$$

which than gives $C_I = 5$. The schedule in this case looks as in Figure 6.2.1. As we can notice the control task misses its deadlines at $t = 4$ and $t = 6$. The control task is then delayed and the worst case response time is exactly $R_c = 6$, which represents the maximum delay of the control task.

SOLUTION 6.3

(a) The situation is depicted in Figure 6.3.1. Let us consider the single instances.

- 0) The task J_A is released and its starts its computations. The bus is locked.
- 1) The task J_C requires to be executed. Since it has high priority the CPU preempts the task J_A .

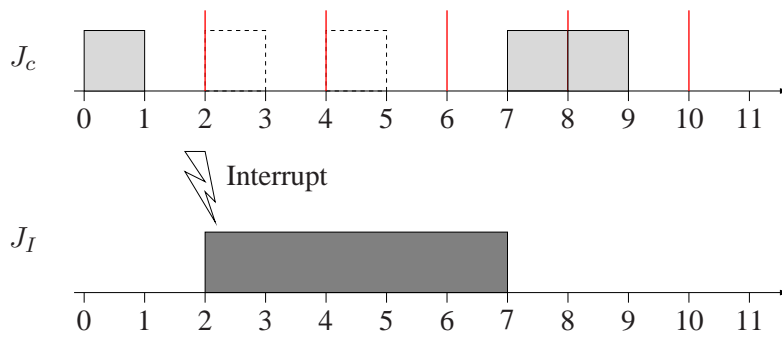


Figure 6.2.1: Schedule for the control task J_c and the task handling the interrupt, of Problem 6.2.

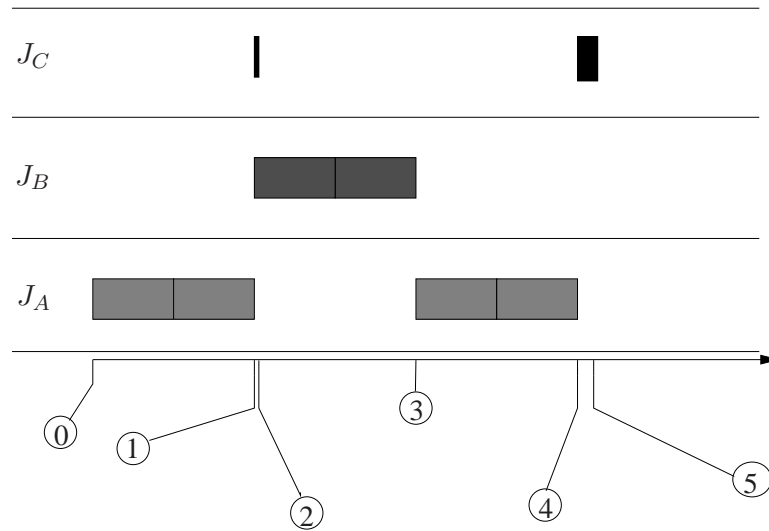


Figure 6.3.1: Priority inversion. Problem 6.3

- 2) Since J_C needs to access the bus, and this resource is occupied by the task J_A the task J_C is stopped. At the same time task J_B asks for CPU time, and since J_C is stopped and it has higher priority of J_A , J_B can be executed. The execution of J_B prevents J_A to release the bus.
- 3) The task J_B with medium priority has finished to be executed and the CPU is given to the task J_A .
- 4) Task J_A finishes to write on the bus, and it releases it.
- 5) Task J_C finally can use the bus and it is scheduled by the CPU.

What we can see from this particular situation is that a high priority task as J_C is blocked by a low priority task. We have a situation called *priority inversion*. The response time for the high priority task J_C in this case is $R_C = 4.1$.

- (b) A possible way to overcome the problem is to use the *priority inheritance protocol*. In this case the task J_A inherits the priority of the task which has higher priority and needs to use the blocked resource. The task J_A , thus acquires high priority and is able to release the resource as soon as possible, so that the high priority task J_C can be served as soon as possible. The priorities are set to the default one, once the bus is released.

SOLUTION 6.4

- (a) Using the EDD algorithm the tasks are scheduled considering the deadlines. Those that have early deadline are scheduled first. From the table given in the problem we see that the order of scheduling

is

$$J_1 \rightarrow J_5 \rightarrow J_3 \rightarrow J_4 \rightarrow J_2.$$

The schedule is shown in Figure 6.4.1. The maximum lateness is equal to -1 due to task J_4 . The fact

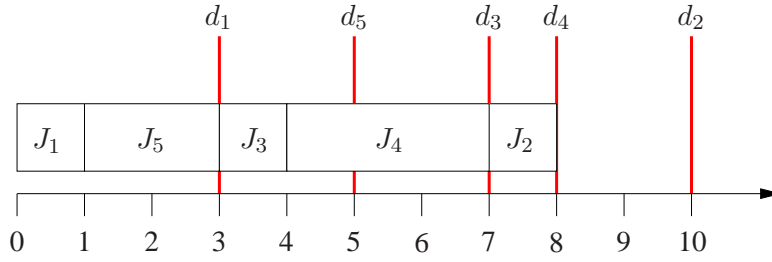


Figure 6.4.1: Jackson's scheduler

that the lateness is negative means that all the tasks meet their deadlines.

- (b) Let σ be a schedule produced by any algorithm A . If A is different from EDD, then there exist two tasks J_a and J_b with $d_a < d_b$ with J_b scheduled before the task J_a . Now, let σ' be a schedule obtained from σ by exchanging J_a with J_b in the schedule σ , so that J_a is scheduled before J_b in σ' . If we exchange J_a with J_b in σ the maximum lateness cannot increase. In fact for the schedule σ we have that the maximum lateness is

$$L_{max_{ab}} = f_a - d_a$$

as can be seen from Figure 6.4.2.

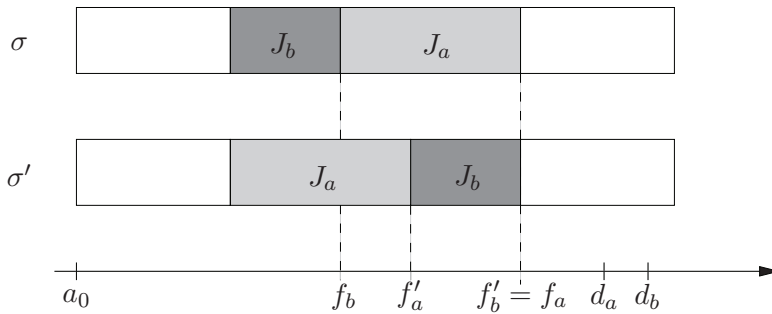


Figure 6.4.2: Jackson's scheduling - proof

In σ' we have that

$$L'_{max_{ab}} = \max(L'_a, L'_b).$$

We can have two possible cases

- $L'_a \geq L'_b$, then $L'_{max_{ab}} = L'_a = f'_a - d_a$, and since $f'_a < f_a$ then we have $L'_{max_{ab}} < L_{max_{ab}}$,
- $L'_a \leq L'_b$, then $L'_{max_{ab}} = L'_b = f'_b - d_b = f_a - d_b$, and since $d_a < d_b$ we have that $L'_{max_{ab}} < L_{max_{ab}}$.

Since in both cases $L'_{max_{ab}} < L_{max_{ab}}$, then we can conclude that interchanging the tasks J_a and J_b in σ we cannot increase the lateness of a set of tasks. By a finite number of such transpositions, σ can be transformed in σ_{EDD} . Since at each step the lateness cannot increase, σ_{EDD} is optimal.

Solutions to real-time scheduling

SOLUTION 7.1

Notice that the four assumptions under which the schedulability test can be use are satisfied. We can remember here the assumptions

- A1 the instances of the task J_i are regularly activated at constant rate,
- A2 all instances of the periodic task J_i have the same worst case execution time C_i ,
- A3 all instances of a periodic task J_i have the same relative deadline, which is equal to the period,
- A4 all tasks are independent i.e., there are no precedence relations and no resource constraints.

The schedulability test gives

$$U = \frac{1}{4} + \frac{2}{6} + \frac{3}{10} = 0.8833 \not\leq 0.7798 = 3(2^{1/3} - 1)$$

thus we cannot conclude if it is possible to schedule the three tasks. We need to use the necessary condition, which ensures that the tasks are schedulable if the worst-case response time is less than the deadline. The priority is assigned so that J_1 has highest priority, J_2 medium priority and J_3 lowest priority. The analysis yields

$$R_1 = C_1 = 1 \leq D_1 = 4$$

for task J_2 we have

$$\begin{aligned} R_2^0 &= C_2 = 2 \\ R_2^1 &= C_2 + \left\lceil \frac{R_2^0}{T_1} \right\rceil C_1 = 2 + 1 = 3 \\ R_2^2 &= C_2 + \left\lceil \frac{R_2^1}{T_1} \right\rceil C_1 = 2 + 1 = 3 \leq D_2 = 6 \end{aligned}$$

for task J_3 we have

$$\begin{aligned} R_3^0 &= C_3 = 3 \\ R_3^1 &= C_3 + \left\lceil \frac{R_3^0}{T_1} \right\rceil C_1 + \left\lceil \frac{R_3^0}{T_2} \right\rceil C_2 = 3 + 2 + 1 = 6 \\ R_3^2 &= C_3 + \left\lceil \frac{R_3^1}{T_1} \right\rceil C_1 + \left\lceil \frac{R_3^1}{T_2} \right\rceil C_2 = 3 + 2 + 2 = 7 \\ R_3^3 &= C_3 + \left\lceil \frac{R_3^2}{T_1} \right\rceil C_1 + \left\lceil \frac{R_3^2}{T_2} \right\rceil C_2 = 3 + 2 + 2 = 9 \\ R_3^4 &= C_3 + \left\lceil \frac{R_3^3}{T_1} \right\rceil C_1 + \left\lceil \frac{R_3^3}{T_2} \right\rceil C_2 = 3 + 3 + 4 = 10 \\ R_3^5 &= C_3 + \left\lceil \frac{R_3^4}{T_1} \right\rceil C_1 + \left\lceil \frac{R_3^4}{T_2} \right\rceil C_2 = 3 + 3 + 4 = 10 \leq D_3 = 10 \end{aligned}$$

Thus the three tasks are schedulable with rate monotonic. The schedule length is given by

$$\text{lcm}(T_1, T_2, T_3) = 60$$

The schedule is shown in figure 7.1.1.

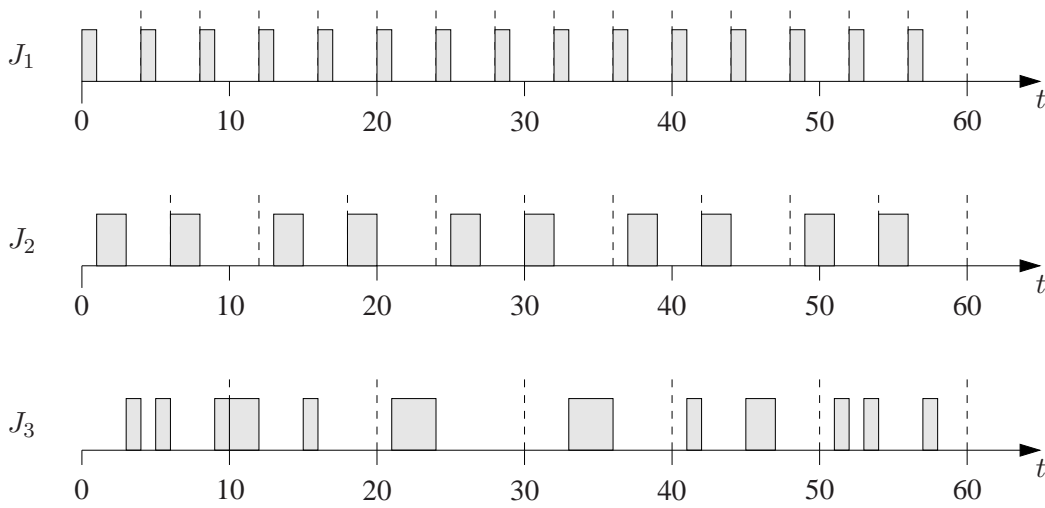


Figure 7.1.1: Rate monotonic scheduling for the tasks J_1 , J_2 and J_3 , in Problem 7.1

SOLUTION 7.2

The CPU utilization is

$$U = \frac{1}{4} + \frac{2}{6} + \frac{3}{10} = 0.8833 \leq 1$$

thus the schedulability test for EDF ensures that the three tasks are schedulable. The schedule length, as in Problem 7.1, is

$$\text{lcm}(T_1, T_2, T_3) = 60.$$

The schedule is shown in figure 7.2.1.

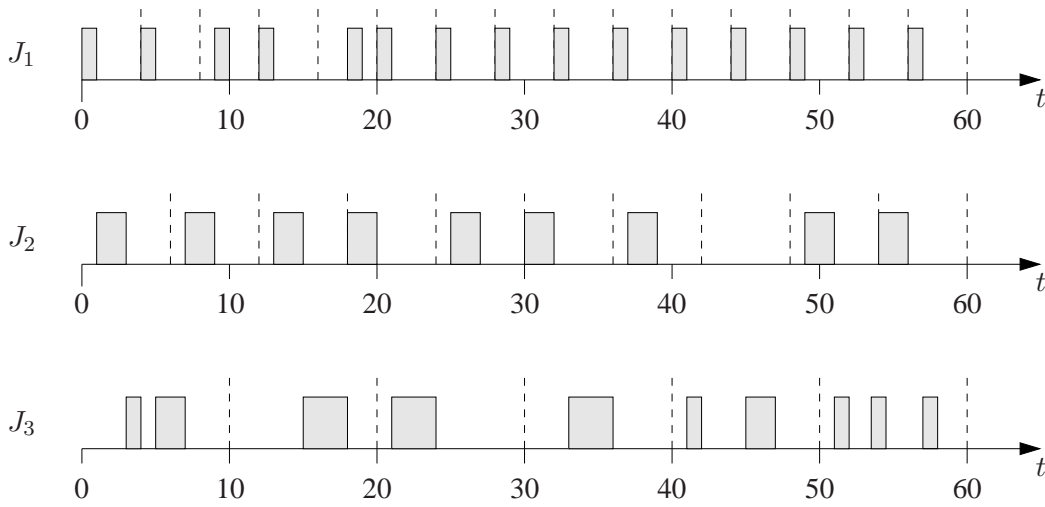


Figure 7.2.1: EDF scheduling for the tasks J_1 , J_2 and J_3 , in Problem 7.2

SOLUTION 7.3

The CPU utilization is

$$U = \sum_{i=1}^3 \frac{C_i}{T_i} = \frac{C_1}{T_1} + \frac{C_2}{T_2} + \frac{C_3}{T_3} = \frac{1}{3} + \frac{2}{4} + \frac{1}{7} = 0.9762 \not\leq 3(2^{1/3} - 1) = 0.7798$$

Let us consider the first 10 steps of the RM schedule. The RM algorithm assigns to J_1 the highest priority, to J_2 middle priority and to J_3 the lowest. The schedule length is

$$\text{lcm}(3, 4, 7) = 84$$

but we draw here only the first 9 steps, as shown in Figure 7.3.1. We can notice that the task J_3 misses its deadline at 7 and it is scheduled in the next time slot. Thus the set of tasks is not schedulable with RM.

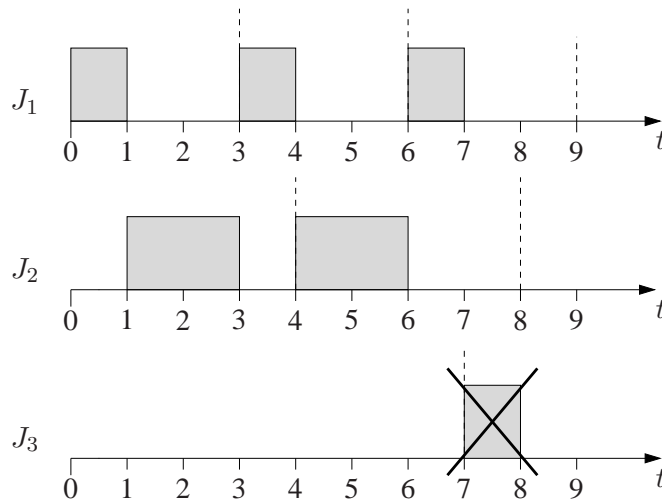


Figure 7.3.1: RM scheduling for the tasks J_1 , J_2 and J_3 , in Problem 7.3. Notice that the task J_3 misses its deadline.

Since the utilization factor U is less than one, then we can use the EDF algorithm to schedule the task. The schedule is shown in Figure 7.3.2.

SOLUTION 7.4

(a) The CPU utilization is

$$U = \frac{1}{4} + \frac{2}{5} + \frac{3}{10} = 0.95$$

The RM algorithm assigns the highest priority to J_1 , middle to J_2 and the lowest to J_3 . The schedule length is given by

$$\text{lcm}(4, 5, 10) = 20.$$

The RM scheduling algorithm assigns the highest priority to the control task J_1 which then has worst-case response time of $R_1 = 1$. In the case of the EDF we need to draw the schedule in order to compute R_1 . The schedule is shown in Figure 7.4.1. The EDF gives a worst-case response time of $R_1 = 2$ as is shown in Figure 7.4.1. With the EDF algorithm is possible to schedule tasks that are not schedulable with RM, but we do not have anymore "control" on the response time of some tasks.

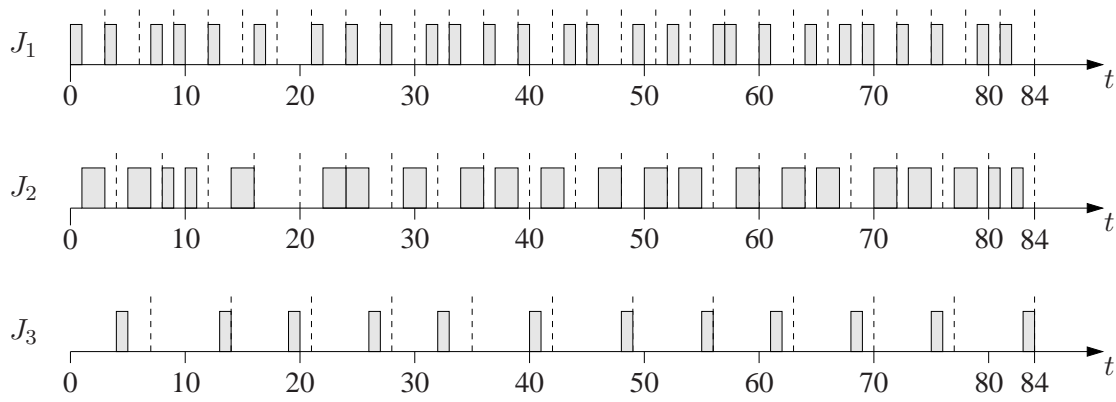


Figure 7.3.2: EDF scheduling for the tasks J_1 , J_2 and J_3 , in Problem 7.3. Notice that all the tasks meet their deadlines.

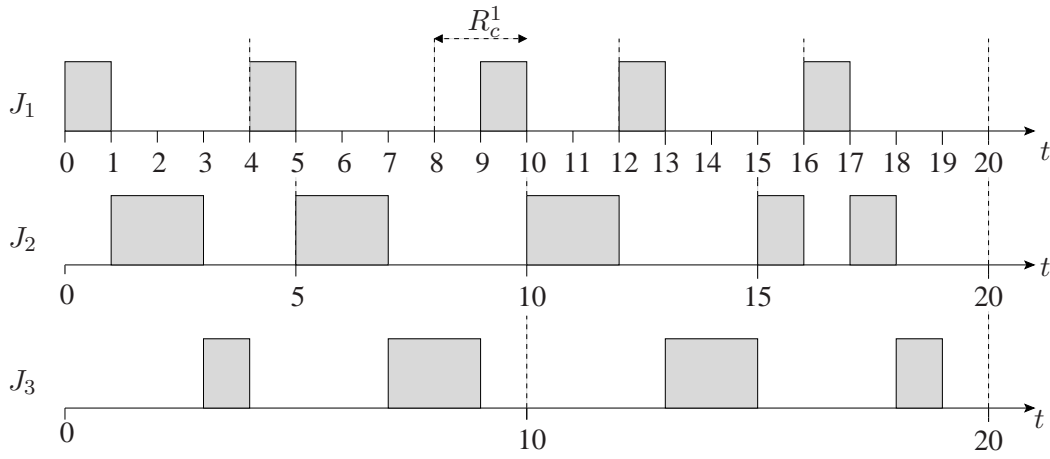


Figure 7.4.1: EDF scheduling for the tasks J_1 , J_2 and J_3 in Problem 7.4.

- (b) In order to have $R_2 = 1$ we need to assign the highest priority to the task 2. We then give to J_1 middle priority and J_3 the lowest. The tasks meet their deadline if the worst-case response time is less than the deadline. In this case we have

$$R_2 = C_2 = 2 \leq D_1 = 5$$

for task J_1 we have

$$R_1^0 = C_1 = 1$$

$$R_1^1 = C_1 + \left\lceil \frac{R_1^0}{T_2} \right\rceil C_2 = 1 + 2 = 3$$

$$R_1^2 = C_1 + \left\lceil \frac{R_1^1}{T_2} \right\rceil C_2 = 1 + 2 = 3 \leq D_1 = 4$$

for task J_3 we have

$$\begin{aligned}
 R_3^0 &= C_3 = 3 \\
 R_3^1 &= C_3 + \left\lceil \frac{R_3^0}{T_1} \right\rceil C_1 + \left\lceil \frac{R_3^0}{T_2} \right\rceil C_2 = 3 + 1 + 2 = 6 \\
 R_3^2 &= C_3 + \left\lceil \frac{R_3^1}{T_1} \right\rceil C_1 + \left\lceil \frac{R_3^1}{T_2} \right\rceil C_2 = 3 + 2 + 4 = 9 \\
 R_3^3 &= C_3 + \left\lceil \frac{R_3^2}{T_1} \right\rceil C_1 + \left\lceil \frac{R_3^2}{T_2} \right\rceil C_2 = 3 + 3 + 4 = 10 \\
 R_3^4 &= C_3 + \left\lceil \frac{R_3^3}{T_1} \right\rceil C_1 + \left\lceil \frac{R_3^3}{T_2} \right\rceil C_2 = 3 + 3 + 4 = 10 \leq D_3 = 10
 \end{aligned}$$

Thus the three tasks are schedulable. The schedule is shown in Figure 7.4.2

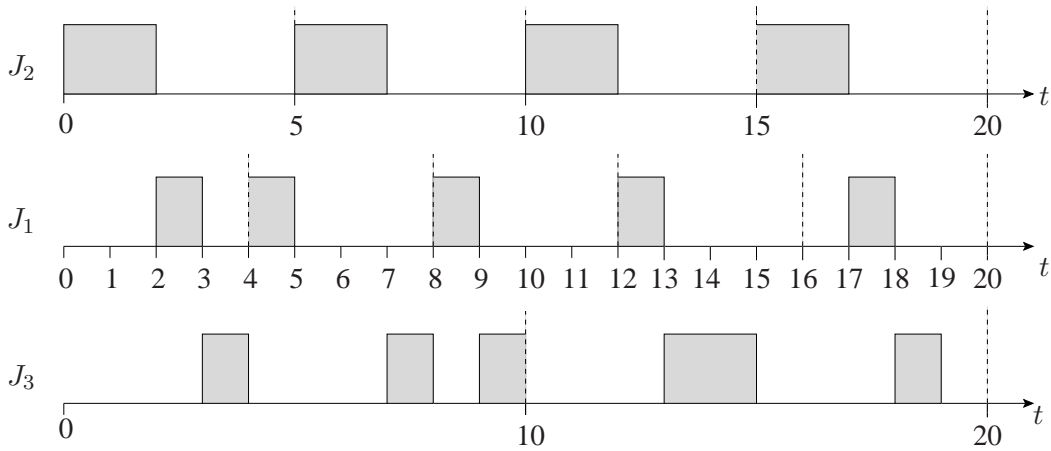


Figure 7.4.2: Fixed-priority scheduling for the tasks J_1 , J_2 and J_3 , in Problem 7.4. J_2 in this case has the highest priority.

SOLUTION 7.5

- (a) The tasks will meet their deadlines since the schedulability condition for RM scheduling is fulfilled:

$$U = \sum_{i=1}^2 \frac{C_i}{T_i} = \frac{C_1}{T_1} + \frac{C_2}{T_2} = \frac{7}{12} < 2(2^{1/2} - 1) \approx 0.8$$

The schedule length is equal to $\text{lcm}(T_1, T_2) = 12$. The time evolution is shown in Figure 7.5.1.

- (b) The worst-case response time for the higher priority tasks is given by the worst-case computation time. Thus $R_1 = C_1 = 1$. For the second task we the worst-case response time is computed by

$$R_2 = C_2 + \left\lceil \frac{R_2}{T_1} \right\rceil C_1.$$

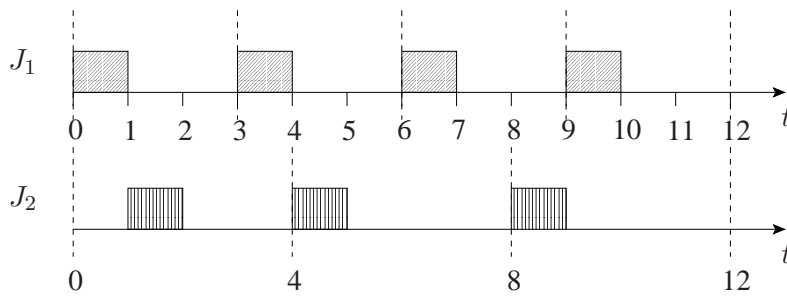


Figure 7.5.1: Rate monotonic scheduling for the tasks J_1 and J_2 , in Problem 7.5

In order to solve the equation we need to use an iterative procedure. Let $R_2^0 = 0$, then

$$R_2^1 = C_2 + \left\lceil \frac{R_2^0}{T_1} \right\rceil C_1 = C_2 = 1$$

$$R_2^2 = C_2 + \left\lceil \frac{R_2^1}{T_1} \right\rceil C_1 = C_2 + C_1 = 2$$

$$R_2^3 = C_2 + \left\lceil \frac{R_2^2}{T_1} \right\rceil C_1 = C_2 + C_1 = 2.$$

Thus $R_2 = 2$. This agrees with the time evolution plotted in (a).

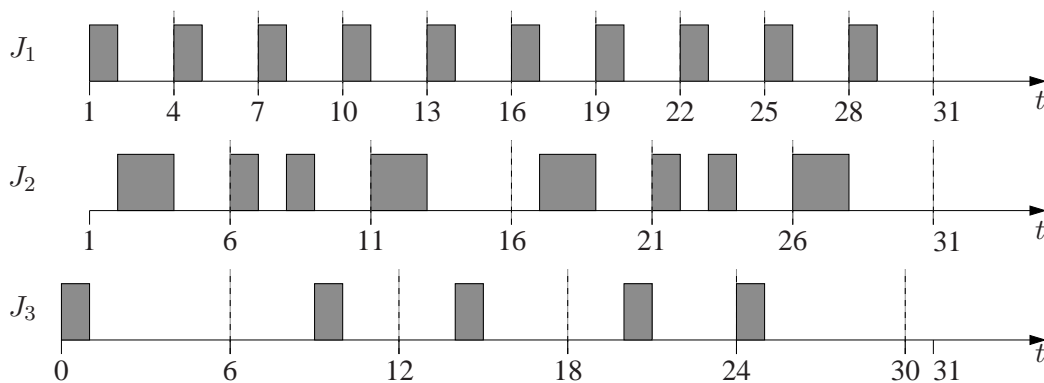
SOLUTION 7.6

Since the tasks have an offset O_i we cannot use the response time formula for rate-monotonic since it assumes that all tasks are released simultaneously at time 0. We need to solve the problem drawing the schedule.

(a) In order to draw the schedule for the three tasks we need to compute the schedule length. The schedule length is the least common multiple (l.c.m.) of the task periods plus the offset. The l.c.m. is 30, thus the schedule length is $30+1=31$.

(b) *Rate-monotonic*

The rate-monotonic assigns the priority so that J_1 has the highest priority, J_2 medium priority and J_3 the lowest.

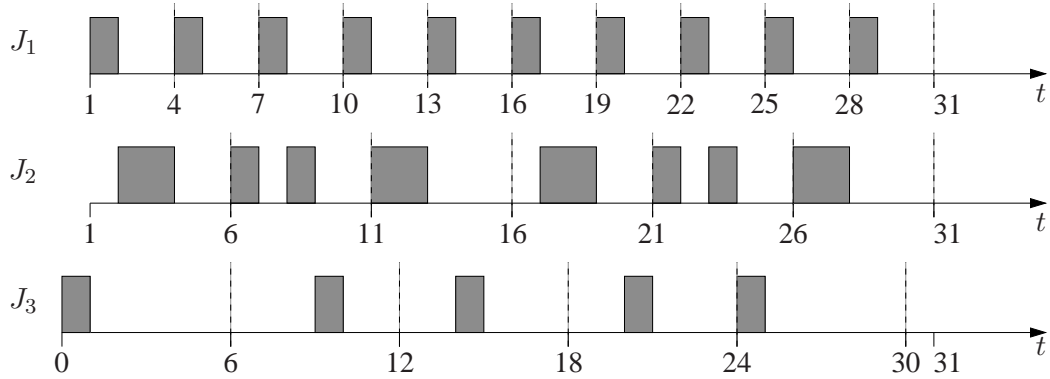


Thus the worst-case response time for task J_2 is $R_2 = 3$.

(c) *Deadline-monotonic*

The deadlines agree by assumption with the periods, so deadline-monotonic scheduling is identical to rate-monotonic scheduling. Thus the worst-case response time for task J_2 is $R_2 = 3$.

- (d) *Earliest-deadline-first*
The schedule is shown below.



The worst-case response time for J_2 is $R_2 = 3$.

SOLUTION 7.7

- (a) The utilization factor

$$U = \sum_{i=1}^3 \frac{C_i}{T_i} = \frac{C_1}{T_1} + \frac{C_2}{T_2} + \frac{C_3}{T_3} = \frac{1}{4} + \frac{2}{5} + \frac{1}{6} \approx 0.82 \not\leq 3(2^{1/3} - 1) \approx 0.76.$$

so we derive the worst-case response times for all tasks. The priority assignment is such that J_1 has the highest priority, J_c intermediate priority and J_2 the lowest priority. Thus the analysis yields

$$\begin{aligned} \Rightarrow R_1 &= C_1 = 1 \\ R_c^0 &= C_c = 2 \\ R_c^1 &= C_c + \left\lceil \frac{R_c^0}{T_1} \right\rceil C_1 = 3 \\ R_c^2 &= C_c + \left\lceil \frac{R_c^1}{T_1} \right\rceil C_1 = 3 \\ \Rightarrow R_c &= 3 \leq D_c = 5 \\ R_2^0 &= C_2 = 2 \\ R_2^1 &= C_2 + \left\lceil \frac{R_2^0}{T_1} \right\rceil C_1 + \left\lceil \frac{R_2^0}{T_c} \right\rceil C_c = 4 \\ R_2^2 &= C_2 + \left\lceil \frac{R_2^1}{T_1} \right\rceil C_1 + \left\lceil \frac{R_2^1}{T_c} \right\rceil C_c = 4 \\ \Rightarrow R_2 &= 4 \leq D_2 = 6. \end{aligned}$$

Thus the set of task is schedulable with the ate monotonic algorithm, with the worst-case response time for the control task equal to $R_c = 3$.

- (b) Since $\tau = R_c = 3 > h = 2$ then the sampled version controller in state space from is

$$\begin{aligned} x(kh + h) &= \Phi x(kh) + \Gamma_0 y(kh - (d-1)h) + \Gamma_1 y(kh - dh) \\ u(kh) &= Cx(kh) \end{aligned}$$

where $\tau = (d - 1)h + \tau'$ with $\tau' \leq h$ and $d \in \mathbb{N}$. Thus if we choose $d = 2$ we get $\tau' = 1$. The state space description becomes

$$\begin{pmatrix} x(kh + h) \\ y(kh - h) \\ y(kh) \end{pmatrix} = \begin{pmatrix} \Phi & \Gamma_1 & \Gamma_0 \\ 0 & 0 & I \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x(kh) \\ y(kh - 2h) \\ y(kh - h) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ I \end{pmatrix} y(kh)$$

$$u(kh) = (C \ 0 \ 0) \begin{pmatrix} x(kh) \\ y(kh - 2h) \\ y(kh - h) \end{pmatrix}$$

A sketch of a computer program is given below. Notice that we assume that initial values of $x(0)$, $y(-2h)$, $y(-h)$ are known.

Computer Code

```

y=array{y(-1),y(-2)};
nexttime = getCurrentTime();
k = 0;
while true do
  new_y=AD_conversion();
  x(k) = Φx(k - 1) + Γ1y(k - 3) + Γ0y(k - 2);
  u(k) = Cx(k);
  DA_conversion();
  y(k) =new_y;
  k = k + 1;
  nexttime = nexttime + h;
  sleepUntil(nexttime);
end while

```

- (c) The set of tasks will be schedulable if the worst-case response time, R_i is less than the D_i for $i \in 1, 2, c$. In this case the control task has the highest priority, J_1 intermediate priority and J_2 the lowest priority. The analysis yields

$$\begin{aligned} \Rightarrow R_c &= C_c = 2 \\ R_1^0 &= C_1 = 1 \\ R_1^1 &= C_1 + \left\lceil \frac{R_1^0}{T_c} \right\rceil C_c = 3 \\ R_1^2 &= C_1 + \left\lceil \frac{R_1^1}{T_c} \right\rceil C_c = 3 \\ \Rightarrow R_1 &= 3 \leq D_1 = 4 \\ R_2^0 &= C_2 = 1 \\ R_2^1 &= C_2 + \left\lceil \frac{R_2^0}{T_c} \right\rceil C_c + \left\lceil \frac{R_2^0}{T_1} \right\rceil C_1 = 4 \\ R_2^2 &= C_2 + \left\lceil \frac{R_2^1}{T_c} \right\rceil C_c + \left\lceil \frac{R_2^1}{T_1} \right\rceil C_1 = 4 \\ \Rightarrow R_2 &= 4 \leq D_2 = 6. \end{aligned}$$

Thus the tasks are schedulable even if the task J_c is a high priority task. In doing this the delay τ is then equal to $R_c = 2$.

SOLUTION 7.8

The schedulability of periodic tasks can be guaranteed by evaluating the interference by the polling server on periodic execution. In the worst case, such an interference is the same as the one introduced by an equivalent periodic task having a period equal to T_s and computation time equal to C_s . Independently of the number of aperiodic tasks handled by the server, a maximum time equal to C_s is dedicated to aperiodic requests at each server period. The utilization factor of the polling server is $U_s = C_s/T_s$ and hence the schedulability of a periodic set with n tasks and utilization U_p can be guaranteed if

$$U_p + U_s \leq (n + 1)(2^{1/(n+1)} - 1)$$

Thus we need to have

$$U_s \leq (n + 1)(2^{1/(n+1)} - 1) - U_p = (n + 1)(2^{1/(n+1)} - 1) - \sum_{i=1}^n \frac{C_i}{T_i}$$

which in this case is

$$U_s \leq 3(2^{1/3} - 1) - \frac{1}{5} - \frac{2}{8} \approx 0.3298 \quad \Rightarrow \quad U_s^{max} = 0.3298$$

SOLUTION 7.9

The tasks and the polling server can be scheduled since we computed the maximum utilization as

$$U = \sum_{i=1}^2 \frac{C_i}{T_i} + \frac{C_s}{T_s} = \frac{1}{4} + \frac{2}{8} + \frac{1}{5} = 0.77 \leq 3(2^{1/3} - 1) = 0.78$$

The schedule is shown in Figure 7.9.1

SOLUTION 7.10

The polling server has period $T_s = 6$ thus has lower priority with respect to J_1 and J_2 . The time evolution is given in the following Figure 7.10.1.

Note that the server is preempted both by J_1 and J_2 , so the aperiodic task is finished to be served only after one period of the server i.e., $T_s = 6$. The server has lower priority, so it runs only after J_1 and J_2 have terminated. At time $t = 0$ the server is ready to be executed, but since no aperiodic tasks are waiting the server is suspended. At $t = 3$ the aperiodic task is requiring the CPU but it will need to wait until the polling server is reactivated i.e., until $t = T_s = 6$. At this time the server is ready to handle the request of the aperiodic task, but the J_1 and J_2 with higher priority preempt the server.

Solutions to models of computation I: Discrete-event systems

SOLUTION 8.1

Let us define the following event alphabet

$$E = \{s_1, \bar{s}_1, s_2, \bar{s}_2, s_3, \bar{s}_3\}$$

where

$$\begin{aligned} s_i & \text{ is triggered when } S_i \text{ becomes } 1 \\ \bar{s}_i & \text{ is triggered when } S_i \text{ becomes } 0 \quad i = 1, 2, 3. \end{aligned}$$

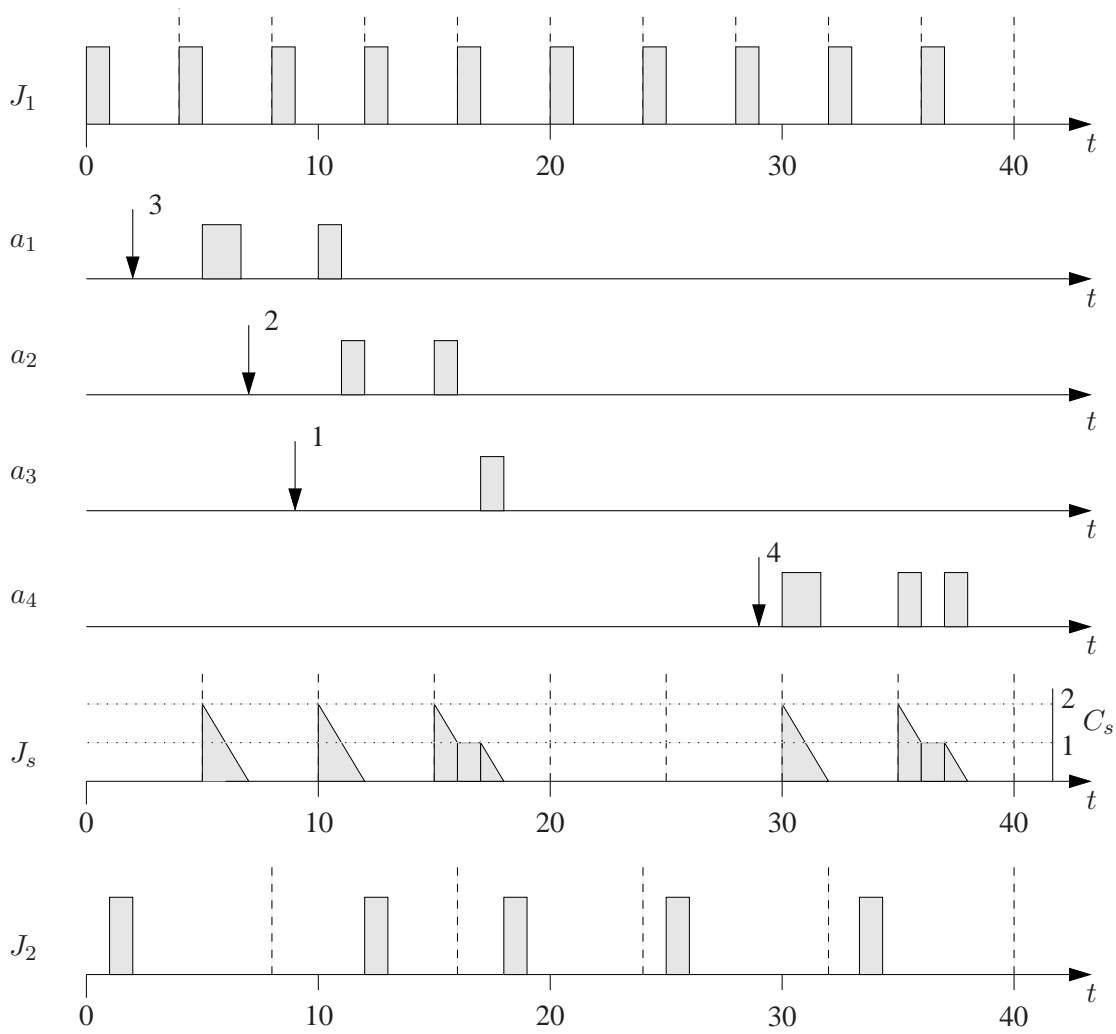


Figure 7.9.1: Polling server of Problem 7.9

We consider the following four possible states for the gate

$$Q = \{O, C, R, L\}$$

where O stands for 'opened', C for 'closed', R for 'raising' and L for 'lowering'. The transition function is defined as following

$$\begin{aligned} \delta(O, s_1) &= L \\ \delta(O, \bar{s}_1) &= O \\ \delta(L, \bar{s}_2) &= C \\ \delta(L, s_2) &= L \\ \delta(C, \bar{s}_3) &= C \\ \delta(C, s_3) &= R \\ \delta(R, s_2) &= O \\ \delta(R, \bar{s}_2) &= R \\ \delta(R, s_1) &= L. \end{aligned}$$

We defined as initial state the state O , and as set of final states

$$Q_m = \{O\}.$$

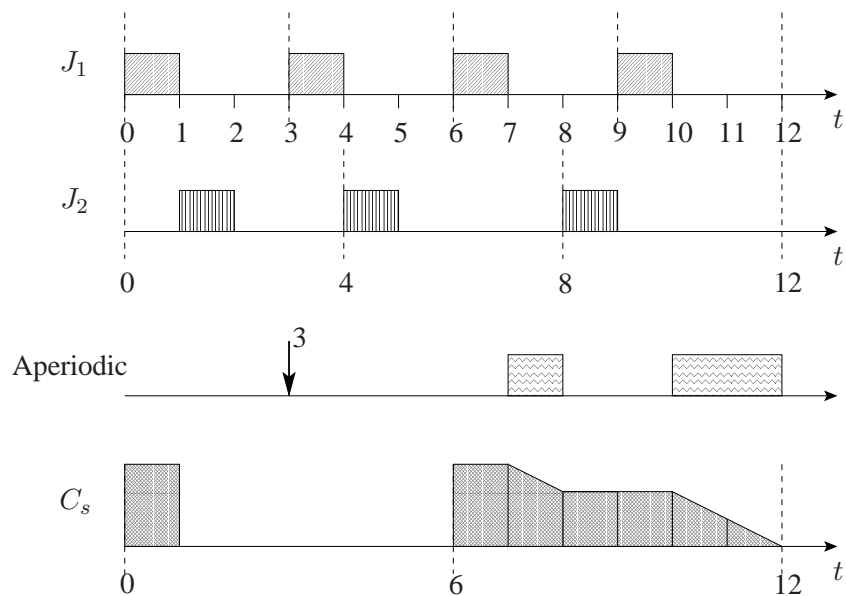


Figure 7.10.1: Polling server of Problem 7.5

Thus the automaton is

$$A = (Q, E, \delta, O, O)$$

where Q , E and δ are defined as before. In Figure 8.1.1 is shown the automaton.

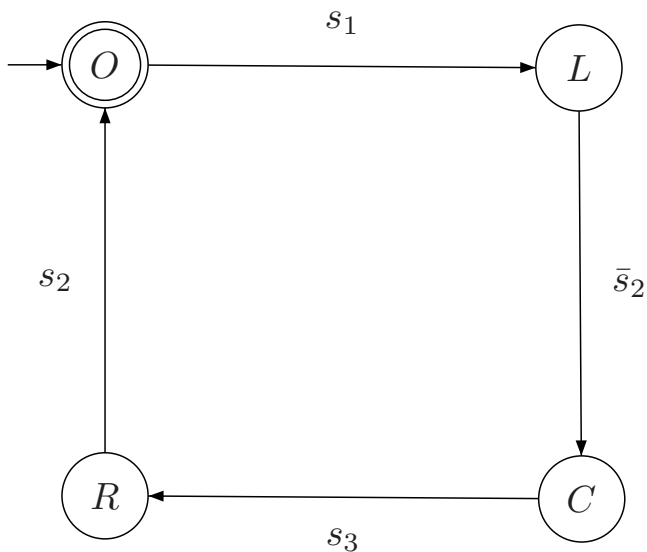


Figure 8.1.1: Control of a gate. Problem 8.1.

SOLUTION 8.2

We consider the following automaton as model of the vending machine

$$A = (Q, E, \delta, q_0, Q_m)$$

where we have the state-space

$$Q = \{0, 10, 20, 30, 40, 25, 35, 45, O\}$$

where the state O denote 'overfull'. The event alphabet is

$$E = \{\mathbb{D}, \mathbb{Q}\}.$$

The initial state is $q_0 = 0$ and the set of final states is $Q_m = \{45\}$. The transition map is

$$\begin{aligned} \delta(0, \mathbb{D}) &= 10 \\ \delta(0, \mathbb{Q}) &= 25 \\ \delta(10, \mathbb{D}) &= 20 \\ \delta(10, \mathbb{Q}) &= 35 \\ \delta(20, \mathbb{D}) &= 30 \\ \delta(20, \mathbb{Q}) &= 45 \\ \delta(25, \mathbb{D}) &= 35 \\ \delta(25, \mathbb{Q}) &= O \\ \delta(30, \mathbb{D}) &= 40 \\ \delta(30, \mathbb{Q}) &= O \\ \delta(35, \mathbb{D}) &= 45 \\ \delta(35, \mathbb{Q}) &= O \\ \delta(40, \mathbb{Q}) &= O \\ \delta(40, \mathbb{D}) &= O. \end{aligned}$$

The automaton is shown in Figure 8.2.1. We can compute the marked language by inspection

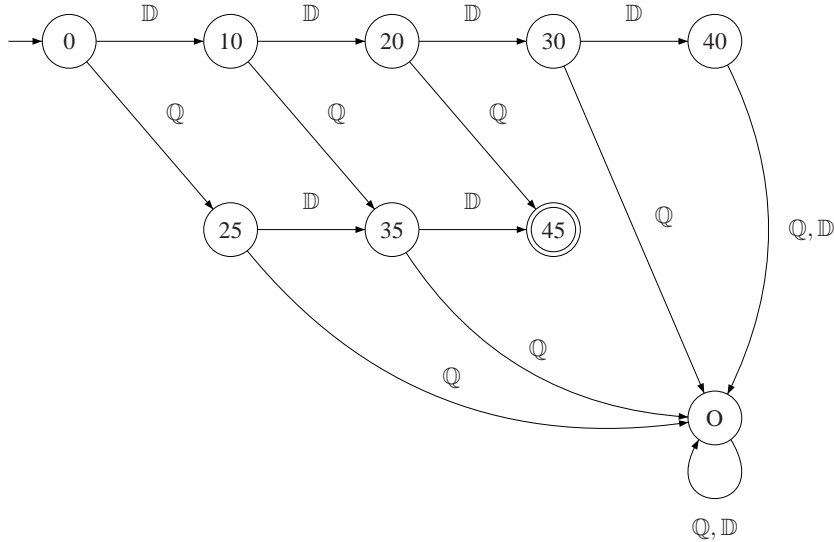


Figure 8.2.1: Automaton A of Problem 8.2.

$$L_m(A) = \{\mathbb{D}\mathbb{D}\mathbb{Q}, \mathbb{D}\mathbb{Q}\mathbb{D}, \mathbb{Q}\mathbb{D}\mathbb{D}\}.$$

We notice directly from Figure 8.2.1 that the machine will not dispense soda if the wrong amount of money is inserted. In that case we can reach the states 30 or 40 or O from where is not possible to reach the state 45 or if we reach the states 25 or 35 and we overpay we will not get a soda. Notice that we can continue to insert coins and the soda will never be dispensed i.e., we have a livelock.

In order to prove it formally we can notice that

$$\mathbb{D}\mathbb{D}\mathbb{D} \in L(A)$$

but clearly

$$\mathbb{D}\mathbb{D}\mathbb{D} \notin L_m(A).$$

Thus the automaton blocks.

SOLUTION 8.3

In order to compute the marked language and generated language we consider the following notation. Let R_{ij}^k be the set of all strings x such that $\delta(q_i, x) = q_j$, and if $\delta(q_i, y) = q_\ell$, for any y that is prefix (initial segment) of x or ϵ , then $\ell \leq k$. That is, R_{ij}^k is the set of all strings that take the finite automaton from state q_i to state q_j without going through any state numbered higher than k . Note that by “going through a state” we mean both entering and leaving that state. Thus i or j may be greater than k . It is possible to define R_{ij}^k recursively

$$R_{ij}^k = R_{ik}^{k-1} (R_{kk}^{k-1}) R_{kj}^{k-1} \cup R_{ij}^{k-1}$$

and

$$R_{ij}^0 = \begin{cases} \{a | \delta(q_i, a) = q_j\} & \text{if } i = j \\ \{a | \delta(q_i, a) = q_j\} & \text{if } i \neq j \end{cases}$$

We then have that

$$L_m(A) = \bigcup_{q_j \in Q_m} R_{0j}^n$$

and

$$L(A) = \bigcup_{q_i \in Q} R_{0i}^n.$$

A set of strings R_{ij}^k can be in general represented by the *regular expressions* r_{ij}^k , which allows a more compact notation.

The automaton is formally defined by the following five-tuple

$$A = (Q, E, \delta, q_0, Q_m)$$

where

$$Q = \{q_1, q_2\}$$

$$E = \{0, 1\}$$

$$\delta(q_1, 0) = q_2$$

$$\delta(q_1, 1) = q_1$$

$$\delta(q_2, 0) = q_2$$

$$\delta(q_2, 1) = q_1$$

$$q_0 = q_1$$

$$Q_m = \{q_2\}$$

The first column of the table is computed by inspection can compute the r_{ij}^0 ,

	k=0	k=1
r_{11}^k	ϵ	ϵ
r_{12}^k	0	0
r_{21}^k	1	1
r_{22}^k	ϵ	10

The second column is computed using the recursive formula introduced before. In particular we have

$$\begin{aligned} r_{11}^1 &= r_{11}^0 (r_{11}^0)^* r_{11}^0 + r_{11}^0 = \epsilon(\epsilon)^* \epsilon + \epsilon = \epsilon \\ r_{12}^1 &= r_{11}^0 (r_{11}^0)^* r_{12}^0 + r_{12}^0 = \epsilon(\epsilon)^* 0 + 0 = 0 \\ r_{21}^1 &= r_{21}^0 (r_{11}^0)^* r_{11}^0 + r_{21}^0 = 1(\epsilon)^* \epsilon + 1 = 1 \\ r_{22}^1 &= r_{21}^0 (r_{11}^0)^* r_{12}^0 + r_{22}^0 = 1(\epsilon)^* 0 + \epsilon = 10 \end{aligned}$$

The marked language is then

$$L_m(A) = r_{12}^2 = r_{12}^1 (r_{22}^1)^* r_{22}^1 + r_{12}^1 = 0(10)^* 10 + 0 = 0(10)^*$$

In this case the generated language is

$$L(A) = r_{11}^2 + r_{12}^2 = 0(10)^* 1 + \epsilon + 0(10)^*$$

The prefix closure $\overline{L_m(A)}$ is

$$\overline{L_m(A)} = \{s \in E^* \mid \exists t \in E^*, st \in L_m(A)\}.$$

We prove that $\overline{L_m(A)} = L(A)$.

Prove $\overline{L_m(A)} \subseteq L(A)$: Let $x \in \overline{L_m(A)}$ then is trivial to see that $x \in L(A)$.

Prove $\overline{L_m(A)} \supseteq L(A)$: Let now $x \in L(A)$. Then $x = \{\epsilon, 0, 010, 01010, 0101010, \dots, 01, 0101, 010101, 01010101, \dots\}$.
The strings in $L_m(A)$ are $0, 010, 01010, 0101010, \dots$ thus prefixes are $\{\epsilon, 0, 01, 0101, 01010, 010101, \dots\}$
and as we can see then $x \in \overline{L_m(A)}$.

We can conclude that $\overline{L_m(A)} = L(A)$, thus the DES is nonblocking.

SOLUTION 8.4

In this case the automaton is

$$A = (Q, E, \delta, q_0, Q_m)$$

where

$$\begin{aligned} Q &= \{q_1, q_2, q_3\} \\ E &= \{0, 1\} \\ \delta(q_1, 0) &= q_2 \\ \delta(q_1, 1) &= q_3 \\ \delta(q_2, 0) &= q_1 \\ \delta(q_2, 1) &= q_3 \\ \delta(q_3, 0) &= q_2 \\ \delta(q_3, 1) &= q_2 \\ q_0 &= q_1 \\ Q_m &= \{q_2, q_3\} \end{aligned}$$

We apply the recursive formula introduced in the solution of Problem 8.3 to compute the marked language and the generated language.

	k=0	k=1	k=2
r_{11}^k	ϵ	ϵ	$(00)^*$
r_{12}^k	0	0	$0(00)^*$
r_{13}^k	1	1	0^*1
r_{21}^k	0	0	$0(00)^*$
r_{22}^k	ϵ	$\epsilon+00$	$(00)^*$
r_{23}^k	1	$1+01$	0^*1
r_{31}^k	\emptyset	\emptyset	$(0+1)(00)^*0$
r_{32}^k	$0+1$	$0+1$	$(0+1)(00)^*$
r_{33}^k	ϵ	ϵ	$\epsilon + (0+1)0^*1$

Certain equivalences among regular expressions has been used to simplify the expressions. For example

$$r_{22}^1 = r_{21}^0 (r_{11}^0)^* r_{12}^0 + r_{22}^0 = 0(\epsilon)^*0 + \epsilon = \epsilon + 00$$

or, for example for r_{13}^2 we have

$$r_{13}^2 = r_{12}^1 (r_{22}^1)^* r_{23}^1 + r_{13}^1 = 0(\epsilon + 00)^*(1 + 01) + 1$$

Since $(\epsilon + 00)^* = (00)^*$ and $(1 + 01) = (\epsilon + 0)1$, the expression becomes

$$r_{13}^2 = 0(00)^*(\epsilon + 0) + 1 + 1$$

and since $(00)^*(\epsilon + 0) = 0^*$ the expression reduces to

$$r_{13}^2 = 00^*1 + 1 = 0^*1.$$

We have that

$$L_m(A) = r_{12}^3 + r_{13}^3$$

where

$$\begin{aligned} r_{12}^3 &= r_{13}^2 (r_{33}^2)^* r_{32}^2 + r_{12}^2 = 0^*1(\epsilon + (0+1)0^*1)^*(0+1)(00)^* + 0(00)^* \\ &= 0^*1((0+1)0^*1)^*(0+1)(00)^* + 0(00)^* \end{aligned}$$

and

$$\begin{aligned} r_{13}^3 &= r_{13}^2 (r_{33}^2)^* r_{31}^2 + r_{13}^2 = 0^*1(\epsilon + (0+1)0^*1)^*(\epsilon + (0+1)0^*1) + 0^*1 \\ &= 0^*1((0+1)0^*1)^* \end{aligned}$$

Thus we have

$$L_m(A) = r_{12}^3 + r_{13}^3 = 0^*1((0+1)0^*1)^*(\epsilon + (0+1)(00)^*) + 0(00)^*.$$

The generated language $L(A)$ is given by

$$L(A) = r_{11}^3 + r_{12}^3 + r_{13}^3$$

where

$$r_{11}^3 = r_{13}^2 (r_{33}^2)^* r_{31}^2 + r_{11}^2 = (0^*1((0+1)0^*1)^*(0+1)0 + \epsilon)(00)^*$$

SOLUTION 8.5

Two states p and q are said to be *equivalent*, $p \equiv q$ if and only if for each input string x , $\delta(p, x)$ is a marked state if and only if $\delta(q, x)$ is a marked place. Two states p and q are said to be *distinguishable* if there exists an x such that $\delta(p, x)$ is in Q_m and $\delta(q, x)$ is not.

We build a table as follows

b	X						
c	X	X					
d	X	X	X				
e		X	X	X			
f	X	X	X		X		
g	X	X	X	X	X	X	
h	X		X	X	X	X	
	a	b	c	d	e	f	g

where an 'X' is placed in the table each time a pair of states cannot be equivalent. Initially an 'X' is placed in each entry corresponding to one marked state and one non-marked state. It is in fact impossible that a marked state and a non-marked state are equivalent. In the example we place an 'X' in the entries (a, c) , (b, c) , (c, d) , (c, e) , (c, f) , (c, g) and (c, h) . Next for each pair of states p and q that are not already known to be distinguishable, we consider the pairs of states $r = \delta(p, a)$ and $s = \delta(q, a)$ for an input symbol a . If states r and s have been shown to be distinguishable by some string x then p and q are distinguishable by the string ax . Thus if the entry (r, s) in the table has an 'X', an 'X' is also placed at the entry (p, q) . If the entry (r, s) does not yet have an 'X', then the pair (p, q) is placed on a list associated with the (r, s) -entry. At some future time, if (r, s) entry receives an 'X', then each pair on the list associated with the (r, s) -entry also receives an 'X'.

In the example, we place an 'X' in the entry (a, b) , since the entry $(\delta(b, 1), \delta(a, 1)) = (c, f)$ is already with 'X'. Similarly, the (a, d) -entry receives an 'X' since the entry $(\delta(a, 0), \delta(b, 0)) = (b, c)$ has an 'X'. Consideration of the (a, e) -entry on input 0 results in the pair (a, e) being placed on the list associated with (b, h) . Observe that on input 1, both a and e go to the same state f and hence no string starting with 1 can distinguish a from e . Because of the 0-input, the pair (a, g) is placed on the list associated with (b, g) . When the (b, g) -entry is considered, it receives an 'X' on account of a 1-input and (a, g) receives a 'X'. The string 01 distinguishes a from g .

At the end of the algorithm $a \equiv e$, $b \equiv h$ and $d \equiv f$. The minimum automaton is shown in Figure 8.5.1.

SOLUTION 8.6

We can construct a deterministic automaton

$$A' = (Q', \{0, 1\}, \delta', [q_0], Q_m)$$

accepting $L_m(A)$ as follows. The set Q' consists of all subsets of Q . These we denote with $[q_0]$, $[q_1]$, $[q_0, q_1]$, and \dots . Since $\delta(q_0, 0) = \{q_0, q_1\}$ then we have

$$\delta'([q_0], 0) = [q_0, q_1]$$

and similarly

$$\delta'([q_0], 1) = [q_1] \quad \delta'([q_1], 0) = [q_0, q_1] \quad \delta'([q_1], 1) = [q_0, q_1].$$

Naturally, $\delta'(\emptyset, 0) = \delta'(\emptyset, 1) = \emptyset$. Finally

$$\delta'([q_0, q_1], 0) = [q_0, q_1]$$

since

$$\delta(\{q_0, q_1\}, 0) = \delta(q_0, 0) \cup \delta(q_1, 0) = \{q_0, q_1\} \cup \{q_0, q_1\},$$

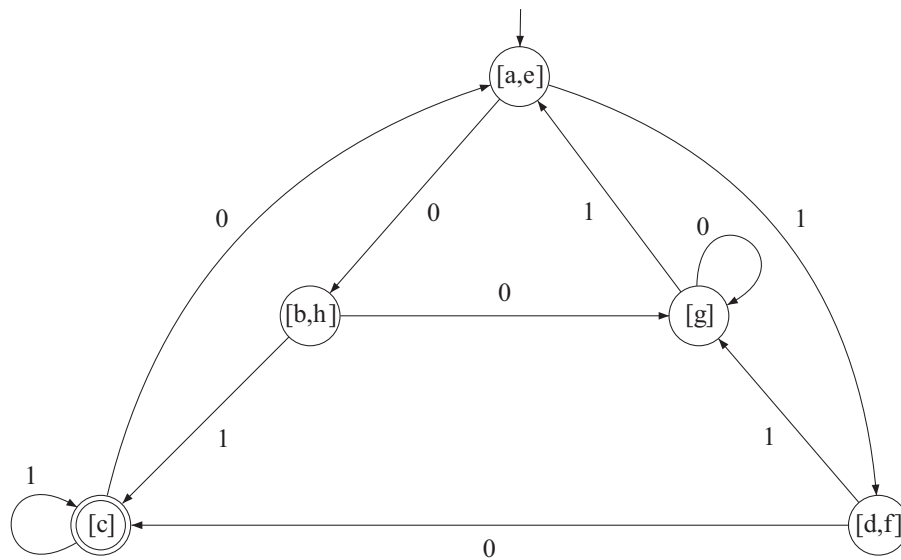


Figure 8.5.1: Minimum automaton of Problem 8.5.

and similarly we get

$$\delta([q_0, q_1]) = [q_0, q_1].$$

The final states are $Q'_m = \{[q_1], [q_0, q_1]\}$.

Solutions to models of computation II: Transition systems

SOLUTION 9.1

An automaton $H = (Q, E, \delta, q_0, Q_m)$ can be easily described by a transition system $T_H = \{S, \Sigma, \rightarrow, S_S, S_F\}$ where

- The state is $S = Q$.
- The generator set is $\Sigma = E$.
- The transition relation is $\rightarrow S \times \Sigma \times S = \delta$.
- The set of starting states is $S_S = q_0$.
- The set of final states is $S_F = Q_m$.

Therefore, a transition system is just another formal way to describe a Discrete Event System

SOLUTION 9.2

The states *MainMenu*, *Contacts* and *LockMenu* of a typical keypad can be modelled by figure 9.2.1.

The transition system is then $T_C = \{S, \Sigma, \rightarrow, S_S, S_F\}$, where

- $S = \{\text{MainMenu}, \text{Contacts}, \text{LockMenu}, \text{UnLock}\}$.
- The generator set is $\Sigma = \{\text{menu}, \text{contacts}, \text{exit}, \#\#\}$.

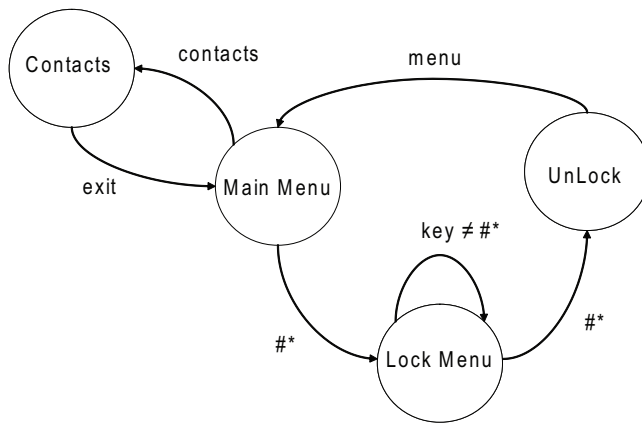


Figure 9.2.1: Possible states of a Keypad of a cellphone.

- The transition relation is $\rightarrow S \times \Sigma \times S = \delta$, where

$$\begin{aligned} \delta(\text{MainMenu}, \#*) &= \text{LockMenu}, \\ \delta(\text{MainMenu}, \text{contacts}) &= \text{Contacts}, \\ &\vdots \end{aligned}$$

- A starting state is $S_S = \text{LockMenu}$.
- A final state is $S_F = \text{LockMenu}$.

SOLUTION 9.3

The queuing system of figure 9.3 has a generator set $\Sigma = \{a, d\}$.

A natural state variable is the number of customers in queue, thus the state-space is the set of non-negative integers $S = \{0, 1, 2, \dots\}$.

The transition relation is

$$\begin{aligned} f(x, a) &= x + 1 & \forall x \geq 0 \\ f(x, d) &= x - 1 & \forall x > 0. \end{aligned}$$

The starting state q_0 is chosen to be the initial number of customers in the system.

In figure 9.3.1, the transition system representing the basic queue system is reported. It is evident that the cardinality of the state is infinite, but it is also countable.

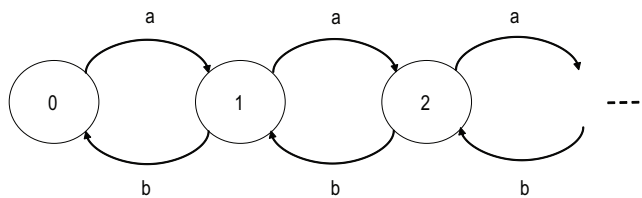


Figure 9.3.1: A basic queue system.

SOLUTION 9.4

The proof is as follows:

1. In each iteration the number of elements in Reach_i increases by at least 1. Since it can have, at most, as many elements as S , there can only be as many iterations as the number of elements in S (minus the number of elements in S_S).
2. Reach_i is the set of states that can be reached in i steps, thus any state that can be reached in a finite number of steps must be in one of the Reach_i .

SOLUTION 9.5

Let $S_S = \{3\}$. By applying the reachability algorithm we have

$$S_S = 3$$

$$\text{Reach}_0 = \{3\}$$

$$\text{Reach}_1 = \{1, 3, 5, 6\}$$

$$\text{Reach}_2 = \{1, 2, 3, 5, 6\}$$

$$\text{Reach}_3 = S$$

$$\text{Reach}_4 = S$$

$$\text{Reach}_T(\{3\}) = S$$

Let $S_S = \{2\}$. By applying the reachability algorithm we have

$$S_S = 2$$

$$\text{Reach}_0 = \{2\}$$

$$\text{Reach}_1 = \{2, 4, 5\}$$

$$\text{Reach}_2 = \{2, 4, 5\}$$

$$\text{Reach}_T(\{2\}) = \{2, 4, 5\}$$

Part III

Hybrid control

Solution to modeling of hybrid systems

SOLUTION 10.1

- (a) Let the water level be $x \geq 0$ m and use four discrete states. Let the thresholds be x_{on} and x_{off} . The suggested model is illustrated in Figure 10.1.1.

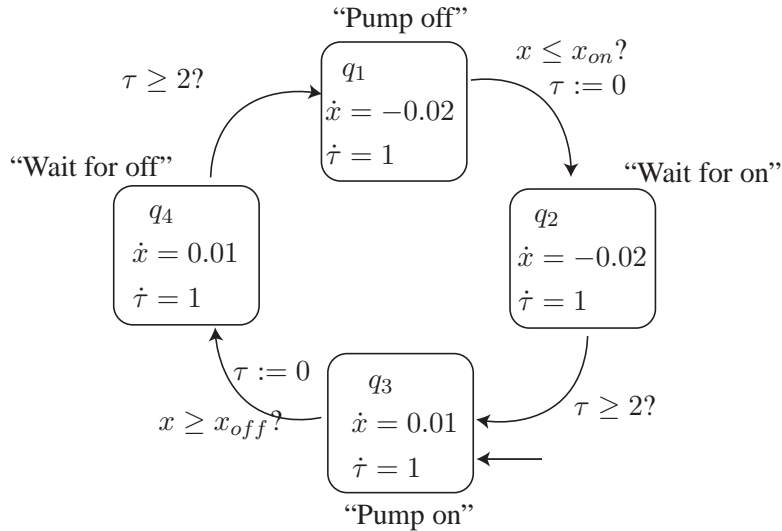


Figure 10.1.1: Hybrid model of the tank and relay controller in Problem 10.1.

(The variable τ is included as a "timer", used to keep track of how long the system stays in the waiting states.)

- (b) The water level decreases $2 \cdot 0.02$ m=0.04 m while waiting for the pump to turn on. Thus we need to choose $x_{on}=0.09$ to prevent underfilling. Similarly, choose

$$x_{off} = 0.12 - 2 \cdot 0.01 = 0.10.$$

SOLUTION 10.2

We see that v can assume 2^k values, called $v_i, i \in \{0, \dots, 2^k - 1\}$. So we introduce $N = 2^k$ discrete states q_i . In state q_i , the system has v_i as constant control signal, illustrated in Figure 10.2.1.



Figure 10.2.1: Dynamics of state q_i in Problem 10.2.

When do we switch states? We define the edges as

$$E = \{(q_i, q_{i+1}), (q_{i+1}, q_i) | i = 0, \dots, N - 2\},$$

i.e., the system can only switch between adjacent states. The switchings are controlled by guards and domains. The guards define when switching is *allowed*, and when the system is outside the domain, it *must* switch:

$$\begin{aligned} G(q_i, q_{i+1}) &= \{u \geq v_i + D/2\} \\ G(q_{i+1}, q_i) &= \{u \leq v_{i+1} - D/2\} \\ D(q_i) &= \{v_i - D/2 \leq u \leq v_i + D/2\} \end{aligned}$$

So the hybrid system can be illustrated as in Figure 10.2.2.

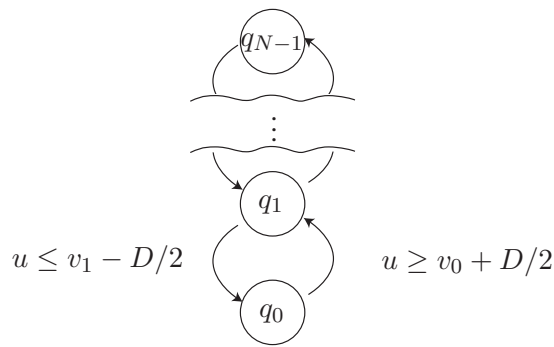


Figure 10.2.2: Control system with quantizer, modeled as a hybrid system. (Problem 10.2)

SOLUTION 10.3

A hybrid system modeling the nuclear reactor is shown in Figure 10.3.1.

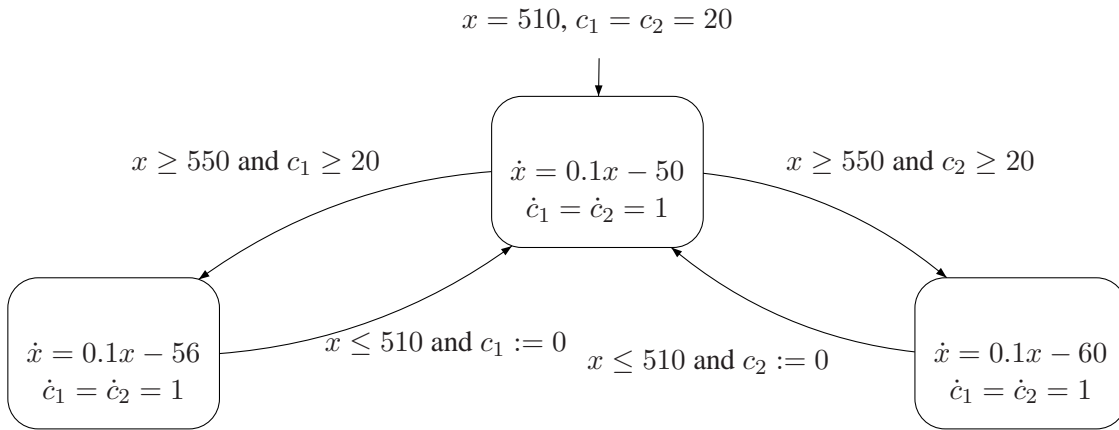


Figure 10.3.1: The hybrid system for the nuclear reactor

SOLUTION 10.4

We model the system on linear state space form with a controller $u = f(y)$ and neglect the time for computing the control signal. (The computing time could otherwise be modeled as an extra waiting state between sampling $y(t)$ and applying the new $u(t)$.) The suggested hybrid model is illustrated in Figure 10.4.1.

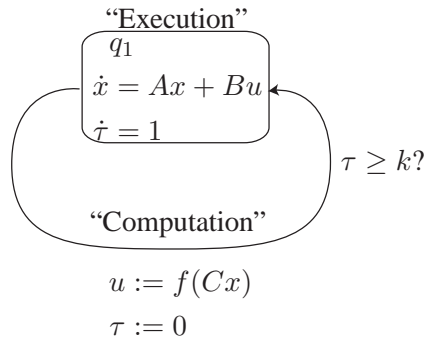


Figure 10.4.1: The sampled control system in Problem 10.4.

The control law, as well as any quantization effects in the D/A or A/D converters are incorporated into $f(y)$.

SOLUTION 10.5

- (a) We need a timer variable t to control the switching, so the full state is $(x, t)^T$. Now the hybrid automaton can be described as $H = (Q, X, \text{Init}, f, D, E, G, R)$, with

$$\begin{aligned} Q &= \{q_1, q_2\} \\ X &= \mathbb{R}^2 \times \mathbb{R}^+ \\ \text{Init} &= q_1 \times (0, 0) \\ f(q_i, x, t) &= \begin{bmatrix} A_i x & 1 \end{bmatrix}^T \\ D(q_i) &= \{(x, t)^T \mid t < 1\} \\ E &= \{(q_1, q_2), (q_2, q_1)\} \\ G(q_1, q_2) &= G(q_2, q_1) = \{(x, t)^T \mid t \geq 1\} \\ R(q_1, q_2, x, t) &= R(q_2, q_1, x, t) = \begin{bmatrix} x & 0 \end{bmatrix}^T. \end{aligned}$$

- (b) The trajectory evolves as follows:

$$x(t) = \begin{cases} e^{A_1 t} x_0, & t \in [0, 1) \\ e^{A_2(t-1)} x(1), & t \in [1, 2) \\ e^{A_1(t-2)} x(2), & t \in [2, 3) \end{cases}$$

In order to express $x(t)$ as a function of x_0 we need to derive $x(1)$ and $x(2)$ as functions of x_0 . We have

$$x(1) = e^{A_1(1-0)} x_0 = e^{A_1} x_0$$

and

$$x(2) = e^{A_2(2-1)} x(1) = e^{A_2} x(1).$$

Combining the two previous equations we have

$$x(t) = \begin{cases} e^{A_1 t} x_0, & t \in [0, 1) \\ e^{A_2(t-1)} e^{A_1} x_0, & t \in [1, 2) \\ e^{A_1(t-2)} e^{A_2} e^{A_1} x_0, & t \in [2, 3). \end{cases}$$

SOLUTION 10.6

(a) The hybrid automaton is defined as $H = (Q, X, \text{Init}, f, D, E, G, R)$ with

$$\begin{aligned}
 Q &= \{q_1, q_2, q_3\} \\
 X &= \mathbb{R} \\
 \text{Init} &= q_1 \times 0 \\
 f(q_1, x) &= 2 \\
 f(q_2, x) &= -1 \\
 f(q_3, x) &= x + 2 \\
 D(q_1) &= \{x \in \mathbb{R} \mid x < 5\} \\
 D(q_2) &= \{x \in \mathbb{R} \mid x > 3\} \\
 D(q_3) &\text{ to be defined} \\
 E &= \{(q_1, q_2), (q_2, q_3)\} \\
 G(q_1, q_2) &= \{x \in \mathbb{R} \mid x \geq 5\} \\
 G(q_2, q_3) &= \{x \in \mathbb{R} \mid x \leq 3\} \\
 R(q_1, q_2, x) &= x \\
 R(q_2, q_3, x) &= -2
 \end{aligned}$$

(b) In order for the hybrid automaton to be live we need to ensure that the domain $D(q_3)$ contains the trajectory of the system $\dot{x} = x + 2$ with initial condition $x_0 = -2$. But

$$x_0 = -2 \Rightarrow \dot{x} = -2 + 2 = 0,$$

so the state will stay constant. Thus all domains that fulfill $\{x = 2\} \in D(q_3)$ guarantee liveness of the system.

(c) The state-trajectory of the hybrid system is shown in Figure 10.6.1, where $\tau_0 = 0$, $\tau_1 = 2.5$ and $\tau_2 = 4.5$.

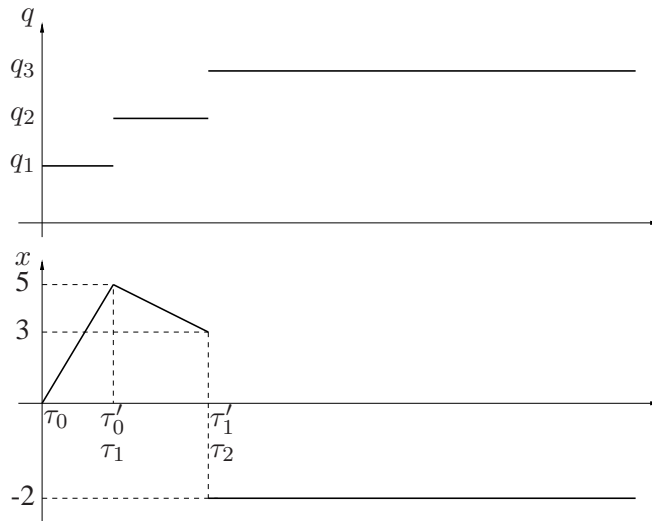


Figure 10.6.1: Trajectory of the hybrid system in Problem 10.6

Solutions to stability of hybrid systems

SOLUTION 11.1

(a) A possible solution is $H = (Q, X, \text{Init}, f, D, E, G, R)$ where

$$\begin{aligned} Q &= \{q\} \\ X &= \mathbb{R}^3 \text{ and } x = (v_1, v_2, v_3)^T \\ \text{Init} &= q \times (1, 0, 0) \\ f(q, x) &= (0, 0, 0)^T \\ D(q) &= \{x \in \mathbb{R}^3 \mid v_1 \leq v_2 \leq v_3\} \\ E &= \{(q, q)\} \\ G(q, q) &= \{x \in \mathbb{R}^3 \mid v_1 > v_2\} \cup \{x \in \mathbb{R}^3 \mid v_2 > v_3\} \\ R(q, q, x) &= \begin{cases} \left(\frac{v_1 + v_2}{2}, \frac{v_1 + v_2}{2}, v_3 \right) & \text{if } v_1 > v_2 \\ \left(v_1, \frac{v_2 + v_3}{2}, \frac{v_2 + v_3}{2} \right) & \text{if } v_2 > v_3 \end{cases} \end{aligned}$$

(b) The collisions result in the following evolution for the continuous state:

$$(1, 0, 0) \rightarrow \left(\frac{1}{2}, \frac{1}{2}, 0\right) \rightarrow \left(\frac{1}{2}, \frac{1}{4}, \frac{1}{4}\right) \rightarrow \left(\frac{3}{8}, \frac{3}{8}, \frac{1}{4}\right) \rightarrow \dots$$

which constitute an infinite sequence. We have that the solution is Zeno since it has an infinite number of discrete transitions over a finite time interval, which in this case is a zero time interval.

(b) Since $v_1(\tau_i) - v_3(\tau_i) = 2^{-i} \rightarrow 0$ as $i \rightarrow \infty$ and $v_1(\tau_\infty) = v_2(\tau_\infty) = v_3(\tau_\infty)$, it follows that the accumulation point is $(1/3, 1/3, 1/3)$. The physical interpretation is that after an infinite number of hits, the balls will have the same velocity (equal to one third of the initial velocity of ball 1).

SOLUTION 11.2

We can consider the following Lyapunov function:

$$V(x) = \frac{1}{2}x^T x$$

It is an acceptable candidate, since

$$V(x) \geq 0,$$

with equality only at $x = 0$. We then have

$$\dot{V} = x^T \dot{x} = x^T Ax = (x_1 \quad x_2 \quad x_3) \begin{pmatrix} -1 & 0 & 2 \\ 0 & -1 & 3 \\ -2 & -3 & -2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

which gives

$$\dot{V}(x) = -x_1^2 - x_2^2 - 2x_3^2 < 0 \quad \forall x \neq 0$$

Thus we can conclude that the system is asymptotically stable.

SOLUTION 11.3

Let P be a positive definite symmetric matrix which satisfies

$$A^T P + P A = -Q.$$

Then the function $V(x) = x^T P x$ is a positive definite function. Let us compute $\dot{V}(x)$

$$\dot{V}(x) = \frac{d}{dt}(x^T P x) = \dot{x}^T P x + x^T P \dot{x} = x^T A^T P x + x^T P A x = x^T (A^T P + P A) x = -x^T Q x.$$

Thus $\dot{V}(x)$ is negative definite. For Lyapunov's stability criteria we can conclude that the system is asymptotically stable.

SOLUTION 11.4

Let us consider the following Lyapunov function

$$V(x) = \frac{x^T x}{2} = \frac{x_1^2 + x_2^2}{2}$$

Let us compute the time derivative of $V(x)$. We have

$$\begin{aligned} \dot{V}(x) &= x_1 \dot{x}_1 + x_2 \dot{x}_2 = -x_1^2 - x_2^2 + x_1 g(x_2) + x_2 h(x_1) \\ &\leq -x_1^2 - x_2^2 + \frac{1}{2} x_1 |x_2| + \frac{1}{2} x_2 |x_1| \\ &\leq -x_1^2 - x_2^2 + |x_1 x_2| \\ &\leq -\frac{1}{2} (x_1^2 + x_2^2) \end{aligned}$$

where we used $(|x_1| - |x_2|)^2 \geq 0$, which gives $|x_1 x_2| \leq \frac{1}{2} (x_1^2 + x_2^2)$. Thus

$$\dot{V} < 0 \quad \forall x \neq 0$$

thus the system is asymptotically stable. Notice that we do not need to know the expressions for $g(\cdot)$ and $h(\cdot)$ to show asymptotic stability.

SOLUTION 11.5

(a) Depending on the sign of x_1 or x_2 we have four systems. If $x_1 > 0$ and $x_2 > 0$ then we have

$$\begin{aligned} \dot{x}_1 &= -1 + 2 = 1 \\ \dot{x}_2 &= -2 - 1 = -3 \end{aligned}$$

If $x_1 < 0$ and $x_2 > 0$ then we have

$$\begin{aligned} \dot{x}_1 &= 1 + 2 = 3 \\ \dot{x}_2 &= 2 - 1 = 1 \end{aligned}$$

If $x_1 < 0$ and $x_2 < 0$ then we have

$$\begin{aligned} \dot{x}_1 &= 1 - 2 = -1 \\ \dot{x}_2 &= 2 + 1 = 3 \end{aligned}$$

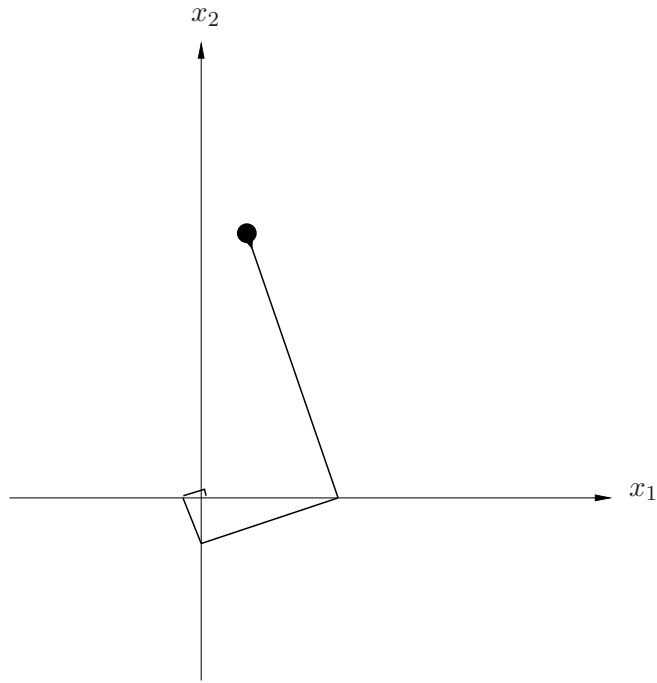


Figure 11.5.1: A trajectory of the system in Problem 11.5. The dot represents the initial state.

If $x_1 > 0$ and $x_2 < 0$ then we have

$$\begin{aligned}\dot{x}_1 &= -1 - 2 = -3 \\ \dot{x}_2 &= -2 + 1 = -1\end{aligned}$$

A trajectory is represented in Figure 11.5.1. We can then represent the discontinuous system as a hybrid system with four states. Formally we have $H = (Q, X, \text{Init}, f, D, E, G, R)$ with

$$\begin{aligned}Q &= \{q_1, q_2, q_3, q_4\} \\ X &= \mathbb{R}^2 \\ \text{Init} &= q_0 \times (x_{10}, x_{20}),\end{aligned}$$

where q_0 depends on the state x_0 .

$$\begin{aligned}f(q_1, x) &= (1, -3)^T \\ f(q_2, x) &= (3, 1)^T \\ f(q_3, x) &= (-1, 3)^T \\ f(q_4, x) &= (-3, -1)^T \\ D(q_1) &= \{x \in \mathbb{R}^2 \mid x_1 \geq 0, x_2 \geq 0\} \\ D(q_2) &= \{x \in \mathbb{R}^2 \mid x_1 < 0, x_2 \geq 0\} \\ D(q_3) &= \{x \in \mathbb{R}^2 \mid x_1 < 0, x_2 < 0\} \\ D(q_4) &= \{x \in \mathbb{R}^2 \mid x_1 \geq 0, x_2 < 0\} \\ R(q_i, q_j, x) &= x\end{aligned}$$

We still need to define the set of edges E and the guards G . Since a trajectory of the given discontinuous system is always such that from the 1st quadrant it moves to the 4th and then 3rd and 2nd (see

Figure 11.5.1), the set of edges is then

$$E = \{(q_1, q_4), (q_4, q_3), (q_3, q_2), (q_2, q_1)\}.$$

The guards are

$$G(q_1, q_4) = \{x \in \mathbb{R}^2 | x_2 < 0\}$$

$$G(q_4, q_3) = \{x \in \mathbb{R}^2 | x_1 < 0\}$$

$$G(q_3, q_2) = \{x \in \mathbb{R}^2 | x_2 \geq 0\}$$

$$G(q_2, q_1) = \{x \in \mathbb{R}^2 | x_1 \geq 0\}.$$

- (b) We can notice that for any initial condition $(x(0) \neq 0)$ a solution of the hybrid system will look like a 'spiral' which is moving towards the origin. As we approach the origin the number of switches increases, until we have an infinite number of switches in finite time. Thus we have Zeno behavior.

SOLUTION 11.6

Consider the Lyapunov function candidate

$$V(x) = \frac{1}{2}x^2.$$

For both $q = 1$ and $q = 2$ it holds that

$$\dot{V}(x) = x\dot{x} = a_q x^2 \leq 0,$$

with equality only for $x = 0$. Then $V(x)$ is a common Lyapunov function for the switched system, and it is asymptotically stable.

SOLUTION 11.7

The matrices A_1 and A_2 commute, *i.e.*

$$A_1 A_2 = \begin{bmatrix} 3 & 0 \\ 0 & 10 \end{bmatrix} = A_2 A_1.$$

Further, they are both diagonal, which means that the eigenvalues are the diagonal elements. All eigenvalues are negative, so each matrix is asymptotically stable.

If the system matrices commute and are asymptotically stable, then the switched system is also asymptotically stable.

SOLUTION 11.8

1st method The second component of the state x satisfies

$$\dot{x}_2 = -c_q x_2$$

where $q = \{1, 2\}$. Therefore $x_2(t)$ decays to 0 asymptotically at the rate corresponding to $\min\{c_1, c_2\}$. The first component of x satisfies the equation

$$\dot{x}_1 = -a_q x_1 + b_q x_2.$$

This can be viewed as the asymptotically stable system $\dot{x}_1 = -a_q x_1$ excited by a asymptotically decaying input $b_q x_2$. Thus x_1 also converges to zero asymptotically.

2nd method A second way to show that the switched system is asymptotically stable consists in constructing a common Lyapunov function for the family of linear system A_q . In this case it is possible to find a quadratic common Lyapunov function

$$V(x) = x^T P x$$

with $P = \text{diag}(d_1, d_2)$ with $d_1, d_2 > 0$, since P must be positive definite. We then have

$$-(A_q^T P + P A_q) = \begin{pmatrix} 2d_1 a_q & -d_1 b_q \\ -d_1 b_q & 2d_2 c_q \end{pmatrix} \quad q = 1, 2$$

To ensure that this matrix is positive definite we can fix $d_1 > 0$ and then choose $d_2 > 0$ large enough so that

$$4d_2 d_1 a_q c_q + d_1^2 b_q^2 > 0, \quad q = 1, 2.$$

Thus this concludes the proof.

SOLUTION 11.9

(a) To model the system as a switching system, we introduce a timer variable τ :

$$\begin{aligned} \dot{x} &= A_q x \\ \dot{\tau} &= 1, \end{aligned}$$

where $q \in \{1, 2\}$ and

$$\begin{aligned} \Omega_1 &= \{x, \tau | k\epsilon \leq \tau < (k + \frac{1}{2})\epsilon, k = 0, 1, \dots\} \\ \Omega_2 &= \{x, \tau | (k + \frac{1}{2})\epsilon \leq \tau < (k + 1)\epsilon, k = 0, 1, \dots\}. \end{aligned}$$

(b) A corresponding hybrid automaton is $H = (Q, X, \text{Init}, f, D, E, G, R)$, with

$$\begin{aligned} Q &= \{q_1, q_2\} \\ X &= \mathbb{R}^n \times \mathbb{R}^+ \text{ (state vector: } (x, \tau)^T) \\ \text{Init} &= q_1 \times (x_0, t_0)^T \\ f(q_1, x, \tau) &= (A_1 x, 1)^T \\ f(q_2, x, \tau) &= (A_2 x, 1)^T \\ D(q) &= \{x, \tau | \tau < \epsilon/2\} \forall q \in Q \\ E &= \{(q_1, q_2), (q_2, q_1)\} \\ G(e) &= \{x, \tau | \tau \geq \epsilon/2\} \forall e \in E \\ R(e, x, \tau) &= (x, 0)^T \forall e \in E. \end{aligned}$$

Note that here we let τ be reset at every switching instant, to simplify the system.

(c) If the system starts in $q = q_1$, with $x = x_0$ at $t = t_0$, we get

$$x(t_0 + \epsilon/2) = e^{A_1 \epsilon/2} x_0$$

and

$$x(t_0 + \epsilon) = e^{A_2 \epsilon/2} e^{A_1 \epsilon/2} x_0.$$

(d) We use the definition of the matrix exponential:

$$e^{A\epsilon} = I + A\epsilon + A^2\epsilon^2 + \mathcal{O}(\epsilon^3)$$

This allows us to write

$$\begin{aligned} e^{A_2\epsilon/2}e^{A_1\epsilon/2} &= (I + A_2\epsilon/2 + \dots)(I + A_1\epsilon/2 + \dots) \\ &= I + (A_1 + A_2)/2\epsilon + A_2A_1\epsilon^2/4 + \dots \approx \{ \text{as } \epsilon \rightarrow 0 \} \approx e^{\frac{A_1+A_2}{2}\epsilon} \end{aligned}$$

So for fast switching (small ϵ), the system behaves like the average of the two subsystems.

SOLUTION 11.10

We choose $Q = I$ and test if

$$P = \begin{bmatrix} p_1 & 0 \\ 0 & p_2 \end{bmatrix}$$

solves the Lyapunov equation

$$PA_1 + A_1^T P = -I \Leftrightarrow \begin{bmatrix} -2p_1 & 0 \\ 0 & -4p_2 \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}.$$

Apparently it does, if we choose $p_1 = 1/2$ and $p_2 = 1/4$. So $V(x) = x^T P x$ is a Lyapunov function for system 1. Does it also work for system 2?

$$\begin{aligned} \dot{V}(x) &= \dot{x}^T P x + x^T P \dot{x} = x^T A_2 P x + x^T P A_2 x \\ &= x^T \begin{bmatrix} -3 & 0 \\ 0 & -5/2 \end{bmatrix} x \leq 0 \quad \forall x, \end{aligned}$$

with equality only for $x = 0$. So $V(x)$ is a common Lyapunov function for systems 1 and 2 and the switched system is asymptotically stable.

SOLUTION 11.11

(a) Without loss of generality we can consider a positive definite matrix P in the form

$$P = \begin{pmatrix} 1 & s \\ s & r \end{pmatrix}.$$

Assume the matrix P satisfies

$$A_q^T P + P A_q < 0 \quad \forall q$$

then for $q = 1$ we have

$$-A_1^T P - P A_1 = \begin{pmatrix} 2 - 2q & 2q + 1 - r \\ 2q + 1 - r & 2q + 2r \end{pmatrix}$$

and this matrix is positive definite only if

$$\begin{aligned} q &< 1 \\ 8q^2 + 1 + r^2 - 6r &< 0 \Rightarrow q^2 + \frac{(r-3)^2}{8} < 1 \end{aligned} \tag{0.0}$$

Similarly for the other matrix we have

$$-A_2^T P - P A_2 = \begin{pmatrix} 2 - \frac{q}{5} & 2q + 10 - \frac{r}{10} \\ 2q + 10 - \frac{r}{10} & 20q + 2r \end{pmatrix}$$

which is a positive definite matrix if

$$q < 10$$

$$600r - 800q^2 - 1000 - r^2 \Rightarrow q^2 + \frac{(r - 300)^2}{800} < 100. \quad (0.0)$$

The inequalities 0.0 and 0.0 represent the interiors of two ellipsoids. As can be seen in Figure 11.11.1 the two ellipsoids do not intersect.

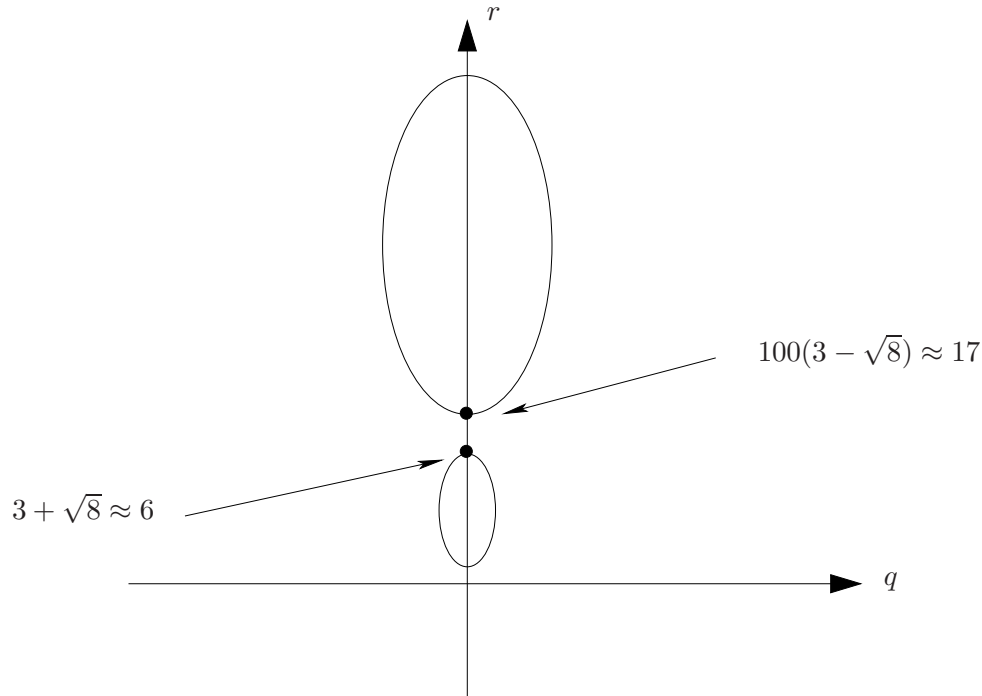


Figure 11.11.1: Ellipsoids representing the inequalities (0.0) and (0.0).

- (b) We would like to show that the system is however asymptotically stable. In order to do that, since the switching is arbitrary, we consider a 'worst-case-switching', which we define as follows. The vectors A_1x and A_2x are parallel on two lines which pass through the origin. If such lines define a switching surface, we then choose to follow the trajectory of the vector field which is pointing outward, so that we have a 'worst-case-switching', in the sense that it should move away from the origin. The situation is shown in Figure 11.11.2. The switching surface can be found solving the equation which describes collinearity

$$(A_1x) \times (A_2x) = 0$$

which gives the equation

$$(-x_1 - x_2)(0.1x_1 - x_2) + (x_1 - x_2)(x_1 + 10x_2) = 0.$$

The equations of the two lines are

$$\begin{aligned} x_2 &= 1.18x_1 \\ x_2 &= -0.08x_1. \end{aligned}$$

In order to know which of the two vectors is pointing outward we can determine when $(A_1x) \times (A_2x) > 0$ which gives that between the line $-0.08x_1$ and $1.18x_1$ the vectors A_2x are pointing outward. We now

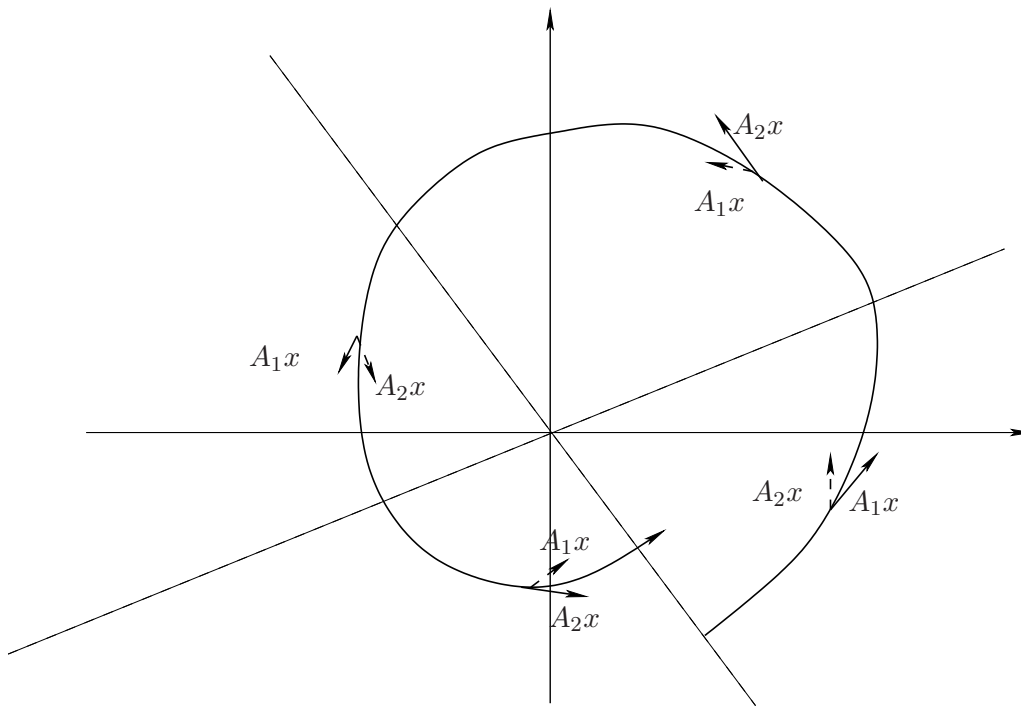


Figure 11.11.2: Worst case switching in Problem 11.11

need to show that the trajectory is converging toward the origin. Let us consider, for example, the following initial condition

$$x_0 = (100, -8)$$

which is a point on the switching surface $x_2 = -0.08x_1$. The trajectory follows the vector field A_1x and intersects the next switching surface at the point

$$\begin{pmatrix} \bar{x}_1 \\ 1.18\bar{x}_1 \end{pmatrix} = e^{A_1\bar{t}}x_0$$

which in our case is the point with coordinates $(25, 29.5)$. Then the trajectory is given by the vector field A_2x and the intersection point with the switching surface is $(-86.3, 6.8)$. The next point is $(-21.7, -25)$. The next point is the intersection of the trajectory with the switching surface that contained the initial condition x_0 . We have the intersection at $(25, -2)$. Thus after one rotation we have that the distance of the trajectory to the origin is decreased from 100.1 to 25.1. Thus the system converges to the origin considering the 'worst-case-switching'.

SOLUTION 11.12

- Use the same ideas as in Problem 11.11.
- Let us consider the following two positive definite symmetric matrices

$$P_1 = \begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix} \quad \text{and} \quad P_2 = \begin{pmatrix} 10 & 0 \\ 0 & 3 \end{pmatrix}$$

If we consider the following Lyapunov equations

$$A_q^T P_q + P_q A_q \quad q = 1, 2$$

we have

$$A_1^T P_1 + P_1 A_1 = - \begin{pmatrix} 10 & 7 \\ 7 & 12 \end{pmatrix} = Q_1$$

$$A_2^T P_2 + P_2 A_2 = \begin{pmatrix} -40 & 20 \\ 20 & 12 \end{pmatrix} = Q_2$$

Since the matrices Q_1 and Q_2 are symmetric matrices and the eigenvalues are $\sigma(Q_1) = \{-3.9, -18.1\}$ and $\sigma(Q_2) = \{-1.6, -50.1\}$, the two matrices are negative definite. Thus the two linear systems are asymptotically stable.

Let us consider the two Lyapunov functions

$$V_1 = x^T P_1 x$$

$$V_2 = x^T P_2 x.$$

In order to prove stability using the Lyapunov function approach we need to prove that the sequence

$$\{V_q(x(\tau_{i_q}))\}$$

is non-decreasing. In this case we can notice that we have the switching when x_1 changes sign. We can notice that

$$\lim_{x_1 \rightarrow 0^+} V_2(x) = 3x_2^2$$

and

$$\lim_{x_1 \rightarrow 0^-} V_1(x) = 3x_2^2$$

Thus the two Lyapunov function form a continuous non-increasing function, see Figure 11.12.1.

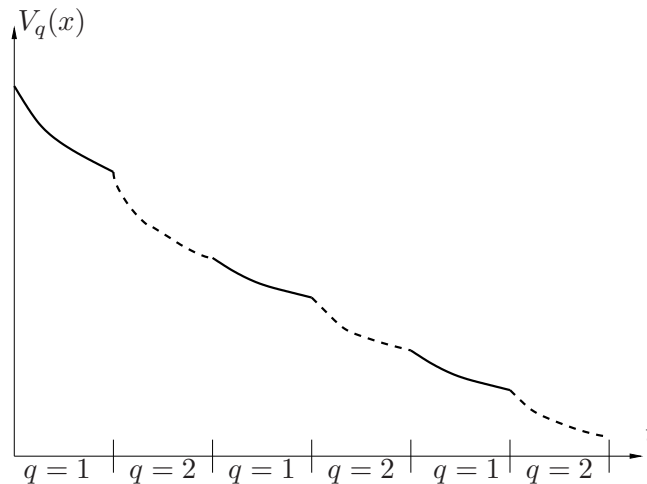


Figure 11.12.1: Multiple Lyapunov functions for Problem 11.12

Solutions to verification of hybrid systems

SOLUTION 12.1

The trajectories of the linear system are shown in Figure 12.1.1. It is easy to prove that no trajectory intersects the *Bad* set.

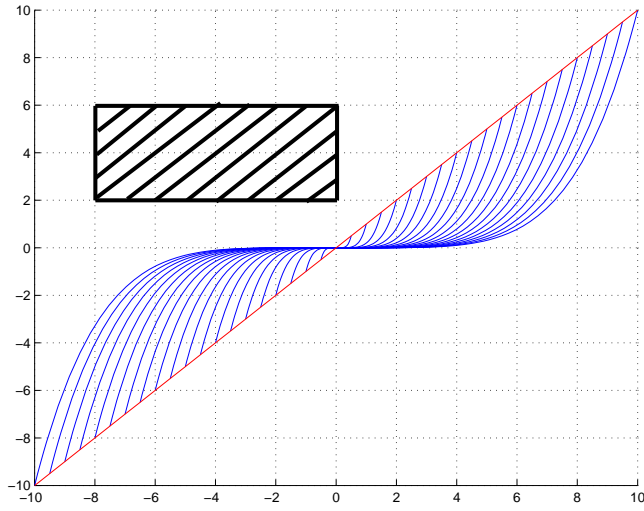


Figure 12.1.1: Trajectories of the linear system in Problem 12.1.

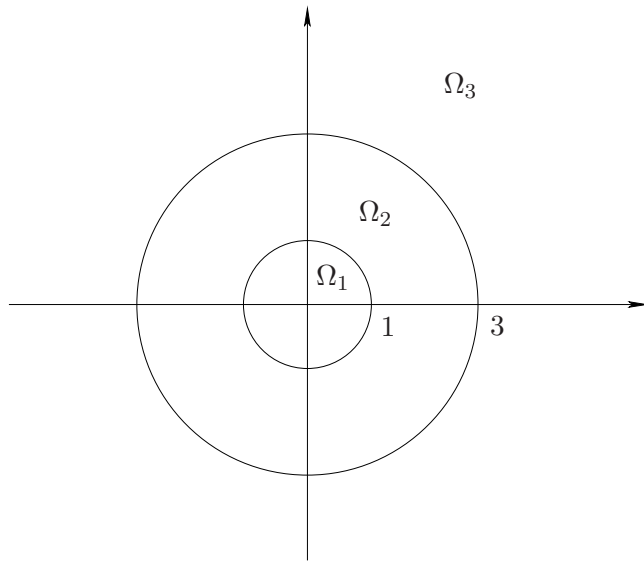


Figure 12.3.1: State-space partition for the system in Problem 12.3

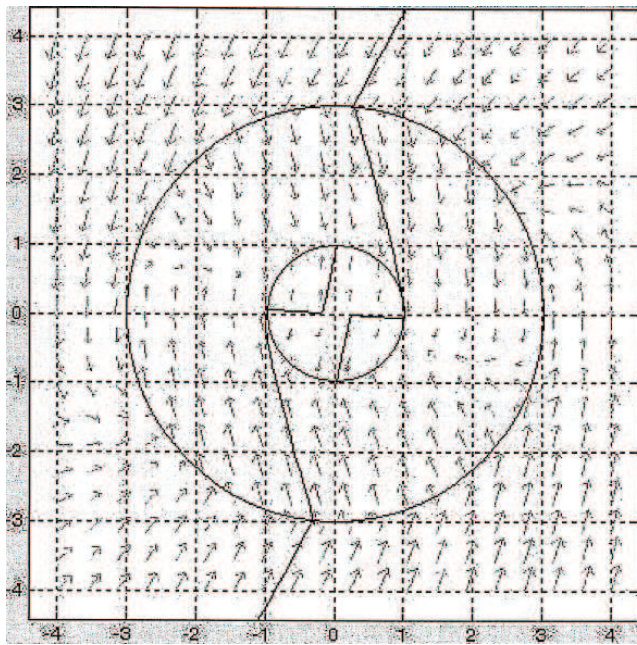


Figure 12.3.2: Trajectories of the linear system in Problem 12.3.

SOLUTION 12.3

The partition of the state-space is shown in Figure 12.3.1.

- (a) Let us consider the system in the region Ω_3 . It is easy to determine a linear state feedback $g_3(x) = K_3x$ which places the poles in $p_1 = p_2 = -1$, since the system is controllable. The feedback control gain is in this case

$$K_3 = \begin{pmatrix} -\frac{4}{3} & -3 \end{pmatrix}.$$

Notice that if we use g_3 as controller for the system 2 we have a closed loop stable system with poles

$$p_1 = -3 \quad p_2 = -\frac{1}{3}.$$

Notice moreover that the closed loop system for system 3 has the equilibrium point in $(-1/3, 0)$ since the closed loop system is

$$\dot{x} = \begin{pmatrix} 0 & -1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ -\frac{1}{3} \end{pmatrix}$$

This means that all trajectories starting in Ω_3 will go towards $(-1/3, 0)$, which is in Ω_1 . Hence they will enter Ω_2 , which also is stable with $(0, 0)$ as equilibrium. Thus the trajectory will enter Ω_1 . An example of trajectories is shown in Figure 12.3.2.

- (b) If $B_1 = (0, 1)^T$ then as you can notice the dynamics in the set $\Omega_3 = \{\|x\| > 3\}$ is the negative of the dynamics in Ω_1 (modulo a bias term that only makes the equilibrium different). Therefore if we use the linear controller g_3 computed in (a), this will not make the system 1 stable. Thus, we need to use a different controller for the two regions.
- (c) A simple choice to stabilize the system 1 is a feedback controller that place the poles in $p_1 = p_2 = -1$. Such controller controller for the system 1 is Let the controller

$$g_1(x_1, x_2) = \begin{pmatrix} -\frac{4}{3} & -7 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

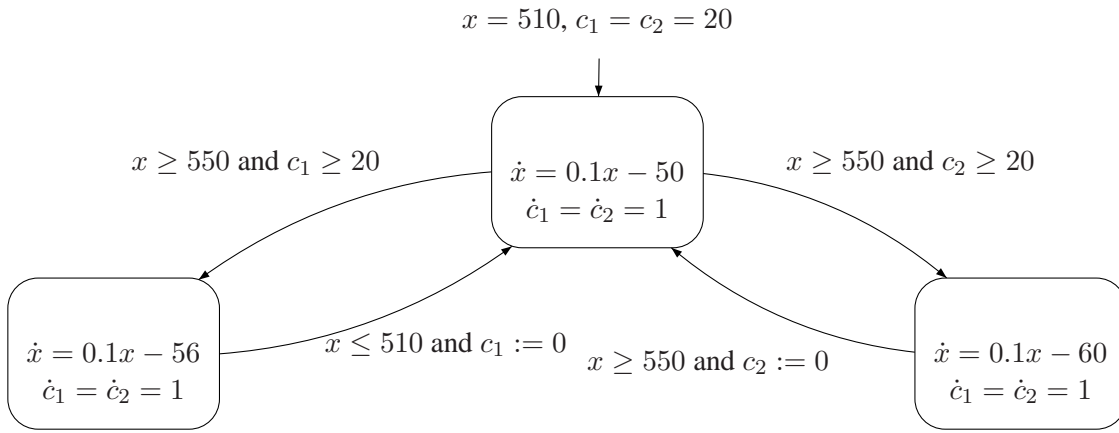


Figure 12.4.1: The hybrid system for the nuclear reactor

Notice that that controllers g_1 and g_3 place the poles of the systems 1 and 3 in $p_1 = p_2 = -1$. Both controllers stabilizes the system in Ω_2 , since the closed loop for the system 2, when g_3 is used, has poles in

$$p_1 = -3 \quad p_2 = -1/3$$

and using g_1 the poles are

$$p_{1,2} = \frac{-11 \pm 4\sqrt{7}}{3}.$$

thus a possible control strategy is

$$g(x) = \begin{cases} g_1(x) & \text{if } \|x\| \leq \alpha \\ g_3(x) & \text{otherwise} \end{cases}$$

with $1 \leq \alpha \leq 3$, which means that we can use either g_1 or g_3 in any subset of the region Ω_2 .

- (d) If the bad region corresponds to Ω_1 then we need to avoid the trajectory to go enter in Ω_1 at all. One possible solution is to design a control law g_1 such that the eigenvalues of the closed loop system for system 1 are placed in $p_{1,2} = \pm i$. In this way we have that the trajectories are confined in a circle with radius 1.

SOLUTION 12.4

A hybrid system modeling the nuclear reactor is shown in Figure 12.4.1.

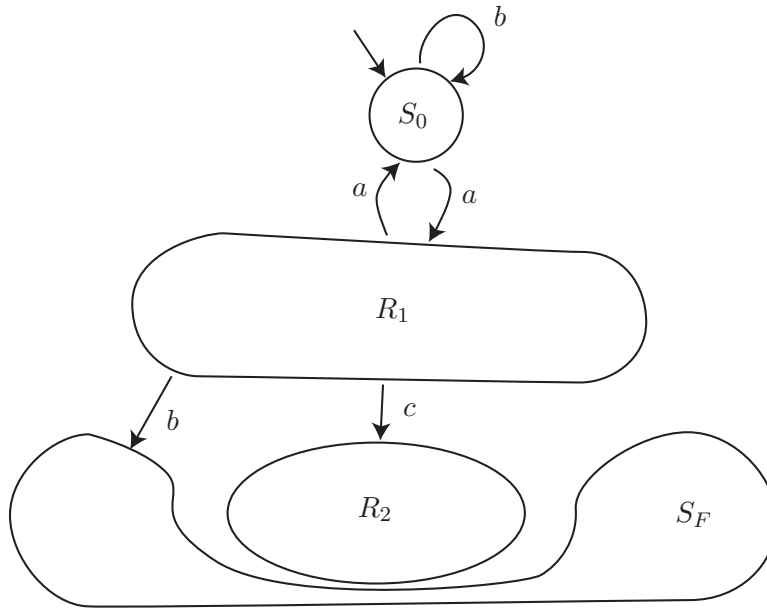


Figure 13.1.1: The resulting quotient transition system \hat{T} , which is bisimilar to T .

Solutions to simulation and bisimulation

SOLUTION 13.1

We use the bisimulation algorithm, and start by defining the quotient transition system as $\hat{T} = \{\hat{S}, \hat{\Sigma}, \hat{\rightarrow}, \hat{S}_0, \hat{S}_F\}$. Further, let $\text{Pre}_\sigma(P)$ denote all states from which we can get to P in one transition σ .

We initialize the algorithm by letting $S_1 = \{q_1, \dots, q_5\}$ and $\hat{S} = \{S_0, S_F, S_1\}$. Now there are $P_1 = S_1$, $P_2 = S_F$ and $\sigma = b$ such that

$$P_1 \cap \text{Pre}_b(P_2) = \{q_1, q_2\} \neq \emptyset.$$

Thus we must partition the set P_1 :

$$R_1 = P_1 \cap \text{Pre}_b(P_2) = \{q_1, q_2\}$$

$$R_2 = P_1 \setminus \text{Pre}_b(P_2) = \{q_4, q_5\}$$

$$\hat{S} := \hat{S} \setminus P_1 \cup \{R_1, R_2\} = \{S_0, S_F, R_1, R_2\}$$

Now there are no more P_1, P_2 and σ that fulfill the criterion in the algorithm, so it terminates. The resulting quotient transition system \hat{T} is given in Fig. III.

BIBLIOGRAPHY

- [1] Wittenmark, B. Åström, K.J. and Årzén K-H., "Computer Control: An Overview", IFAC Professional Brief, 2003.
- [2] Wittenmark, B. and Åström, K.J. , "Computer Controlled Systems", Prentice Hall, 3rd Edition.
- [3] Buttazzo, G.C., "Hard Real-Time Computing", Kluwer Academic Publisher, 4th edition, 2002. Systems"
- [4] Cassandras, C.G. and Lafortune, S., "Introduction to Discrete Event Systems", Kluwer Academic Publisher, 1999.
- [5] Hopcroft, J.E. and Ullman, J.D. "Introduction to automata theory, languages, and computation", Addison-Wesley, 1979.
- [6] Murata, T., "Petri nets: Properties, analysis and applications", Proceedings of the IEEE, Volume: 77, Issue: 4, 1989
- [7] David, R. and Alla, H., "Petri Nets and Grafcet", Prentice Hall, 1992.
- [8] Johansson, K.H. , "Hybrid Control System", in Control Systems, Robotics, and Automation , from Encyclopedia of Life Support Systems (EOLSS)
- [9] Johansson, K.H. and Lygeros, J. and Sastry, S., " Modeling of Hybrid Systems", in Control Systems, Robotics and Automation, from Encyclopedia of Life Support Systems (EOLSS)
- [10] Branicky, M.S. "Stability of Hybrid Systems", in Control Systems, Robotics and Automation, from Encyclopedia of Life Support Systems (EOLSS)
- [11] Schaft, A. van der and Schumacher, H., "An Introduction to Hybrid Dynamical Systems", Lecture Notes in Control and Information Sciences, 251, Springer, 2000
- [12] Liberzon, D., "Control Using Logic and Switching, Part I: Switching in Systems and Control", Handout notes at Workshop CDC 2001.
<http://www.ece.ucsb.edu/~hespanha/cdc01/switchingcdc01.htm>
- [13] Oppenheim, A. V. and Willsky, A. S. and Nawab, S. H. "Signals & Systems", Prentice Hall, 1997.
- [14] Hespanha, João P. Lecture notes on Hybrid Control and Switched Systems, Reachability, University of California at Santa Barbara, 2005.