LUND UNIVERSITY

Department of
AUTOMATIC CONTROL

Real-Time Systems

Exam August 29, 2019, hours: 8.00–13.00

Points and grades

All answers must include a clear motivation and a well-formulated answer. Answers may be given in **English or Swedish**. The total number of points is 25. The maximum number of points is specified for each subproblem.

Accepted aid

The textbooks Real-Time Control Systems and Computer Control: An Overview - Educational Version. Standard mathematical tables and authorized "Real-Time Systems Formula Sheet". Pocket calculator.

Results

The result of the exam will become accessible through LADOK. The solutions will be available on WWW:
http://www.control.lth.se/course/FRTN01/

1. Consider the continuous-time system

$$\frac{dx}{dt} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u$$
$$y = \begin{pmatrix} 1 & 0 \end{pmatrix} x$$

a. Sample the system using ZOH and sampling period $h = 2$. (1 p)

b. Compare the poles of the continuous-time and discrete-time systems. Did you expect those discrete-time poles? Why? (1 p)

2. Consider the two SISO controller structures in Figure 1 and Figure 2.

a. How do you motivate a change from the structure in Figure 1 to the structure in Figure 2? (1 p)

b. What is the transfer function from r to y in Figure 2? (1 p)

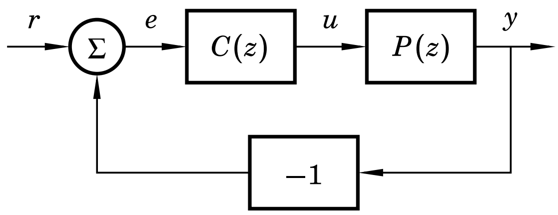

Figure 1 The old structure in Problem 2.

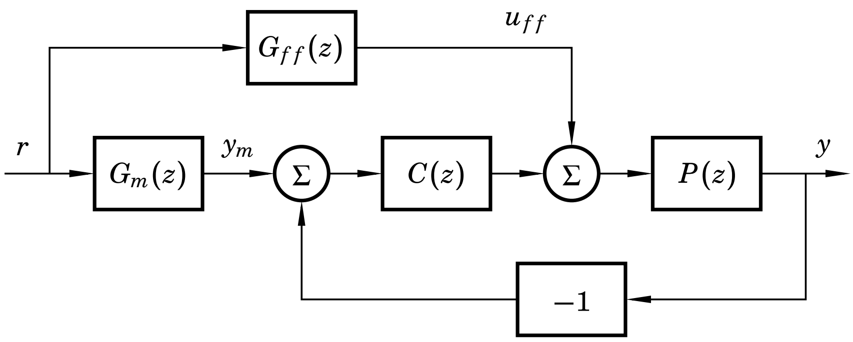

Figure 2 The new structure in Problem 2.

3. Given the following continuous-time system:

$$\dot{x}(t) = \begin{pmatrix} -6 & 1 \\ -9 & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u(t)$$
$$y(t) = \begin{pmatrix} 1 & 0 \end{pmatrix} x(t).$$

a. Compute the continuous-time transfer-function and determine if the system is stable. (1 p)

b. Approximate the continuous-time transfer-function with a discrete time pulse-transfer function using forward difference and $h = 1$. Determine if the discrete-time system is stable. (1 p)

c. Now approximate the continuous-time system using Tustin instead. Determine the poles of the discrete-time system and whether it is stable or not. (1 p)

4. Consider the following discrete-time system

$$x(k+1) = \begin{pmatrix} 1 & h \\ 0 & 1 \end{pmatrix} x(k) + \begin{pmatrix} h^2/2 \\ h \end{pmatrix} u(k)$$
$$y(k) = \begin{pmatrix} 1 & 0 \end{pmatrix} x(k).$$

a. Design a state feedback controller $u(k) = -Lx(k)$. Choose the closed loop characteristic polynomial in such a way that the sampled dynamics with $h = 0.1$ s corresponds to a continuous-time double pole in $s = -1$. (1 p)

b. Design an observer in the form

$$\hat{x}(k+1 \mid k) = \Phi\hat{x}(k \mid k-1) + \Gamma u(k) + K\Big(y(k) - C\hat{x}(k \mid k-1)\Big)$$

such that the observer dynamics is twice as fast as the closed loop dynamics, i.e., corresponds to a continuous-time double pole in $s = -2$. (1 p)

c. Design a model and feedforward generator in such a way that

1. The states of the model are compatible with the process states.
2. The discrete-time model dynamics corresponds to a continuous-time double pole in $s = -2$.
3. The steady state gain from the command signal u_c to y should be 1.

(2 p)

d. Write pseudo-code for the calculations that should be implemented in the controller, i.e., the calculations needed to implement the state feedback, the observer, and the model and feedforward generator. Write the code so that the input-output latency is minimized, i.e. split up the code in two parts: CalculateOutput and UpdateState, where the amount of code in CalculateOutput is minimized. Do all possible pre-calculations in UpdateState. Your pseudo-code may contain matrix expressions involving both scalar and vector variables, e.g., it is OK to write $u = -Lx$ as a statement. Use $\hat{x}$ to denote the estimated state vector and x_m to denote the model state vector. (2 p)

5. The operation

$$x = a \cdot b$$

should be performed using a fixed-point implementation. In the operation, a is a constant with value 1.35. 8-bits signed variables should be used to represent x, a and b.

a. Choose an appropriate number of fractional bits for a and convert a into the corresponding fixed-point representation. (1 p)

b. Zero fractional bits are used to represent x and b. Complete the code skeleton below to compute x. In case of overflow, the result should be saturated. 16-bits variables may be used for intermediate results. (1 p)

```
#include <inttypes.h>
// Insert in the next two lines the
// results from the first subproblem
#define n ... // number of fractional bits
#define a ... // fixed-point representation of a

int8_t x, b;
// define more variables if needed
...

// assume b is initialized to a value (you don't need to writ
// the code for the initialization of b)
// write the code to compute x
...
x = ...
```

6. An engineer has implemented a real-time control system using three threads. The first one is the control thread, which must execute at a sampling rate of 4 milliseconds. The second one is a reference generator, which should update the reference value every 12 milliseconds. The third one is a user interface thread, which should update the plotters 50 times per second. The engineer has timed precisely the execution of the reference generator, 4 milliseconds, and of the user interaction thread, 9 milliseconds. However, she does not know exactly how much time the control thread takes to execute. Assume an ideal kernel.

a. What is the maximum controller execution time allowed if the task set is to be schedulable using Earliest Deadline First scheduling? (1 p)

b. Assume that the deadline D_i for each thread is equal to the period T_i. Assume that the controller execution time is 0.5 milliseconds and that all blocking due to inter-process communication can be ignored. Will the task set be schedulable using Earliest Deadline First scheduling? And using rate monotonic priority assignments? (2 p)

7. The Traffic Control Agency (T.C.A) are currently developing a new structure for the traffic light in a crossing. They have just heard about Petri nets, and the ability to represent real-time systems with it.

So far they have developed two Petri nets (see Fig. 3), one for each traffic light, that they wish to use. However, they wish to connect the two traffic lights with each other and have now turned to the Automatic Control community for help. The only specification they have so far is:

- The green lights should be mutually exclusive (only one can shine at every given moment).
- The original Petri net developed by the T.C.A must be part of the solution.

a. The first person T.C.A reached out to for help was Klas Theodor Hoppsan (K.T.H), who studied at an unknown university in Sweden. He came up with the solution shown in Fig. 4. Explain and illustrate whether or not his solution is:

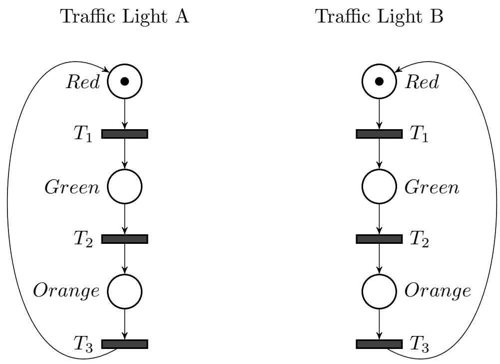

Figure 3 Petri Net over two traffic lights, developed by the Traffic Control Agency. The goal is to control the two traffic lights and assure that the green lights are mutually exclusive.

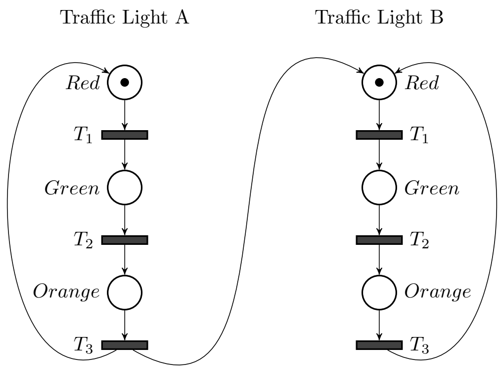

Figure 4 K.T.H's solution to the problem of the Traffic Control Agency.

- unbounded
- mutually exclusive (w.r.t. the green lights)

(2 p)

b. Disappointed with what K.T.H produced they now turn to you for help. Using the Petri net in Fig. 3 as a starting point, extend it and develop a new solution that is:

- mutually exclusive (the green lights)
- unbounded

(2 p)

8. Consider the following two Java classes:

```
public class MyObject {

    public int a = 2;
    public static int b = 1;
```

```
}

public class MyClass {

    public static void myMethod(int a, MyObject obj) {
        a = 0;
        obj.a = 3;
        obj.b = 9;
    }

    public static void main(String[] args) {
        int a = 3;
        MyObject ref1 = new MyObject();
        MyObject ref2 = new MyObject();
        a = 7;
        myMethod(a,ref1);
        System.out.println("Output 1 = " + a);
        System.out.println("Output 2 = " + ref1.a);
        System.out.println("Output 3 = " + ref2.b);
    }
}
```

a. What will printed on the screen when the class `MyClass` is executed, i.e., `> java MyClass`? (1 p)

b. Explain why this is the case. (2 p)