Department of Energy Technology

AALBORG UNIVERSITET

Matlab and Control Theory — INTRO 1st semester 2016

January 17, 2017

Written exam

09.30-13.30 CET (4 hours)

The set consists of 10 problems

Rules

- All usual helping aids are allowed, including text books, slides, personal notes, and exercise solutions.
- Calculators and laptop computers are allowed, provided all wireless and wired communication equipment is turned off. Internet access is strictly forbidden.
- Any kind of communication with other students is not allowed.

Remember

1. To write your study number on all sheets handed in.
2. It must be clear from the solutions, which methods you are using, and you must include sufficient intermediate calculations, diagrams, sketches etc. so the line of thought is clear. Printing the final result is insufficient.

Problem 1 (5 %)

1. Linearise the following expression and determine the linearisation coefficients:

$$f(x, y) = \frac{\sqrt{x}}{y}$$

The linearisation point is given by (x_0, y_0).

Problem 2 (10 %)

The system shown in figure 1 is considered. $G(s)$ is given by:

$$G(s) = \frac{130}{(s+2)(s+8)}$$

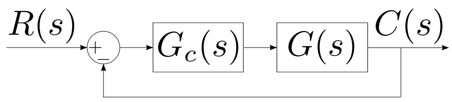


Figure 1: System for which the controller should be designed.

The Bode diagram for $G(s)$ is shown in figure 2.

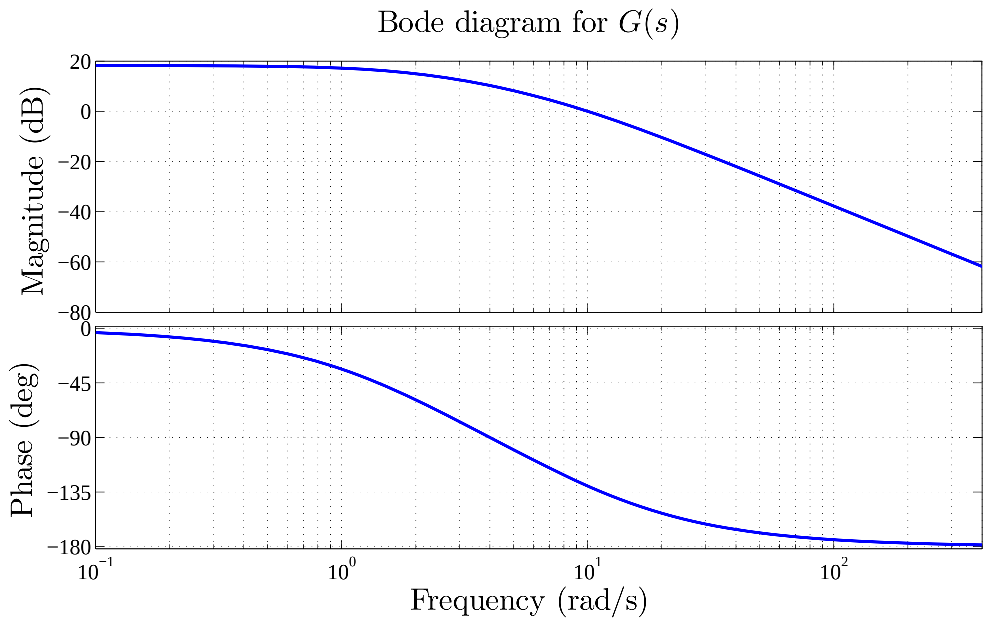


Figure 2: Bode diagram for $G(s)$.

1. Determine the gain- and phase margin for the system without the controller. Explain how these are found.
2. Design a suited controller, $G_c(s)$, for the systems, so the phase margin becomes $\phi_m \approx 45°$ and there is no steady state error for a step input. There is no need to improve the transient response.

Problem 3 (10 %)

A system has the Bode diagram approximation shown in figure 3.

1. Determine the transfer function for the system. Any second order systems in the transfer function may be assumed to have a damping ratio of $\zeta = 1$.

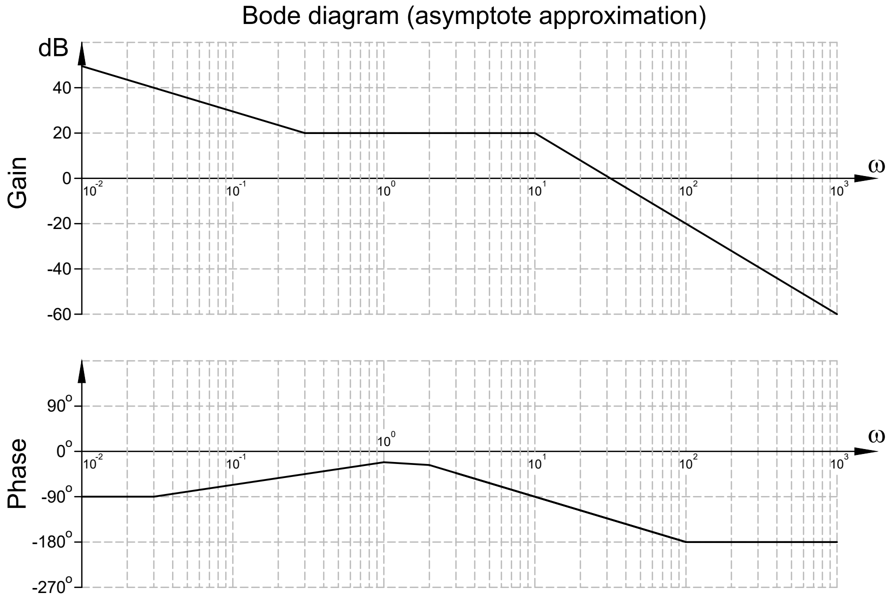


Figure 3: Bode diagram asymptote approximation.

Problem 4 (15 %)

The systems shown in figure 4 is considered, where a PD-controller is included.

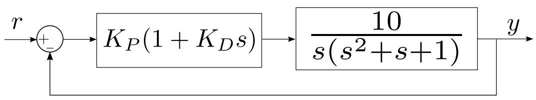
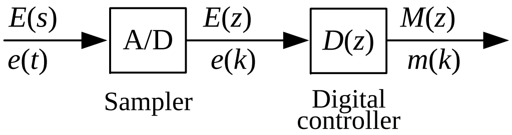


Figure 4: System considered in Problem 4.

1. Determine the steady state error for the system when applied a unit step input, i.e. $R(s) = \frac{1}{s}$. It may be assumed that the system is stable.
2. Determine the characteristic equation for the system.
3. Determine, using Routh-Hurwitz stability criterion, what has to apply for respectively K_D and K_P for the system to be stable.
4. It is now informed that $K_D = 1$. Sketch the root locus for the system, when it is the proportional gain K_p that is varied along the root locus. NOTICE, it is ***not*** sufficient just to use Matlab!

Problem 5 (13 %)

The figure below shows a continuous signal $e(t)$ fed to a digital controller through a sampler.



The sampling time is $T = 2$ seconds and the controller is represented by the difference equation

$$m(k) - 0.64m(k-1) + 0.8m(k-2) = e(k) - 0.25e(k-1),$$

where $m(k)$ and $e(k)$ are the output and input, respectively, at sampling instant k.

1. Find the discrete transfer function $D(z)$ that corresponds to this difference equation
2. Determine the order and type for the controller besides all zeros and poles
3. Does the controller have integral action?

Suppose now that the input error is

$$e(t) = 1 + \cos\left(\frac{\pi}{5}t\right).$$

4. Skecth $e(t)$ in the interval $0 \le t \le 10$ and calculate the sampled values $e(k)$ for $k = 0$ and $k = 1$
5. Is the given sampling time sufficient to avoid aliasing?
6. Find a formula for $E(z)$ using z-Transform tables

Problem 6 (13 %)

A continuous-time controller $C(s)$ is defined by

$$C(s) = \frac{M(s)}{E(s)} = K_c \frac{s - z_0}{s - p_0}$$

where K_c, z_0 and p_p are constant parameters.

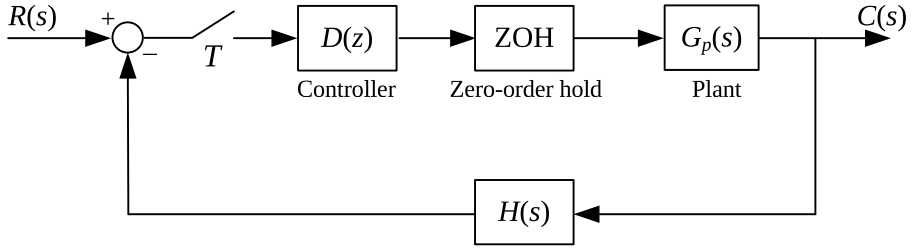
$C(s)$ must be converted to an equivalent discrete-time controller $C(z)$ using the Forward Euler method (*forward rectangular rule*). The sampling time is denoted T.

1. What is the gain at 0 Hz for the continuous-time controller $C(s)$?
2. What is the high-frequency gain (i.e. when the frequency becomes infinitely large) for $C(s)$?
3. Find an expression for $C(z)$ and list all zeros and poles for $C(z)$
4. For which values of the sampling time T is $C(z)$ stable?
5. For the specific case where $K_c = 1000$, $z_0 = -5$ and $p_0 = -30$ which sampling time T will you recommend to use?
6. Write a MATLAB script that can plot the impulse response for $C(s)$ and $C(z)$.

   Note: you do not have to show any results
7. List the general advantages and disadvantages for the "design by emulation" method

---

Problem 7 (14 %)



The figure shows a closed-loop control system having the plant transfer function

$$G_p(s) = \frac{10}{s^2}$$

Also, $H(s) = 0.1$.

The discrete controller is a simple proportional gain $D(z) = K$, where K is a positive constant. The sampling time is $T = 2$ seconds.

1. Calculate the Nyquist frequency – and explain why is this important information to have?
2. Find an analytical expression for the pulse transfer function $G(z)$
3. Determine the characteristic equation
4. Plot all poles and zeros in a diagram and sketch the root locus for the system

   Note: You do not need to calculate precise values for breakaway points, asymptotes, etc. — just a rough sketch
5. Discuss the stability properties for the closed loop system
6. Now, a bright student suggests to reduce the sampling time to $T = 2 \cdot 10^{-3}$ second (2 milli-seconds). Still using a proportional controller $D(z) = K$, how does this reduction of the sampling time effect the stability properties for the closed loop system?

Problem 8 (5 %)

In ski jumping, points for style are awarded by five judges from five different countries. Maximum each judge can give is 20 points. To be extra fair, the highest and the lowest scores are always discarded and the remaining three scores are summed. Write a function that will perform this action given a vector of five points.

Examples:

17.5 18.5 19.0 18.5 19.0 → 18.5 + 18.5 + 19.0 = 56.0
16.5 17.5 17.5 18.0 17.0 → 17.5 + 17.5 + 17.0 = 52.0
19.0 20.0 16.0 19.0 18.5 → 19.0 + 19.0 + 18.5 = 56.5

```
>> skijumpingscoring([19 20 16 19 18.5])

ans =

56.5
```

Problem 9 (5 %)

Write a **polynomialfit(x,y)** function that will accept any x and y data vectors and will output coefficients of a polynomial that fits these points exactly. Example:

```
x = [0.5079    0.0855    0.2625    0.8010]
y = [0.0292    0.9289    0.7303    0.4886]
```

should yield

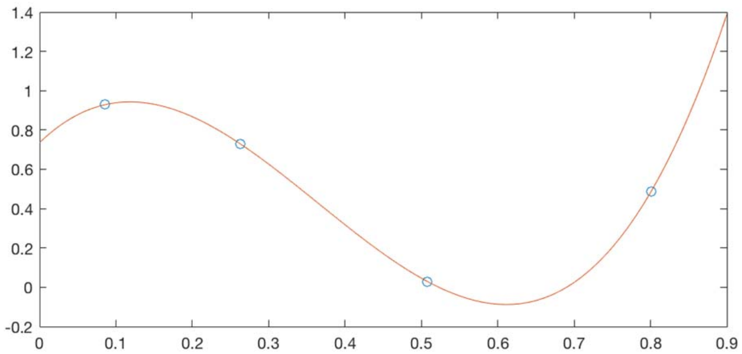
```
>> polynomialfit(x,y)

ans =

17.2253  -18.8516    3.7391    0.7362
```

corresponding to

$$y = 17.2253 \cdot x^3 - 18.8516 \cdot x^2 + 3.7391 \cdot x + 0.7362$$



Problem 10 (10 %)

Implement a bubble sort algorithm in a MATLAB function. After Wikipedia: “Bubble sort is a simple sorting algorithm that repeatedly steps through the list to be sorted, compares each pair of adjacent items and swaps them if they are in the wrong order. The pass through the list is repeated until no swaps are needed, which indicates that the list is sorted.”

```
>> bubblesort([3 7 9 5 1])

ans =

1      3      5      7      9
```
