Lecture Energy Systems: Hardware and Control - Control Part
University of Freiburg – Winterterm 2017/2018

Exercise Sheet 1

Prof. Dr. Moritz Diehl, Dr. Gianluca Frison and Benjamin Stickan

For questions on the exercise please contact Benjamin Stickan (`benjamin.stickan@imtek.uni-freiburg.de`)

Within the control part of the "Energy Systems: Hardware and Control" course there will be 45 min. exercise sessions after each lecture. The exercises are guided by tutors and will contain some **MATLAB**-based tasks. Therefore, a **MATLAB** installation including the Control System Toolbox is needed.

Getting started

1. Except **MATLAB** is not yet installed on your computer, the first thing you need to do is install it. Detailed installation and licensing instructions can be found at
`https://www.rz.uni-freiburg.de/services-en/beschaffung-em/software-en/matlab-license`
Remember that the Control System Toolbox is required.

2. If you are new to **MATLAB**, the first thing you will appreciate is the extensive help system. You can simply type `doc` into the console and the documentation opens. If you type `doc plot`, you will find a detailed description of function `plot`.

3. Here are some useful commands for the exercises:
```
hold on/off
figure
close all
clear
clc
```

Problem 1: Dynamical System, ODE, Simulation and Solution

A simple pendulum is sketched in figure 1. The point-mass m is fixed to a solid, massless rod of length l, which is connected to a frictionless hinge on the other side. All movements take place in the vertically oriented x-y-plane and the gravitation g acts in y-direction.

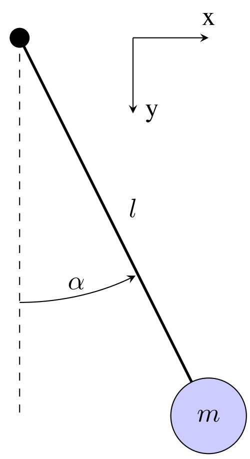


Figure 1: Sketch of a simple pendulum

(a) Derive the equation of motion for the pendulum and note it in the shape $\ddot{\alpha} = f(\alpha)$. How do the mass or the length determine the motion of the pendulum?

(b) What are the states x, which are needed to completely describe the system?

(c) Convert the ODE to the system of equations $\dot{\mathbf{x}} = f(\mathbf{x})$.

(d) Simulate the motion of the pendulum for 10 seconds using different initial values x_0. Therefore write a function `function dx = nonlin_pendel(t, x)` which implements the system of equations. For the simulation use `lsode` and the following constants: $l = 1\,\text{m}$, $g = 9.81\,\frac{\text{m}}{\text{s}^2}$.

(e) What characterizes steady states? Calculate the steady states for the pendulum.

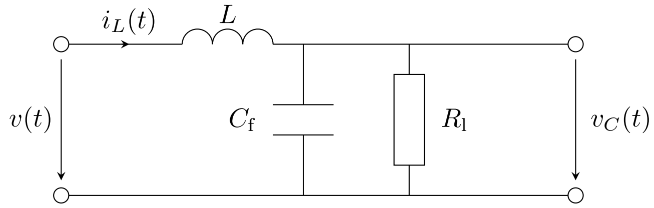
(f) Linearize the system at the steady state $x_{\text{ss}} = [0, 0]^\top$ and write the system of equations in the shape $\dot{\mathbf{x}} = A\mathbf{x} + Bu$ and $y = \alpha = C\mathbf{x}$. Compute the state space matrices A,B and C.

(g) Compare the linear and the nonlinear system via simulations using increasing initial values for $\alpha(0)$ ranging from $\pi/8$ to π.

Problem 2: Buck-converter, Modelling and Stabilization

1. The electrical circuit sketched below shows a simplified buck-converter with a constant load at the output. The system can be described in state-space representation as

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u, \quad y = \mathbf{C}\mathbf{x}, \quad \mathbf{D} = [0].$$



   (a) Derive the the I/O-ODE (Input/Output-Ordinary Differential Equation) for the given circuit using equations

$$i_C = C_\mathrm{f}\frac{dv_C}{dt}, \quad v_L = L\frac{di_L}{dt} \quad \text{and} \quad i_R = \frac{v_C}{R_\mathrm{l}}$$

   (b) Convert the I/O-ODE to state space representations, i.e. set up the $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$-matrices for

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \quad (1)$$
$$y(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}u(t) \quad (2)$$

   using

   (i) the control canonical form.

   (ii) the observer canonical form.

   (c) Now derive matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ for the state vector given as $\mathbf{x} := \begin{bmatrix} i_L & v_C \end{bmatrix}^\mathsf{T}$, input $u := v$ and output $y := v_C$.
   (Hint: Use Kirchhoff's voltage law for inductors and current law for capacitors)

   (d) Derive the characteristic polynomial. Evaluate the eigenvalues of the system for $L = 4.7$ mH, $C_\mathrm{f} = 100\ \mu$F and

   i. $R_\mathrm{l} = \infty\ \Omega$

   ii. $R_\mathrm{l} = 100\ \Omega$

   Is the system BIBO-stable in both cases?

   (e) Write down the time constant τ in seconds and the resulting oscillating frequency in Hz for both values of R_l.

   (Hint: In this example, the time constant τ is a measure for the amplitude decay (damping) and is defined as $\tau = -\frac{1}{\mathrm{Re}(\lambda)}$. The oscillating frequency is defined as $f_0 = \frac{\omega_0}{2\pi} = \frac{|\mathrm{Im}(\lambda)|}{2\pi}$)

   (f) Create a new MATLAB script and define variables L $= 4.7$ mH, Cf $= 100\ \mu$F and Rl $= \infty$. Also define matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D} = 0$ according to task (1c).

   (g) Use the `ss(A,B,C,D)` command to create a state-space model `sys_ol` and evaluate the systems step response with the `step(sys,Tfinal)` function (Tfinal = 0.1 s) for

   i. $R_\mathrm{l} = \infty\ \Omega$

   ii. $R_\mathrm{l} = 100\ \Omega$

   (h) Is the system controllable and/or stabilizable?

2. Now we want to introduce a state feedback with gain K to stabilize the system in the case where no load is connected ($R_\mathrm{l} =$ inf).

   (a) Where do the two poles have to be shifted to obtain the following characteristics for the closed-loop system?: $\tau = 10$ ms, $f_0 = 100$ Hz

   (b) Use the MATLAB function `place(A,B,p)` to calculate the corresponding feedback vector K and implement the system matrix $\mathbf{A}_\mathrm{cl}$ as well as the closed-loop model `sys_stable` for the stabilized system.

   (c) Simulate `sys_stable` with the `step()` command and verify frequency and damping is as desired.

Problem 3: Stability, Output Feedback, Root Locus

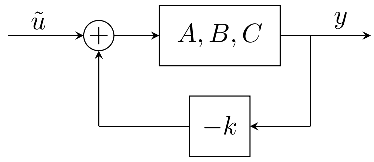
Consider the following system given by:

$$A = \begin{bmatrix} -1/4 & 1/4 & 0 \\ 0 & -1/5 & 2/5 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}, \quad D = [0]$$

(a) Compute the characteristic polynomial.

(b) Is the system BIBO stable and why?

Now a proportional feedback control loop with gain $-k$ is implemented as follows



(c) Write down the system equation and the output equation and add the feedback block.

(d) Transform the equation(s) to the standard steady state representation with $\tilde{u}(t)$ as input, i.e.

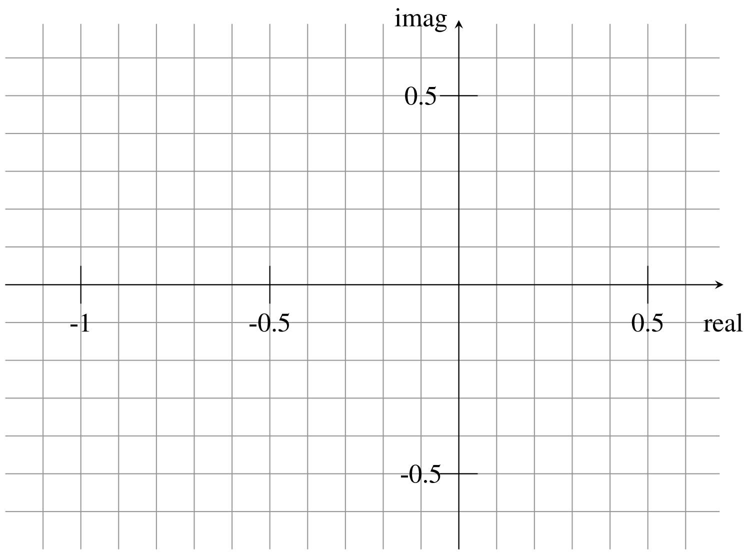
$$\dot{\mathbf{x}}(t) = \tilde{A}\mathbf{x}(t) + B\tilde{u}(t) \qquad (3)$$
$$y(t) = \tilde{C}\mathbf{x}(t) \qquad (4)$$

Identify the system matrices $\tilde{A}$, $\tilde{B}$ and $\tilde{C}$ for the state space representation of the closed loop dynamics.

(e) Calculate the characteristic polynomial of the closed loop system.

(f) Compute the roots (using Octave/MATLAB) for $k \in \{0.005, 0.008, 0.05, 0.1, 2\}$. For which values is the closed loop asymptotically stable (i.e. the transfer system is BIBO stable)?

(g) Draw a diagram by marking these roots in the complex plane with colored crosses (x)



(h) Connect the roots to trajectories (with k as parameter) and give an interpretation of the system dynamics (damping, oscillations).

(i) Verify this behavior by plotting the step responses (MATLAB) for the k values given in e)

(j) Try the rlocus() function of MATLAB by setting up the A,B,C (open loop) system.