Assignment -2

General Problem

The Pendulum-Cart system defines a classic problem in dynamics and control theory and is widely used as a benchmark for testing control algorithms. A simplified variation of this system is illustrated in Figure 2 where a mathematical pendulum is connected to a pair of only-rolling wheels with one translation degree of freedom. This models the Wheeled Inverted Pendulum system.

In the simplified model, the point mass mp is connected to the cart with the mass mc via a massless arm with the length L. q1 is the displacement of the cart and q2 is the angle of the pendulum. As input, u is assumed to be a force.

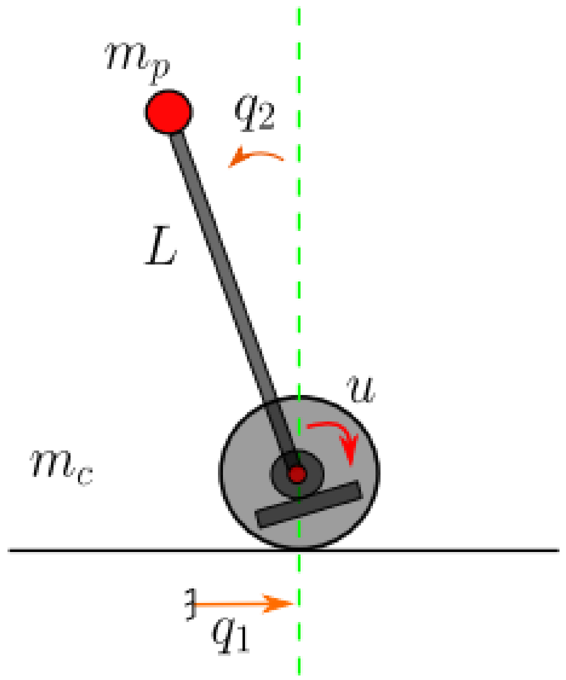

Abbildung 2: Pendulum-Cart System

In our exercises, we will handle the stabilization problem around the upper position of the pendulum. We will only consider the linearised model. The overall objective will be broken down into the following tasks:

- ☐ Creating the dynamical system using MATLAB.
- ☐ System Analysis, controller design using MATLAB tools and simulation.
- ☐ Design of state-space and optimal controllers and simulation.

- ☐ Design of state observers and investigating the closed loop including an observer
- ☐ . Time-discrete controller and observer design.

Dynamic Model of the System

The equations of motion for the described system can be derived using different methods such as Lagrange-Formalism or Newton-Euler. The result is in any case a nonlinear system in the following form:

$$\boldsymbol{M}(\boldsymbol{q})\,\ddot{\boldsymbol{q}} + \boldsymbol{h}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \boldsymbol{g}_q\,\boldsymbol{u}$$

with

$$\boldsymbol{M}(\boldsymbol{q}) = \begin{pmatrix} m_c + m_p & -L\,m_p\,\cos q_2 \\ -L\,m_p\,\cos q_2 & L^2\,m_p \end{pmatrix}$$

$$\boldsymbol{h}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \begin{pmatrix} L\,m_p\,\dot{q}_2^2\,\sin q_2 \\ -L\,g\,m_p\,\sin q_2 \end{pmatrix} + \begin{pmatrix} d_1\,\dot{q}_1 \\ d_2\,\dot{q}_2 \end{pmatrix}\ ,\ \boldsymbol{g}_q = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

where d_1 and d_2 are the so-called *damping factors* representing friction in the cart displacement and the joint respectively.

A state-space representation can be derived using the equations above defining $\boldsymbol{x} = \begin{pmatrix} \boldsymbol{q} \\ \dot{\boldsymbol{q}} \end{pmatrix}$

$$\dot{\boldsymbol{x}} = \boldsymbol{F}(\boldsymbol{x}) + \boldsymbol{G}(\boldsymbol{x})\,u = \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ -\boldsymbol{M}^{-1}(\boldsymbol{q})\,\boldsymbol{h}(\boldsymbol{q},\dot{\boldsymbol{q}}) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \boldsymbol{M}^{-1}(\boldsymbol{q})\,\boldsymbol{g}_q \end{pmatrix} u$$

Linearising about the operation point $q_1^* = q_2^* = 0$ yields the linear state-space equation for the system

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\,\boldsymbol{x} + \boldsymbol{B}\,u \tag{1}$$

with

$$\boldsymbol{A} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \dfrac{g\,m_p}{m_c} & -\dfrac{d_1}{m_c} & -\dfrac{d_2}{L\,m_c} \\ 0 & \dfrac{g\,(m_c+m_p)}{L\,m_c} & -\dfrac{d_1}{L\,m_c} & -\dfrac{d_2\,(m_c+m_p)}{L^2\,m_c\,m_p} \end{pmatrix}\ ,\ \boldsymbol{B} = \begin{pmatrix} 0 \\ 0 \\ \dfrac{1}{m_c} \\ \dfrac{1}{L\,m_c} \end{pmatrix}.$$

We the following values:

$$m_c = 1.5,\ m_p = 0.5,\ g = 9.82,\ L = 1,\ d_1 = d_2 = 0.01$$

Note: If you want to know how these equations are derived,you can search for this on web.

Task 1 :

Within the first task, we analyse the system and compare its different representations.

1. Within a MATLAB script, build the state-space model of the system using the matrices **A, B** and proper matrices **C** and **D** by the command **ss**! Suppose the angle q2 as the system output!

2. Use Matlab commands like **pole, eig** and **zero** to analyse the system. Is the system stable? Is it minimal-phase? Draw the zero-pole-map of the system using **pzmap**! Are the results different? Why?

3. Convert the system into the corresponding transfer function using **tf**! How can you do the same using **ss2tf**? Create also the **Zero-Pole-Gain** representation of the system by converting using **zpk.**

Task 2:

In this task, we consider the results from the task 1 to continue with system analysis and a first controller design. Use your MATLAB-function from the previous exercise to regenerate the system.

1. Reconstruct the system matrices A, B, C and D from the state-space system generated by the function from the previous task using **ssdata**! The output of the system is still **q2**.

2. Create a SIMULINK model and simulate the system dynamics using an **integrator** and the matrices as **gain-blocks**!

3. Use rlocus to plot the root locus diagram of the system! Can a P-controller stabilize the closed loop? Why?

4. Open the **SISO-TOOL** for the system. Try to find a controller using an integrator, a real pole and a complex zero pair to stabilize the system! Consider minimizing the rise time in the step response of the closed loop and the maximum value for the input! Export the controller into the workspace to use it in the simulation!

5. Create the closed loop system using feedback and analyse the stability.

6. Add the designed controller to your SIMULINK model and simulate the closed loop! Set the Initial value for the pendulum angle $q2(t = 0) = 10^{\circ}$! Does the output convert to zero as desired? Do all of the states convert to zero?
If you simulate the closed loop long enough (e.g. 20 s or longer), it seams that something is going wrong. How would you explain that? What should change to avoid this problem? What is the practical solution?

Note: Remember to use **help** command in matlab to know about any function or see the documentation.