Technische Universiteit Eindhoven University of Technology

Design Assignment

Policy Learning for an Unbalanced Disc

5SC28 Machine Learning for Systems and Control

Authors:
Maarten Schoukens, Roland Tóth, Gé van Otterdijk, and Máté Kiss

April 28, 2025

Contents

1 Introduction

The unbalanced disc as depicted in Figure 1 acts as a pendulum. The primary objective of this design assignment is to obtain a controller for the system that can swing up the pendulum and keep it stable at the top position. From a mechanical point of view, the system behaves as an inverted pendulum. Systems that exhibit dynamics similar to an inverted pendulum are quite a common sight in the streets nowadays: segways, hover boards, and (electric) unicycles are all examples of inverted pendulum systems.

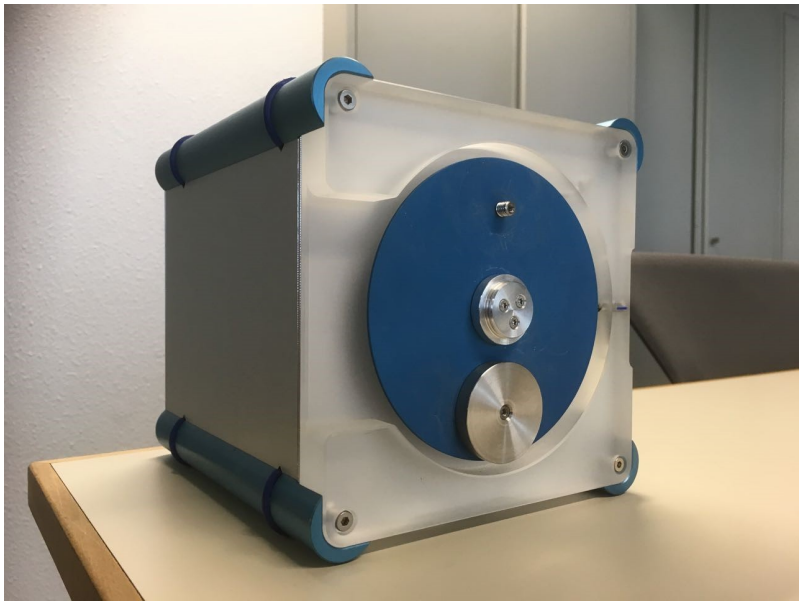

Figure 1: The unbalanced disc setup.

In recent years, several machine learning methods have been introduced based on, for example, Gaussian processes and neural networks. These approaches can be seen as general function approximators which can be trained to capture unknown relationships directly form data. This training process can be seen similar to learning, in which the corresponding AI algorithm gradually builds up a model of the relationship by observing incoming data or results of experiments (interactions with a given system). These methods can also be used to learn a map from response of the system to actuation inputs in order achieve a desired task, like swinging up and stabilizing a pendulum connected to a DC motor. While most control design methods rely on the knowledge of the system dynamics, i.e. they are model based control methods, a learning based approach can obtain a controller of a system in an automated way without any user interaction.

2 First Principles Model

The ideal motion dynamics of the inverted pendulum can be modeled as:

$$\begin{aligned} \dot{\theta}(t) &= \omega(t), \\ \dot{\omega}(t) &= -\omega_0^2 \sin\left(\theta(t)\right) - \gamma\omega(t) - F_c \text{sign}(\omega(t)) + K_u u(t), \end{aligned} \tag{1}$$

where θ is the angle of the disk in radians with respect to the bottom position (measured clockwise), ω is the angular velocity of the disk in rad/s and $u(t)$ represents the system input in volt. Both θ and ω are measured in the available system setup. Furthermore, in eq. (1), ω_0 is the base frequency of the pendulum in rad/s, γ the dynamics friction coefficient, F_c the Coulomb friction coefficient and K_u the input proportionality constant. The system input voltage is software saturated. Throughout this design assignment, *the input voltage should be limited between -3 and +3V*. This results in an underactuated setup which makes the control problem somewhat more challenging.

3 System Setup, Simulation and provided data

The files which will be needed to complete the assignment are available at https://github.com/GerbenBeintema/gym-unbalanced-disk. On that page you can find

- the provided benchmark datasets which will need be used for identification.
- Simulators of the unbalanced disc in the form of python scripts, MATLAB scripts and a Simulink (all implementations are equivalent)
- Instructions on how to connect to the setup through either python or MATLAB.

Due to the large number of students, only limited access is available for the students to perform real-life experiments on the unbalanced disc system. Therefore, a computer simulation of the unbalanced disk is implemented in Simulink (requires Matlab R2018b), Matlab and Python. These simulation environments will be used for some of the design assignment problems.

The setups will be available in Flux on reservation. You need to reserve the timeslot to access the setups. These timeslots will need to be reserved through Canvas. Do not book more than 2 timeslots ahead in time to make sure that all groups have the opportunity to access the setups. The exact guidelines on where to obtain a setup for use in Flux only and from when on you can access them will be communicated through Canvas.

4 Problems

4.1 Modeling the System Dynamics

50% of the project grade (25% for the ANN part + 25% for the GP part).

The first part of the assignment is to identify the system dynamics using a Gaussian Process (GP) model *and* a (Deep) Artificial Neural Network (ANN) model. Regarding the representation and parametrization of the model structure, NARX and NOE input-output models, a nonlinear state-space and other types of model structures can be considered to capture the system dynamics. For the GP model, you are asked to fit and analyze the obtained model using only one of these model structures. For the ANN models, we expect you to obtain at least one result using the more simple approaches (e.g. NFIR, NARX, NOE), and one result that makes use of more advanced learning architectures. In all cases, we expect a clear systematic approach and motivation in obtaining the model.

The considered system input is the applied voltage to the system motor. The noisy output is the measured disk angle. Do not use the angular velocity while solving this problem.

The Matlab toolboxes and Python scripts that were used during the practical sessions are recommended be used to complete this objective. Next to the options discussed during the lectures and the practical sessions, the students are free to use any other toolbox or coding environment if desired. The final result is evaluated in terms of how the study goals listed below are achieved.

The data that can be used for this exercise can be obtained through aforementioned github link. The final model needs to be tested on the provided test datasets using the provided test functions. Make sure to validate / test your models using both the prediction and simulation task. The final obtained test outputs need to be shared with the lecturers together with the final project submission.

Finally, note that the
Study goals:

- Apply the learned concepts of GP and ANN based modelling on an application example under realistic experimental conditions and master the use of available software tools. Be

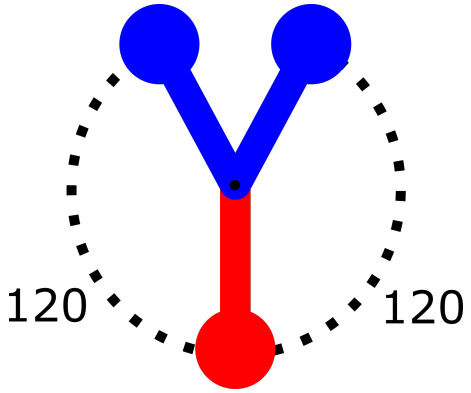

Figure 2: Range of interest for the identified pendulum model.

able to illustrate this by achieving high-quality model estimates starting from a dataset of a physical system.

- Understand and apply the tuning of hyper-parameters and assess the resulting accuracy of the model w.r.t. data from a physical system.
- Understand and apply the system identification cycle (Lecture 1) on data from a practical setup.

4.2 Learning a Policy

4.2.1 Learning a Swing-Up Policy

40% of the project grade (20% for the Q-learning part + 20% for the actor-critic / model internalization part).
This problem should first be solved using the simulation environment, and can be applied on the real-life system in a second step.

The second part of the design project is to obtain a policy (controller) that can swing up the pendulum starting from the stable bottom position and keep the pendulum at the target upright position by rejecting disturbances (see Figure 3). Both the measured angle and angular velocity can be used to solve this problem (i.e. full state feedback).

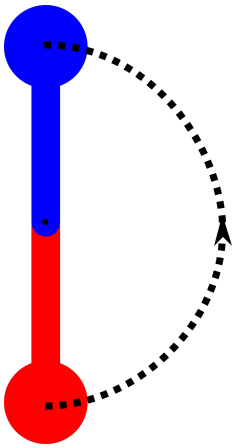

Figure 3: Learning a policy for pendulum swingup.

The students are expected to apply both Q-learning based and actor-critic or model internalization based reinforcement learning methods on the unbalanced disc setup (start on the simulator,

next, move to the real-life setup). Explore the advantages and disadvantages of the applied techniques.

Note: the Matlab toolbox PILCO is available through https://github.com/UCL-SML/pilco-matlab .

Study goals:

- Apply Q-learning based reinforcement learning (vanilla Q-Learning, basis function expansion Q-Learning, or DQN) on an application example under realistic experimental conditions and master the use of available software tools.
- Apply the learned concept of reinforcement learning (model internalization or actor-critic methods) on an application example under realistic experimental conditions and master the use of available software tools.
- Understand the tuning of the learning algorithm and the resulting convergence together with the final performance of the control policy w.r.t. a physical system.

4.2.2 Learning a Single Policy for swing up and reference tracking

10% of the project grade.
This problem should first be solved using the simulation environment.

The final part of the assignment is to extend the swing-up policy to also track a reference while the disc is in the top position using actor-critic or model internalization based approaches. Design a policy that can move the pendulum from the bottom position to the upright position, and that can track a reference motion of +- 15 degree around that top position (see Figure 4). Both the measured angle and angular velocity can be used to solve this problem. *Note: a switching controller combining multiple policies trained on a single target is not considered to be a valid solution to this problem.*

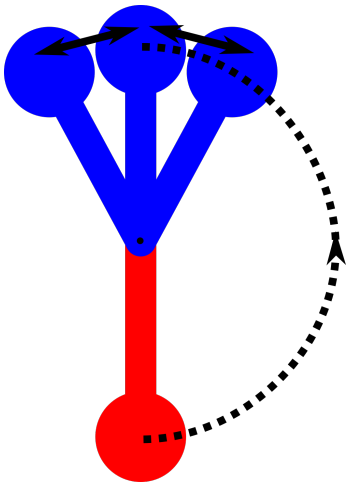

Figure 4: Learning a single policy for multiple targets.

Possible solution: extend the PILCO toolbox to handle an externally provided target (see [1]). Other solutions include extending the actor-critic cost and experimentation to obtain a single a reference tracking controller.

Study goals:

- By developing an extension of the simple one-target policy, the student is expected to mastering the theoretical concepts of the advanced RL techniques in more detail.

- By being able to modify the existing software tools, a good understanding of the algorithmic implementation of the applied reinforcement learning tools is obtained.

5 Grading

The grading of the course is explained in the study guide which is available through Canvas. Note that the diversity of the obtained solutions is taken into account (e.g. only obtaining ANN NARX modelling solutions covers only a limited part of the course material). Also a clear motivation of the modelling and reinforcement learning choices is requested.

All group members are expected to have participated in both the modelling and the reinforcement learning part of the assignment.

References

[1] M. P. Deisenroth, P. Englert, J. Peters and D. Fox, Multi-task policy search for robotics, 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 2014, pp. 3876-3881.