Lund University
Automatic Control
FRTN70 Project in Systems, Control and Learning
Spring 2021

Reinforcement Learning for Double Pendulum

Project Plan Group H

Cem Alpturk[1] Pukashawar Pannu[2]

Project Advisor: Emil Vladu

[1]ce5368al-s@student.lu.se
[2]pu6218pa-s@student.lu.se

1 Project Purpose

The goal of this project is to teach a model to control a double pendulum on a cart to be standing upright, in it's equilibrium point, using reinforcement learning.

2 Equipments and Material

This project does not require any physical equipment. It will be fully simulated.

3 Modelling and System Design

Focus of the project is to train a controller for the dobule pendulum and not to model the actual double pendulum on a cart.

4 Division of Labour

We have decided to split our project into three parts: Setting up Environment, Train the Agents, Presentation.

4.1 Setting up Environment

In order to teach an agent to control a double pendulum we need to simulate it. This section is about setting up the requirements for the training phase. Equations for the double pendulum on a cart will be researched online and from other available sources and implemented using Python. Model will be verified by comparing with online models and an attempt will be made to animate the results to see if it looks as expected.

4.2 Training the Agents

This will be the main focus of the project, which is to create models and train them to control the double pendulum. The training will utilize reinforcement learning where the agent will learn from experience by trial and error. If the agent manages to keep the pendulum upright for a long enough time, the time has not been decided as of yet, it is considered to have solved the problem.

We have decided to split our project into three parts: image analysis, hardware and servo position implementation, and finding ball trajectory and general code structure.

4.3 Image analysis

The image analysis is a major part of our project. Without it working, we will not have a reference for the ball position. Therefore, if the image analysis is not working, we will not have a working project. We will be using two cameras (see Section 2) to determine the position of the ball.

Olle Flitig and Sara Rask will be in charge of this section. We decided that it was a two person job since it is a quite daunting task.

4.4 Hardware and servo position

We will try to control the position of the servos based on a 3D point (which tells us where the ball is supposed to land). Most of the hardware implementation and the servo control will be handled by this package. Björn Duktig is in charge of this section.

4.5 Ball trajectory and general code structure

We will try to estimate the final position of the ball based on a few Cartesian points (identified by the image analysis section). Furthermore, the general structure of the code will be handled by this task (e.g. the observer-observable implementation and the semaphores in the monitor). Amanda P. Solver is in charge of this section. It is also planned that she will aid in the "optimal" trajectory estimation for the robot arm (Hardware section).

5 Time Plan

5.1 Subtasks

- Create a Monitor package including classes used by the image analysis and the hardware systems. Implement Observer-Observable and real time behaviour. **Estimated deadline**: 31/3
- Assemble the robot arm. **Estimated deadline**: 9/4
- Set up a display in such a way that the cameras and the robot arm have stationary positions. **Estimated deadline**: 14/4
- Implement communication between arduino and PC. **Estimated deadline**: 21/4
- Find mathematical model for how the ball trajectory should be estimated. Implement this into the ballTrajectory package. **Estimated deadline**: 29/4
- Find a model for the arm trajectory. **Estimated deadline**: 29/4
- Create a package for how the servos should respond to the given arm trajectory. **Estimated deadline**: 6/5
- Perform image segmentation to find the ball in the two cameras. **Estimated deadline**: 13/5
- Make a computer vision system to map the ball segments in the images to a 3D-coordinate. Including calibration and on-line calculations. **Estimated deadline**: 13/5
- (Optional); Implement feedback control of robot arm position **Estimated deadline**: N/A

5.2 Important dates

- Mar 29 - Hand in project plan.
- Apr 22 - Feedback seminar 1 on the modeling and design.
- May 5 - Report should be pushed to git to allow peer review by other groups.
- May 11 - Peer review done
- May 12 - Feedback seminar 2 on the design and implementation.
- May 20 - Project done and demonstrated and final report handed in
- May 27 - Demo film upload and peer review of final report done
- June 3 - Final presentation and demonstration and revised final report handed in