Master's thesis proposal: Modeling of a two-link mechanical manipulator for studying Reinforcement Learning algorithms

Reinforcement learning (RL) has received considerable attention during the last years via the rapid progress within deep learning and related fields. Somewhat simplified, RL is a method where a suitable way to control a system is learnt via repeated trials using the real system or a simulator, and the performance of the control system is evaluated using a reward or cost function.

This is the first project in a series of Master's thesis projects to implement, test, and evaluate one or several approaches to RL on a two-link manipulator, where the task is to make the robot follow a given path. The aim of the current project is to model a two-link rigid manipulator mathematically and implement the so-called environment(s) in Python. The environment consists of i) a dynamical model of the two-link rigid manipulator where all four states (two joint angles and two angular velocities) are measured and given as the output and ii) a cost function which quantifies how far the current state of the manipulator is from the path. This is version V0 of the environment. More complex environments can be defined in the following ways (depending on the time available)

- V1- Errors in the measurements of the state variables.
- V2- Only the joint angles can be measure, but without errors.
- V3- Errors in the measurements of the joint angles.
- V4- Mechanical flexibilities on the joints.
- V5- External disturbances.
- ………………………………

The project will consist of the following main parts:

- Literature study
- Mathematical modeling of two-link manipulator (for example, using Modelica).
- Implementing the task as environment V0 and integrating it with Gym. (Gym is a standard Python library for developing and comparing reinforcement learning algorithms in python.)
- Implementation of a classical control approach to accomplish the problem objective or if time allows, trying an RL approach (RL part is optional).
- Implementing other environments (optional, see V1-V5 above).
- Overall evaluation and documentation.

The project will require the following background:

- Solid background in automatic control, represented by the courses TSRT92, Modellering och inlärning av dynamiska system, TSRT09 Reglerteori, and TSRT08 Optimal styrning (valuable but not required).
- Solid background in Python.

Further information:

Farnaz Adib Yaghmaie, farnaz.adib.yaghmaie@liu.se

Svante Gunnarsson, svante.gunnarsson@liu.se