Reinforcement learning based control of an underactuated double pendulum system

Master's Thesis Nr. xxx

Scientific Thesis for Acquiring the Master of Science Degree
at the School of Engineering and Design
of the Technical University of Munich.

Thesis Advisor Laboratory for Product Development and Lightweight Design
Prof. Dr. Markus Zimmermann

Supervisor Laboratory for Product Development and Lightweight Design
Akhil Sathuluri
Hans Zweitkorrektor (Second corrector)

Submitted by Chi Zhang
Karl Köglsperger Straße 9, 80939, München
Matriculation number: 03735807
chi97.zhang@mytum.de

Submitted on Garching, 15.11.2023

Declaration

I assure that I have written this work autonomously and with the aid of no other than the sources and additives indicated.

Garching, 15.11.2023

Chi Zhang

Project Definition (1/2)

Initial Situation

Add your Project Brief here. If you don't need it, comment out the creation of this Project Brief in the main document `Thesis.tex`.

Goals

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Project Definition (2/2)

Contents of this Thesis

- Lorem ipsum dolor sit amet
  - Lorem ipsum dolor sit amet
  - Lorem ipsum dolor sit amet
  - Lorem ipsum dolor sit amet
- Lorem ipsum dolor sit amet
  - Lorem ipsum dolor sit amet
  - Lorem ipsum dolor sit amet
  - Lorem ipsum dolor sit amet
- Lorem ipsum dolor sit amet
  - Lorem ipsum dolor sit amet
  - Lorem ipsum dolor sit amet
  - Lorem ipsum dolor sit amet
- Lorem ipsum dolor sit amet
  - Lorem ipsum dolor sit amet
  - Lorem ipsum dolor sit amet
  - Lorem ipsum dolor sit amet

Project Note

Master's Thesis	Nr. xxx
Supervisor	Akhil Sathuluri
Partners in industry/research	DFKI GmbH, Robotics Innovation Center
Time period	15.05.2023 - 15.11.2023

The dissertation project of Akhil Sathuluri set the context for the work presented. My supersivor Akhil Sathuluri mentored me during the compilation of the work and gave continuous input. We exchanged and coordinated approaches and results monthly.

An accurate elaboration, a comprehensible and complete documentation of all steps and applied methods, and a good collaboration with industrial partners are of particular importance.

Publication
I consent to the laboratory and its staff members using content from my thesis for publications, project reports, lectures, seminars, dissertations and postdoctoral lecture qualifications.

The work remains a property of the Laboratory for Product Development and Lightweight Design.

Garching, 15.11.2023

Chi Zhang

Akhil Sathuluri

Contents

1 Introduction

Nonlinear systems, as their name suggests, don't adhere to linear relationships between inputs and outputs. This lack of linearity means the system's response to input changes is intricate and often unpredictable. In the real world, nearly all systems exhibit some form of nonlinearity. This nonlinearity manifests in various phenomena. For instance, in systems with multiple inputs and multiple outputs, interdependencies between variables become complex, posing a coupling problem. Chaotic behavior, often referred to as the butterfly effect, is another common issue. Even a slight alteration in initial conditions can drastically alter the system's outcome. The classic butterfly effect example illustrates this sensitivity — a butterfly's wings in Brazil potentially triggering a tornado in Texas. When dealing with control tasks, especially in many real-world systems, accounting for this nonlinearity is essential.

Gaining proficient control over nonlinear systems has been a primary objective in the field of control theory for a substantial period. Throughout history, control engineers have developed a diverse range of approaches to handle these intricate systems. The advent of robotics in recent times has brought forth new methodologies specifically designed to tackle the challenges posed by nonlinearities.

1.1 Motivation

Robots, which are programmable mechanical entities, are purposefully designed for autonomous or semi-autonomous task execution, showcasing mobility, manipulation, and interaction with their surroundings.

In the realm of modern robotics, these machines exemplify intricate, highly nonlinear mechanical systems. Some well-known instances include quadruped robotics, autonomous vehicles, quadcopters, and humanoid robots.

[pictures here]

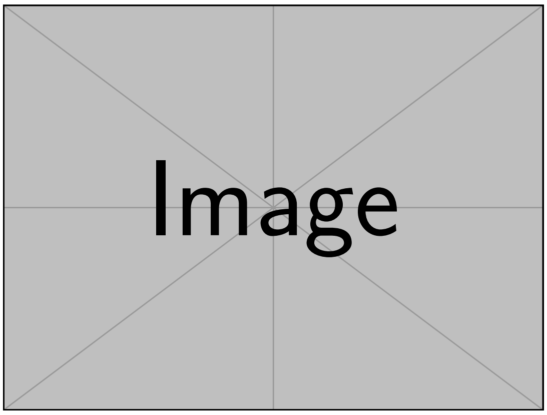

Figure 1.1: Caption for Image 1

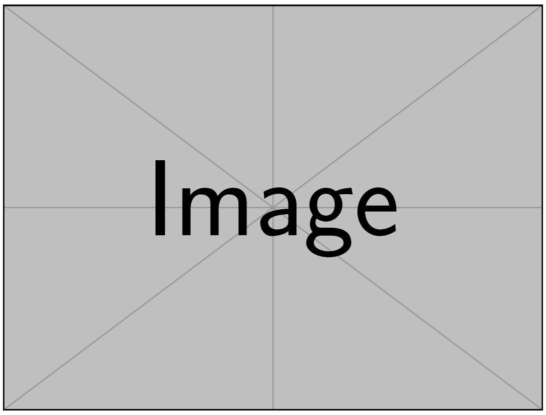

Figure 1.2: Caption for Image 2

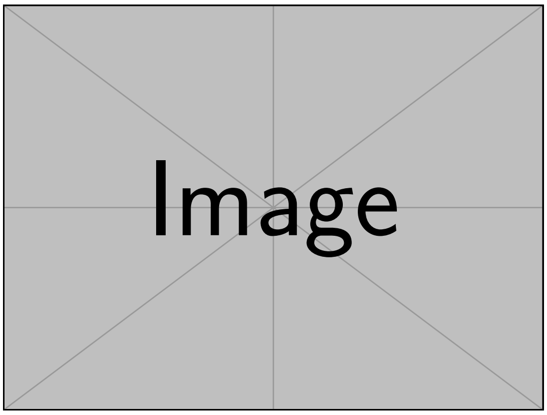

Figure 1.3: Caption for Image 3

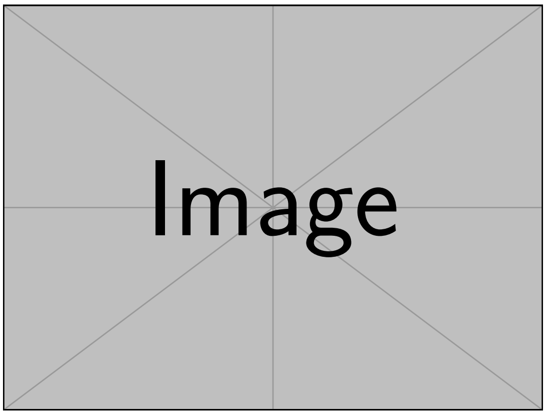

Figure 1.4: Caption for Image 4

1.1.1 Trajectory planning and tracking

To enhance their motion capabilities, reliable nonlinear control methods are essential. Traditionally, for planning complex motion in systems like those mentioned above, a two-step approach is followed, namely trajectory planning and trajectory tracking.

Trajectory planning involves computing a smooth and feasible path that a robot should follow to reach a specific target position or work point. A common method employed is trajectory optimization, aiming to minimize a cost function that factors in travel time, energy consumption, and smoothness of motion. Techniques such as gradient descent or genetic algorithms are often used for this optimization. Adhering to constraints such as maximum velocity and accelerations is crucial during this process.

Once the trajectory is successfully planned, trajectory tracking involves implementing control algorithms to guide the robot along the planned trajectory. Typically, a feedback control approach is utilized, continuously monitoring the robot's position and adjusting control inputs. However, in real-world systems, even with a well-planned trajectory, external disturbances, uncertainties, or system limitations may cause significant deviations. Hence, accurate state estimation and robust control are crucial in the realm of feedback control.

In the domain of industrial robot control, it's typical to separate the process into distinct planning and execution phases. This division arises from the fact that real-time responsiveness is not a stringent necessity in this context. When a task is defined, the planning phase kicks in, utilizing algorithms like linear interpolation and A* for trajectory generation. Following this, a steady and reliable control policy is implemented to ensure precise tracking of the generated trajectory during the execution phase.

Yet, in dynamic domains like automotive and flight control, the demand for real-time responsiveness takes center stage. Model Predictive Control (MPC) stands as a prime illustration of this critical requirement. MPC seamlessly integrates trajectory planning and execution into a unified framework. The process commences by projecting a sequence of control actions into the future as part of the planning stage. Subsequently, the calculated control inputs are meticulously fine-tuned to minimize the deviation between the actual system state and the planned trajectory. This strategic implementation effectively guides the system to closely track the intended trajectory.

MPC's brilliance lies in its ability to concurrently devise and optimize trajectories while swiftly adapting in real time to stay closely aligned with the planned trajectory, even when facing various disturbances and uncertainties. This amalgamation plays a pivotal role in achieving precise control and adaptability, especially in rapidly changing and intricate environments.

Though the traditional approach of trajectory generation and tracking has proven effective for many systems, it has the following drawbacks:

- **Limited Adaptability:** Trajectory planning typically relies on predefined paths or trajectories, limiting adaptability to unforeseen changes or dynamic environments. If the environment changes significantly, the planned trajectory may no longer be optimal or even feasible.

- **Difficulty in Complex Environments:** In highly complex and cluttered environments, planning a feasible trajectory that avoids obstacles while reaching the goal can be challenging. The complexity increases with the number of obstacles and the intricacy of the environment.
- **Difficulty with Nonlinear Systems:** Trajectory planning struggles with highly nonlinear systems where the dynamics are hard to model accurately. Linearizing the system for planning purposes may lead to suboptimal or infeasible trajectories.
- **Static Planning:** Traditional trajectory planning is often static, assuming a stationary environment. It does not readily adapt to changing circumstances or dynamic obstacles, which limits its applicability in real-world scenarios.
- **High Computational Demands:** Some trajectory planning algorithms can be computationally intensive, especially for high-dimensional or complex robotic systems. This computational demand becomes a drawback, particularly in real-time or time-critical applications.

1.1.2 Reinforcement learning based control

In contrast to trajectory-based control, reinforcement learning (RL)-based control extracts an optimal policy through interactions with the environment, offering several advantages:

- **Adaptability and Flexibility:** RL enables systems to adapt and learn optimal behavior in environments that are dynamic and chaning. The control policy can continuously evolve based on new experiences and acquired knowledge, ensuring adaptability to varying circumstances.
- **Less Model Information Required:** In contrast to traditional control methods that often necessitate a precise mathematical model of the system, RL can directly learn from interactions with the environment without relying on an explicit model. This characteristic is particularly valuable in scenarios where system dynamics are complex or unknown.
- **Effective Handling of Nonlinearities and Complex Systems:** RL proves highly effective in dealing with highly nonlinear systems and complex control tasks that might pose challenges for traditional control methods. The use of neural network-based function approximation allows for the capture of intricate relationships between states and actions.
- **Efficient Handling of High-Dimensional Input Spaces:** RL demonstrates an ability to efficiently manage high-dimensional and continuous input spaces, a crucial feature in many real-world applications such as robotics, finance, and game playing.

Reinforcement learning, by learning directly from the environment, offers a dynamic and adaptable approach to control, making it particularly suitable for complex and nonlinear systems.

1.2 Problem setup

Within the realm of nonlinear systems, a particularly challenging class is underactuated systems. These systems are characterized by having fewer control inputs than degrees of freedom. This makes them notably harder to control compared to fully actuated systems. Interestingly, a majority of robots and even living beings in nature fall into the category of underactuated systems. Consequently, studying the control of underactuated mechanical systems holds significant universal relevance.

The double pendulum, a simple setup comprising two links connected by two rotational joints, is a prime example. The joints involved are the shoulder joint, directly connected to the world frame, and the elbow joint, situated between the two links. The end effector is positioned at the tip of the second link. Active control is achieved by attaching motors to the shoulder and elbow joints. In the domain of underactuated control, if the shoulder joint is actuated, the setup is known as a pendubot. On the other hand, if the elbow joint is actuated, it's referred to as an acrobot.

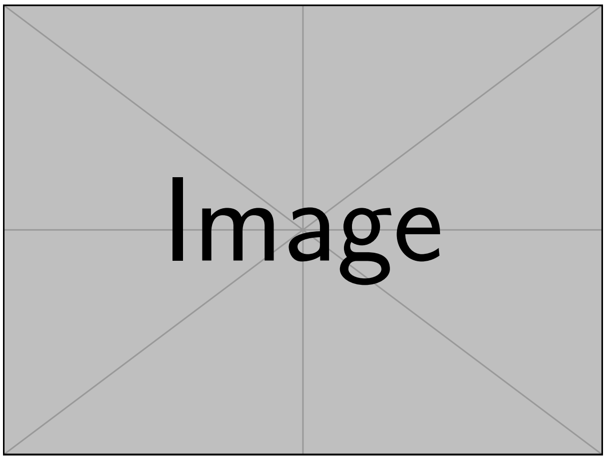

Figure 1.5: This is a sample image.

Despite its simple configuration, the system exhibits highly nonlinear and chaotic behavior. The double pendulum setup presents two classic tasks: swing-up and stabilization around the highest point. Research on swing-up and stabilization of the double pendulum can be traced back to the 1990s, and it continues to be a crucial testbed for validating the effectiveness of newly designed control algorithms.

Our project's motivation is to develop a reinforcement learning-based control method suitable for underactuated control of the double pendulum system, specifically addressing swing-up and stabilization tasks. To evaluate the efficacy of this control method, we conduct both simulations and real system experiments.

1.3 Contribution

In this paper, our main contribution is as follows: developing an effective control strategy to achieve two key objectives with the double pendulum. The first task involves swinging the double pendulum from its lowest point to its highest point. The second task is to maintain stability at the highest point.

To tackle the swing-up task, we utilized a well-known model-free reinforcement learning algorithm called soft actor-critic. This algorithm allowed us to train a policy capable of reaching the region of attraction (RoA) of a continuous-time linear quadratic regulator (LQR) controller. Once the system enters the RoA, we seamlessly transition to the LQR controller to maintain stability around the highest point.

1.4 Content

The paper is structured as follows:

- **Chapter 2: Introduction to State-of-the-Art**
  - This chapter provides an overview of current advancements in the field, summarizing essential theories, including those related to nonlinear and underactuated control.
- **Chapter 3: Methodology**
  - This chapter delves into the methodology, encompassing fundamental aspects of reinforcement learning, with a specific focus on the SAC algorithm. It explains the reward

function used for training, the training procedure, and covers the concept of the LQR controller and how the combined controller was composed.

- **Chapter 4: Simulation Results**
  - In this chapter, we present the results obtained from simulations, showcasing the performance and behavior of the designed control strategy.
- **Chapter 5: Hardware Results**
  - This chapter reports the outcomes of experiments conducted on the hardware, providing insights into how we addressed the sim2real transfer problem.
- **Chapter 6: Discussion and Future Work**
  - The final chapter engages in a discussion about the obtained results and explores potential future directions for research and development.

2 State of the art

This chapter is about the state of the art.

ad;falknv.xzfvhlsakjdgfnmdflk asdjgfmndfgb;lkjesf;mngfbl;kjfx.,mszedk.jfal;j

2.1 Theory

This section is about the theory, for example dynamics and underactuated control.

2.2 Related work

This section is about the related work about this project.

3 Methodology

This chapter is about the Methodology.

ad;falknv.xzfvhlsakjdgfnmdflk asdjgfmndfgb;lkjesf;mngfbl;kjfx.,mszedk.jfal;j

3.1 Soft actor critic

This section is about SAC.

3.2 Linear quadratic regulator

This section is about LQR

3.3 Combining SAC and LQR with region of attraction

This section is about how to use ROA to combine SAC and LQR

3.4 Reward shaping

This section is about the reward shaping problem of reinforcement learning training.

4 Experiment: training and simulation

This chapter is about the training and simulation.

ad;falknv.xzfvhlsakjdgfnmdflk asdjgfmndfgb;lkjesf;mngfbl;kjfx.,mszedk.jfal;j

4.1 Training setup

This section is about training setup and environemnt building.

4.2 Training process

This section is about learning curve and tuning parameters

4.3 Simulation results

This section is about simulation results in pendubot and acrobot.

pendubot:

acrobot:

5 Experiment: hardware system

This chapter is about the hardware experiment.

ad;falknv.xzfvhlsakjdgfnmdflk asdjgfmndfgb;lkjesf;mngfbl;kjfx.,mszedk.jfal;j

5.1 Hardware setup

This section is about hardware setup and its features.

5.2 system identification

This section is about the system identification problem when using hardware system.

5.3 sim2real problem

This section is about sim2real problem.

5.4 real hardware results

This section is about simulation results in pendubot and acrobot.

pendubot:

acrobot:

6 Discussion

This chapter is about the discussion of results.

ad;falknv.xzfvhlsakjdgfnmdflk asdjgfmndfgb;lkjesf;mngfbl;kjfx.,mszedk.jfal;j

6.1 introduction to leaderboard results

This section is about hardware setup and its features.

6.2 interpretation of simulation results

This section is about explaining the simulation results.

6.3 interpretation of hardware results

This section is about explaining the hardware results.

6.4 future work

This section is to talk about things to be done.

Appendix

A An appendix

You can structure appendices, just like your thesis, with the `\chapter`, `\section`, and `\subsection` commands. Referencing also works as usual.

If your thesis does not contain an appendix, comment out the creation of the appendix at the appropriate place in the `Thesis.tex` file.