CSC3180: Fundamentals of Artificial Intelligence, Spring 2023
Project Proposal, Group 10
March 25, 2023

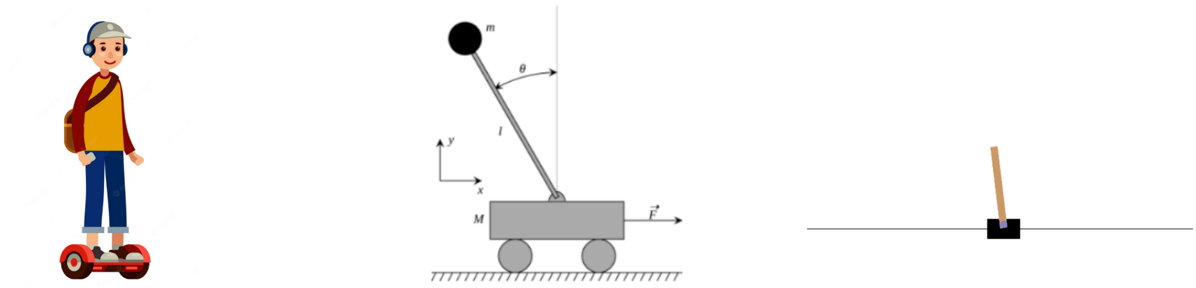
Adaptive Dynamic Stabilization of a Self-Balancing Scooter

Hongyi Yang, Muhan Lin, Xuanyang Xu, Zixing Jiang

---

1 Introduction

The self-balancing scooter (Figure 1a) is a common urban commuter tool. One of the most critical features in self-balancing scooter is dynamic stabilization, which allows the scooter to maintain its balance while in motion. Without this feature, it would be very difficult for riders to stay upright and control the scooter, and even cause safety accidents in severe cases. In order to achieve dynamic stabilization, the traditional automatic control methods, including Proportional-Integral-Derivative (PID) control and Model Predictive Control (MPC), model the self-balancing scooter as a cart-pole system (Figure 1b) and designs the controller by deriving its system dynamics. However, the dynamics-based control is sensitive to the rider's weight and center of mass (CoM). For riders of different weights and CoMs, the performance of stabilization varies. In this project, we aim to utilize the learning ability of artificial intelligence to design a dynamic stabilizer that adapts to different rider weights and CoMs.



(a) Man on a self-balancing scooter (b) Cart-pole system dynamics (c) OpenAI Cart Pole Environment

Figure 1: Illustrations of (a) the self-balancing scooter, (b) the cart-pole system dynamics, and (c) the OpenAI Cart Pole environment.

2 Problem Statement

In this project, we model the scooter as the cart-pole system (Figure 1b). The wheel of scooter is the cart the rider-scooter mass is the pole. Then the adaptive stabilization problem can be formulated as follows: control the inclination (θ) of a pole with varying mass and CoM within a certain range by controlling the movement (x) of the cart (leftward or rightward). We intend to apply reinforcement learning (RL) to solve this problem. Namely, given the observation of the environment $(\theta, \dot{\theta}, x, \dot{x})$, we want our agent (cart) to learn a policy (move leftward or move rightward) to stable the varying pole through trail and reward. We plan to implement our RL algorithm based on the OpenAI Gym environment for cart-pole (Figure 1c).

3 Work Distribution

- Zixing Jiang: RL algorithm design and implementation
- Xuanyang Xu: RL algorithm design and implementation
- Muhan Lin: OpenAI Gym environment modification, Algorithm implementation
- Hongyi Yang: OpenAI Gym environment modification, Algorithm implementation

Report and presentation will be done jointly by all members.

4 Project Schedule

- Week 9: modify the OpenAI Gym environment according to our needs
- Week 10: design and implementation RL algorithms for the adaptive stabilization task
- Week 11: design and implementation RL algorithms for the adaptive stabilization task
- Week 12: validate and optimize (if possible) our algorithm
- Week 13: record video demo and draft report
- Week 14: group presentation