

Robotic Unicycle

HARVEY HUGHES

SUPERVISOR: PROFESSOR CARL RASMUSSEN

May 26, 2020

TECHNICAL ABSTRACT

The application of machine learning in determining effective controllers opposed to using control theory could prove to highly useful when its not possible to determine system dynamics. This means new and complex tasks could be tackled with minimal knowledge about system behaviour. A robotic unicycle is considered in this project to provide a sufficiently complex problem to fully test the PILCO algorithm and its success outside simulations.

This project builds upon previous students work dating back to 2005. Throughout the projects history the unicycle being tested has evolved from a large and dangerous model, to a smaller model five years ago. In addition to model change, different control algorithms have been used and extensively built up.

The last two years of project work has seen significant improvements and simplifications to the software and electronics. This helped to pinpoint bottle-necks in the learning process but unfortunately the smaller unicycle showed great difficulty at balancing.

Speculation that the unicycle roll limit of 17° impeded learning was thoroughly investigated among other variables through simulations. These simulations showed that a

mechanical redesign is necessary for learning to occur. This design aimed to upgrade and simplify the physical robot to match the progress made in electronics and software. The new design has a roll limit of 47° , softens impacts and reliably attaches sensors.

Numerous problems were found in existing communications methods between phone, computer and Raspberry Pi. These problems introduced large variation in delay between readings, and operated too slowly to provide sufficient learning. For sufficient learning a frequency above 20Hz was shown to be required. To solve these issues and provide additional simplicity UDP protocols were implemented for all transmissions. Unifying all transmissions enables the possibility of removing a computers involvement should it be desired in future.

Upon starting rollouts further problems were located and fixed. The initial orientation was measured to remove offsets created in the dynamics learnt and therefore improve model accuracy. Trajectories end when the unicycle hits the ground, this detection proved to be unreliable and rollouts became invalid. Additional constraint checks involving future predictions were implemented in addition to manual confirmation.

Unfortunately few rollouts were conducted due to problems charging the battery during the COVID-19 lockdown. The rollouts performed showed little progress at learning, but this is attributed to low battery and insufficient policy optimisations.

During simulations and real rollouts policy optimisation often plateaued, or only learnt to wander instead of stay at the origin. The implementation of a quadratic controller and policy exploration techniques is believed to alleviate these issues.

Future work should seek to investigate these issues, taking advantage of the updated and simplified mechanical, electrical, and software systems to identify key requirements.