6.4212 Project Proposal

Timothy Kostolansky

Motivation

Multi-model motion planning (MMMP) is the problem of finding a feasible trajectory for a robot to follow in order to complete a desired objective. This is done while interacting with its world, hence the multi-modal name. This proejct will focus on two methods to approach MMMP problems: task-and-motion planning and reinforcement learning.

Task-and-motion planning (TAMP) is a method for agents to plan and act within an environment. TAMP uses a more structured approach to solve MMMP problems. The structure comes in the form of discrete representations of strategies that can be implemented by agents. These representations can allow for simpler and more interpretable solutions to MMMP problems relative to other, more machine learning-centric approaches. This project will aim to integrate TAMP into solving of MMMP problems and benchmark its success rates in solving these problems.

Reinforcement learning (RL) is an alternative method for enabling autonomous agents to learn how to act within their environment using exploration. Agents use information gaiend through experience with their environments to create a policy with which to act on. The policy is iterated on in order to maximize a reward that is provided to the agent through built-in or inferred cues. RL has been used to solve many MMMP problems, and this project will also aim to use it to provide benchmarks for the same MMMP problems TAMP is used to solve.

The RL framework works in many applications today, although it has certain bottlenecks. One primary trouble with RL is the uncertainty in an agent's environment, which makes it difficult to determine how to explore and choose a suitable policy to maximize its reward. Due to a potential sparsity of rewards and general unstructured nature of the exploration in RL, some problems are very difficult for RL agents to solve, i.e., for the agents to determine a suitable policy with which to interact with the environment.

RL and TAMP are both useful methods to solve MMMP problems, and this project aims to implement the two methods with available computation and simulation software, as well as to compare the performance of the methods over a number of tasks. Due to differences in their structure, RL and TAMP may respectively find different problem regimes where they perform better. RL may have a hard time determining policies which require long-horizon decision-making, which is where we expect TAMP to perform better. As such, we will provide a number of tasks that span a variety of problem difficulties in order to test the capabilities of RL and TAMP.

This project is related to future research that I plan to conduct. Specifically, this project is a precursors to research that will hopefully aid in creating interpretability for AI systems.

Project

As mentioned in the Motivation section, this project entails implementing TAMP and RL methods on available computation and simulation software and then benchmarking them applied to a number of MMMP problems. The reasoning for selecting multiple problems is described above as well. This benchmark will provide the time it takes for each method to complete tasks, the computation resources used by each method, and the problems that each method runs into. This project will be completed within simulation with the Kuka iiwa robotic arm, with work to be transferred to the real world pending successful completion of the simulated testing.

Each of TAMP and RL will be applied to the following problems:

1. Simple pick-and-place: moving an object from one spot to another
2. Object sorting: taking cluttered objects and partitioning them in some desired manner
3. Object stacking: moving objects to be stacked in a desired manner

These problems provide a variety of difficulties and nuances that require robust trajectory optimizations to solve. We believe that this set of tasks will allow the two methods to showcase their strengths and weaknesses.

Implementation of this project will likely entail use of Drake, its associated simulation software, and RL and TAMP integrations with Drake. One specific codebase we plan to use is OpenTAMP, developed by members of the Algorithmic Alignment Group in MIT CSAIL. With these frameworks, integration and simulation should work as demonstrated in the Deepnote notebooks accompnaying the class textbook.

Prior Work

There is a wealth of work relating to RL and RL benchmarking, for example this paper on benchmarking a number of deep RL methods. It doesn't seem that there exists benchmarks comparing RL and TAMP though.

Timeline

- November 1: Learn how to set up RL and TAMP problems using Drake/other resources (e.g., OpenTAMP)
- November 8: Set up simulation, run toy problems with RL and TAMP
- November 15: Run simulation and benchmarking of TAMP pick-and-place
- November 22: Run simulation and benchmarking of RL pick-and-place
- November 29: Run simulation and benchmarking of object sorting and object stacking