Using Reinforcement Learning to Train a 7-DOF Robot to Catch Falling Objects

Seth Robles
sirobles
10/17/25
6.4212 Robotic Manipulation
Prof. Lorenzo-Perez

Executive Summary:

This project investigates how reinforcement learning (RL) can be used to control a 7-DoF KUKA iiwa 7 arm to catch falling objects. The simulation will be built in Drake and make use of Drake's implementation of OpenAI Gym Environments. The deliverable for this project will be a trained RL policy that enables the robot to intercept and catch a falling object within a defined workspace. Success will be measured by the catch success rate (arbitrarily set at 75%) across varied drop positions and velocities. The final submission will include a demonstration video, training logs, and analysis of the robot's performance across conditions.

Relevant Class Terms: Motion planning with model constraints; Geometric perception, reinforcement learning; Contact modeling

1 Motivation and Previous Work:

I am interested in reinforcement learning because it is becoming a component of modern robotic control, particularly for systems that must adapt quickly or with uncertainty. I believe that catching is a good testbed—it looks simple, but it brings together motion prediction, planning, and precise control with contact. The task provides a safe but challenging way to practice RL within Drake and demonstrate principles from class.

There has been much past work similar to this project. *Revisiting Ball Catching* (Dong et al., ICRA 2020) demonstrates discrete replanning for mobile-manipulator intercepts, while *Rollin' Justin* (Leidner et al., Humanoids 2016) shows real-time perception and control for airborne catches. *TossingBot* (Zeng et al., RSS 2019) introduces residual-physics learning that transfers effectively to dynamic tasks. Finally, both underactuated robotics and this course reference the *Drake Gym* interface to create reproducible RL environments.

2 Project Plan:

2.1 Project Setup:

The project will be broken into three stages: situation perception, trajectory prediction, and motion planning.

Situation perception will include the falling object's centroid and motion trajectory in simulation. Initially, this will use ground-truth information. Later, I plan to add simulated cameras that are closer to real life scenarios. **Trajectory prediction** will encompass analyzing projectile motion to provide a time-based intercept estimate. Assuming a ground-truth, this is relatively simple. With increased complexity of perception, however, this will also grow more difficult. **Motion planning** is the main focus of this project. Given a trajectory, this will plan the iiwa's movement to catch an object while maintaining a stable orientation. The catching mechanism could use the default 2 finger gripper used in the class or include some attachment to simplify grasping dynamics.

Since the emphasis is on the RL component, limiting perception complexity will allow me to familiarize myself with training a policy. I will do my best to avoid pursuing this extensively, ensuring that the scope of the project always lends itself somewhat to reality. In a really ideal scenario, if my simulation work goes well, I would like to try implementing this on an actual iiwa. I have discussed with an MIT lab with an iiwa available and plan on continuing communication to learn about the process of real2sim either as part of my final demo or as a follow-up after this class.

2.2 Timeline

Week Number	**Date Range**	**Due Dates / Milestones**	Notes / Planned Work
Week 8	**Oct 17 – Oct 23**	Project Pre-Proposal Due (10/17)	Finalize concept, have an initial setup simulation with the robot and a collection of objects I will use to drop
Week 9	**Oct 24 – Oct 30**	Project Proposal Final Revision Due (10/31)	Write and submit full proposal; start building environment (arm + object + physics).
Week 10	**Oct 31 – Nov 6**		Implement preliminary control baseline (PD or scripted catch); test environment stability.
Week 11	**Nov 7 – Nov 13**	Reinforcement Learning Lectures (11/12–11/13)	Begin RL integration (DrakeGymEnv + PPO/SAC); debug training loop.
Week 12	**Nov 14 – Nov 20**	Problem Set 8: Deep Perception and RL (11/20)	Train first working policy, collect performance metrics; start domain randomization.
Week 13	**Nov 21 – Nov 27**	Thanksgiving Break	Run longer training sessions; visualize results and performance under noise.
Week 14	**Nov 28 – Dec 4**		Work on the report and video (cool demo). **If things go really well, sim2real would be incredibly cool, but I understand this is very difficult and likely not feasible time-wise
Week 15	**Dec 5 – Dec 12**	**Final Project Report and Video Due (12/10)**	Record demonstrations, finalize report and video, polish figures and analysis.

References

K. Dong et al., “*Revisiting Ball Catching with a Mobile Manipulator: A Discrete Trajectory Planning Approach*,” ICRA, 2020.

D. Leidner et al., “*Playing Catch and Juggling with a Humanoid Robot*,” IEEE-RAS Humanoids, 2016.

A. Zeng et al., “*TossingBot: Learning to Throw Arbitrary Objects with Residual Physics*,” RSS, 2019.