Alex Brinkman

Team D: Project HARP (Human Assistive Robotic Picker)

Teammates: Abhishek Bhatia, Lekha Mohan, Feroze Naina, Rick Shanor

ILR #6

Submitted 1/28/2016

Individual Progress

Our project has evolved greatly since last semester. We have decided to migrate from the PR2 to the UR5 provided by Universal Robot, confident the benefits of the UR5 platform far outweigh the program risk the migration introduces. Over break, I drove the communication between our sponsor and our contact from Universal Robot. Additionally, I migrated our codebase from using ROS Groovy and the PR2 to ROS Indigo and UR5. More recently, I have been working on ROS and Moveit! control of the UR5, improving the system architecture to be more modular and configurable, and helping integrate the vision pipeline and localization nodes into our architecture.

On December 23rd, I received word from Tom Moolayil that they could accommodate our request for a loaner UR5. Tom wanted endorsement from Maxim and a proposal for how we would use the robot and offer exposure for Universal Robots. I handled the communication between our team, Maxim, Venkat, Andrew, and Tom to ensure that everyone understood our decision and agreed with our reasoning. We were able to send the proposal and got Maxim's endorsement. Since then, Tom has promised to bring the UR5 in person to help with setup the week of February 1st.

We established team goals over break and I set my sights on migrating the codebase to the UR5. Many Groovy-specific packages and messages had to be worked around to get our code to work and UR5 configuration changes needed to be made to include our custom hardware.

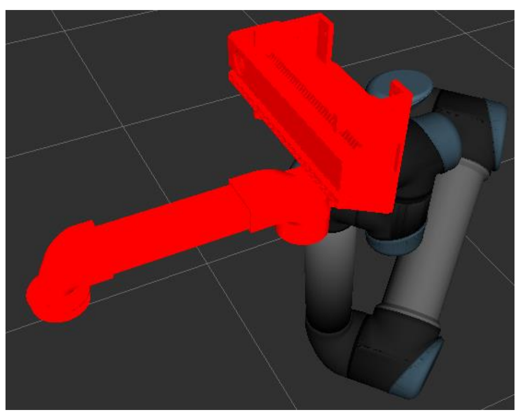

Figure 1: Custom Hardware Simulated in Moveit!

Currently, we are in the process of migrating the UR5 from the default OMPL planner to the SBPL planner in hopes of reducing the computational complexity of path planning and creating deterministic plans. The SBPL planner will also display the intermediate steps of the motion plan, which this feature is currently lacking on the UR5 fake controller.

With the addition of the vision pipeline and localization features, it became increasingly clear we needed an improved way of managing the TF offsets between the different scripts and tests we need to support and manage. I decided to encode the transforms between certain components in a yaml file that stores these offsets and loads them for static and dynamic TF publishers. For example, the UR5 to Kinect transform will eventually need to be dynamically calibrated, but for now the offset is manually determined and loaded as the default transform. The calibration routine can be added in later without conflict with the current system.

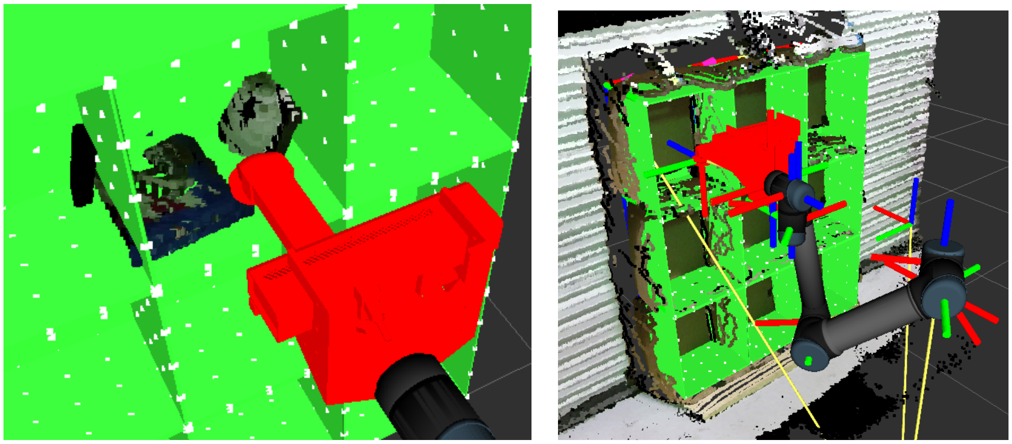

Figure 2: Vision Pipeline Working in our Framework

Challenges

The largest challenges we face are immediate tasks to prepare for the Amazon submission. To prepare we would like to create a high quality video to demonstrate our work and explain our system. The first thing we need is to get the UR5. It has been promised to us soon but we may have to submit without actual hardware. The challenge will be to get up and running on the UR5 as soon as possible and hopefully that is ahead of the February 5th submission. We expect many unforeseen problems as we integrate on hardware for the first time and are unsure if we can work through them fast enough.

The next challenge is to get the SBPL plugin working on the UR5. Andrew Dornbush has been helping migrate the SBPL planner to the UR5 and has provided us with the code and packages required. We have been able to run the demos provided with the packages but have not yet integrated it within our system yet. The experimental nature of the planner means it will not plan for the end effector so we will have to manually add the end effector offsets in the coordinate frame of the robot and fix our framework accordingly. Also, the SBPL plugin requires manually defined collision objects and will ignore the collision model provided in the URDF file. We will need to demonstrate stable collision avoidance in order to run the UR5 which will take some time.

Finally, a more general problem will be to improve the perception system to account for the new rules. Our algorithm works well on the old rules but the new rules allow for partial occlusions and include more items on each shelf, significantly complicating the perception problem. We have three ways to improve this 2 ROS packages offer a vision pipeline that Rick and Abhishek are currently analyzing and and SBPL based pipeline that Venkat is working on. We will bench mark these results with the results of the system developed last semester and determine a path forward.

Teamwork

I worked a lot with Rick to integrate the perception pipeline and localization code. Lekha and I developed the JSON input handling functionality. Feroze and I worked on adding collision objects and improving the performance of the Moveit! simulation. Finally, Abhishek and I worked to debug some coordinate frames management issues as the camera, robot, and shelf systems were integrated.

Plans

Specific goals have not been defined yet for the next review but we are working towards completing the following tasks:

Create a database of predefined grasps for each item

Improve perception using ORK, Simtrack ROS packages

Build and install UR5 upon arrival

Integrate SBPL planner on the UR5

Make a video for the Amazon submission