Carnegie Mellon University

16-682

MRSD Project II

Task 07 Progress Review 10
Team C - COBORG

Jonathan Lord-Fonda
Teammates: Husam Wadi, Feng Xiang, Yuqing Qin, Gerry D'ascoli

October 28, 2021

Table of Contents

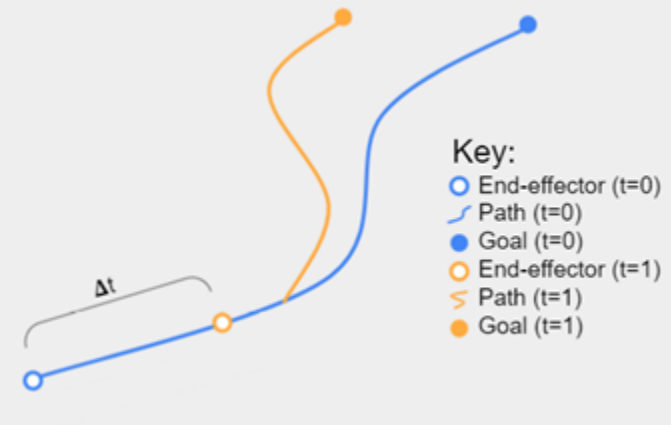
1. Individual Progress

My primary tasks for this progress review were centered on debugging the smart manipulation software node and integrating the actuated manipulation software. During a discussion with the team we realized that adding an extra camera caused a series of edge cases in which hand images were split across the two frames. The vision side of the team had assumed that the actuated manipulation side would handle this. The actuated manipulation side of the team (including me) had assumed that the vision side would handle this. After thorough discussion we worked out that the vision side should not only handle the edge cases, but also send updated local goals to the actuated manipulation subsystem, which had previously taken global goals and updated them internally. The change simplified the project and the new structure made it easily expandable, but it did require revising the existing actuated manipulation pipeline. Thankfully, most of the new logic required for the vision side could be recycled from the now unused actuated manipulation code. Since I had written this code I helped explain it to Jason and Yuqing so that they could implement it easily.

On the actuated manipulation side of things I spent a lot of time debugging the path stitching feature whose purpose was to allow the end effector to adapt to the user's movement as it stretched out towards the goal. Unfortunately, as I neared the end, shortly before Progress Review #10, I hit a snag where the only solution seemed to be to circumvent MoveIt's execution of our trajectories. Since this activity comes with both significant time investment and risk we decided to simply exclude this feature until the system was fully integrated and functional. Excluding path stitching in the final product is easily justifiable because the user is not expected to be able to move much while they themselves support the panel and there is nothing that they need to do except wait for the arm (i.e. they don't need to turn and grab tools, etc.). To that end I fixed up my side of the actuated manipulation to run with Yuqing's

revised node. After that I had some spare time waiting for Gerry and Jason's side, so I wrote an interpolator node that could be used to circumvent MoveIt's execution process. I hope to be able to use it in the final product but it's still untested so it all depends on how well the rest of integration comes together and whether we'll have time.

Figure 1 - Path Stitching



This figure shows the basic idea behind the path stitching feature. On every loop of the actuated manipulation system it projects into the future where the robot is expected to be and plans a path from that spot to the updated goal. It then stitches that new path onto the coincident point with the old path, chops off the beginning of the trajectory, up until where the robot currently is, and then sends the updated plan to MoveIt, which should integrate seamlessly with the current plan being executed. Unfortunately, MoveIt's async execute function simply fails if you send it a second async execute while it is executing. However, since we already have access to all of the trajectory information we need (positions, times, start state), we can just write a separate interpolator node that spits out the correct location to the motors at a set rate. All of the pieces are there, but since this code isn't necessary it will have to wait until the required parts are in place and functional.

Beyond these primary tasks I also worked with Gerry and Jason as they tested and finalized their code, offering insight and suggestions on how to fix their resolved rate problems. In addition Husam and I created a spreadsheet to track all of the commands, states, and statuses of and between the various subsystems. As we near the end of integration and begin running full use cases it is important that we clarify exactly what the communication between these systems will be and having a centralized location will prevent mix-ups in what a given integer represents.

2. Challenges

Debugging the actuated manipulation and path stitching took a really long time. Path stitching also ended in a temporary dead end. These are problems that can be fixed, given time, but time is in short supply at this point.

Thankfully though path stitching is not a required component of the system. Integrating the actuated manipulation code is proving to be challenging as we've mostly worked along parallel paths, but it shouldn't be too difficult. The true difficulty in this department is that we're a little behind schedule on all fronts, so hardware delays have pushed into software delays, which, in turn, have prevented us from trying out the full system early enough to change large things if that's required. Through a few small tests though I've come to believe that what we have will be sufficient and that we'll have enough time to tune it to functionality as long as we continue to push strong as a team.

3. Teamwork

Jason did tons of work during this previous cycle. He spent lots of time helping me debug and test the actuated manipulation code from my side of things. Additionally he worked with Gerry to build and test the resolved rate half of the actuated manipulation code. Even beyond that he helped Yuqing build and test the goal getter node.

Gerry's work during this previous cycle included finalizing and testing the goal stabilization code with Jason, as well as integrating it into Yuqing's new goal getter code. He also worked with Husam on the hardware upgrade, creating and installing custom-fit cables to improve the project's aesthetic and reduce the risk of pulled cords. Additionally he wore the Coborg for integrated testing.

Yuqing's work during this previous cycle was primarily focused on the goal getter node. She worked through the edge cases and challenges presented by multiple cameras, each with their own frames, and worked with Jason to finalize this. Additionally she worked closely with the rest of the team to guarantee that our nodes would communicate correctly.

Husam's work during this previous cycle primarily focused on preparing and assembling the Coborg's hardware components. He 3-d printed (and reprinted), spray painted, and assembled many of the parts. Additionally we worked with Gerry on updating the electrical components and ensured that we would have parts in time for FVD, despite the schedule setbacks we've had.

4. Plans

Before the next progress review I will finalize the integration of the actuated manipulation system, as well as tune it to function appropriately. I also intend to finish the path stitching feature and integrate that into the system as well.

Finally I will be finalizing the testing structure and running the entire use case with the system. As a final goal I intend to update the communications and decision-making of the main state machine with the other subsystems.