Carnegie Mellon University

16-682

MRSD Project II

Individual Lab Report 10
Team C: COBORG

Author: Gerald D'Ascoli

Remaining Team C Members:
Jonathan Lord-Fonda | Yuqing Qin | Husam Wadi | Feng Xiang

Sponsor:
Biorobotics Lab

November 11, 2021

Table of Contents

1 Individual Progress

For this progress review, I worked with the whole team on software integration, testing, and tuning. I worked to improve resolved rate by trying to implement orientation into resolved rate stabilization. I updated the “voice_recog” node with new needed functionality. I built and added new network cables to the Coborg. I tuned parameters in the obstacle avoidance functionality to optimize use of processing resources.

I wanted to add orientation into the resolved rate goal position because with only cartesian based goals, the end effector sometimes moves to weird angles with the supported part causing the passive end effector pad to not be flat against the part. Unfortunately, adding the roll/pitch/yaw to the end-effector and goal states causes the arm to destabilize at any goal, not just the goals outside of the task space or at singularities like with pure cartesian position control. This is most likely an issue with trying to reach a 6DOF goal with a 4DOF arm, but that will be discussed further in the challenges section. We may need to explore alternative methods to stabilize resolved rate like singularity rejection and enforcement of task space limitations.

The “voice_recog” node need to be updated with a state feedback publisher to inform the “machine_state_machine” and with a speaker subscriber callback function to allow other apps to easily interface with system audio output. I added the voice recognition node state feedback publisher to publish initializing, idling, processing, and completed recognition states. I wrote the speaker subscriber callback to read in a string on the “/speaker” topic where the string is the name of an mp3 file in the /voice_recog/src/Sounds, then process and play the mp3 audio segment to the system speaker. This allows the main state machine and actuated manipulation nodes to easily give audio feedback to the user without requiring their own audio interface implementation. While working on the “voice_recog” node, I also tuned the keyword trigger times a bit and added another surprise voice command for FVD.

To assist in the hardware upgrade, I took charge of creating and wiring the Coborg arm with power cables and network cables. The cables had to have tight tolerances to prevent extra cable length from splaying out away from the arm and getting recognized as an obstacle by the obstacle avoidance functionality. For this PR, after several failed attempts at salvaging custom cables from other ethernet cables, we purchased a roll of CAT5e cable, connectors, and crimping tools to build the cables. With Jonathan’s help in making some of the cables, I measured, crimped, and wired up the Coborg motors with the networking cables and connected them to the Coborg internal switch. These custom cables are shown connected to motor 1 (the first actuator) in Figure 1.

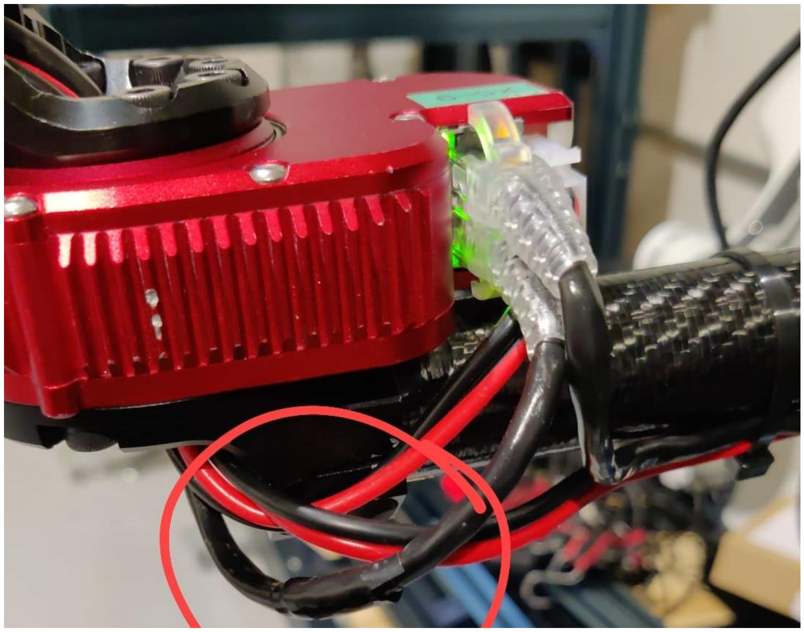

Figure 1. Ethernet cables attached to Motor 1

One significant problem we are running into with all systems integrated is limited processing resources. To help reduce the processing load, I tuned the obstacle avoidance parameters to reduce processing resource usage and optimize planning speed. One parameter I changed was the range of obstacle detection. Reducing the range reduces the chances that the part is detected as an obstacle which can cause the move-it planner to consider the path to the goal impossible which causes a use case failure.

A major time commitment for this progress review has been integrating, testing, and debugging in a cycle. I helped integrate my subsystems but mostly helped with the testing by wearing the Coborg and running the use case over and over again. We tested the use cases for both cases: mounting a part in front of the user and mounting a part over the user's head. Figure 2 shows me wearing the Coborg running the overhead use case.

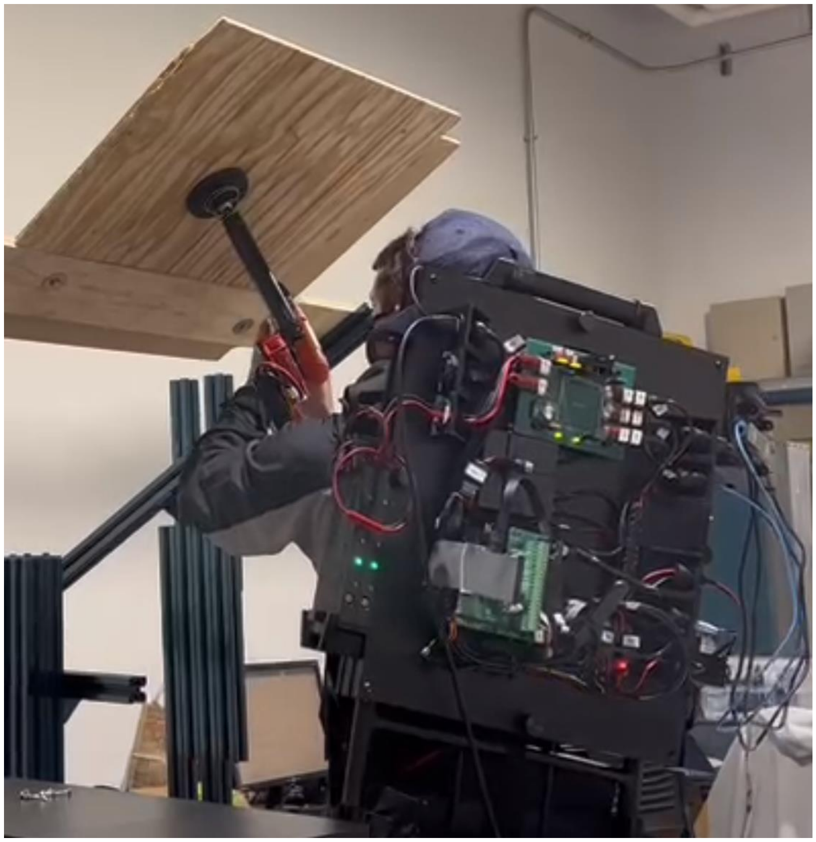

Figure 2. Wearing Coborg for overhead object support

2 Challenges

Integrating resolved rate into the use case has brought about all kinds of challenges. When I attempted to expand the current position and goal position considering in the closed loop solution with orientation to make it a 6 DOF position, it destabilizes resolved rate at all points, not just at singularities like when it is only closing on the cartesian error. I believe this is because the Coborg arm is 4DOF and it’s trying to actualize a 6DOF position. If this is the case, then adding orientation to resolved rate will be inapplicable to the Coborg system.

Another significant challenge to our use case is the excessive force effort exerted by the Coborg. The Coborg pushes with a bit too much force when holding the part to the surface, causing the user to have to brace a little bit and fight against the arm to reach the part with a drill. To resolve this, we are going to have to tune and test the effort PID controller on multiple motors on the Coborg arm.

A critical error with the current state of the Coborg is e-stop causes the Nvidia Jetson Xavier to power cycle. The e-stop properly powers off the arm when engaged, but when disengaged the inrush current to the motors draws excessive power away from the variable DC-DC regulator powering the computer, causing the voltage to drop from ~19VDC to ~10VDC. This causes the computer to shut off until the supply reaches ~19VDC again. Currently, we have some capacitors in parallel that increase the RC decay constant limiting the power drop to 10VDC instead of the 3VDC that would occur without any parallel capacitance. To resolve this power cycle issue, we are installing larger capacitors with 20 times more capacitance to dramatically increase the RC time constant and hopefully limit the power drop to within the Xiaver's power range.

3 Teamwork

Feng dedicated a significant amount of time working with the whole team to integrate goal getter, smart manipulation, and resolved rate motion compensation into a full pipeline and use case; calibrating and fine-tuning the camera mount and robot arm with Yuqing; and debugging through edge cases in actuated manipulation during use case testing. He's facing challenges with resolved rate control moving the robot arm to positions harmful to the user; the robot arm actuation being too fast causing it to collide with objects not in camera view; and the robot arm "bricking" in several edge cases causing users to have to hard restart the pipeline again. Before FVD, Feng plans to implement edge case behavior to ensure the robot arm continues to perform nominally; to calibrate the URDF and implement safety checking to ensure that the robot arm does not collide with the user; to implement more intuitive intermediate positions for the robot arm; and to tune the robot arm effort controller gains and limits to avoid user harm and prevent requiring excessive force from the user to operate use case.

Jonathan worked with the whole team to integrate software, run the use case, debug problems, calibrate, and tune the system; worked in parallel on path stitching feature; built custom ethernet cables with me; added system feedback feature with Husam; and adjusted testing structure to allow overhead use case with Yuqing. He is facing challenges with resolved rate not having forward thinking for path planning or obstacle avoidance; the t265 drifting when the Coborg requires centimeter-scale; and the arm being too weak and too strong, too long and too short (just in general not seeming to fit well in this use case). Before FVD, he plans to improve the Coborg consistency by tuning for the task space; get the system to automatically mitigate errors; and add adjustments to resolved rate to improve its performance.

Yuqing has worked on integration with the whole team; switched the vision model between YOLO v3, v3 Tiny, and v4 and experimented with different moving average window sizes to optimize hand recognition; added a distance threshold to the perception to resolve corner case issues; and calibrated the cameras with Feng. She is facing challenges with YOLO being less accurate in different lighting conditions and with processing resource limitations with all subsystems integrated together. For FVD, she plans to tune the moving average window size, timeout threshold, and other parameters in the vision system; debug the error that occurred during PR11 causing the use case failure; and continue with full integration testing.

Husam almost finalized the Coborg V2 hardware by redoing the Coborg frame for better user fit; creating a completely new wiring framework on the Coborg frame; and helping me with wiring up the robot arm. He also modified the "main_state_machine" node to engage and disengage the voice-activated e-stop. He faced challenges dealing with "endless weeks" working on the hardware to minimize blocking the software team from progress. He is facing challenges making the backpack enclosure mold while dealing with the vacuum mold machine being non-operational. He is also facing challenges with the Xavier shutting off due to the aforementioned inrush current power drop. For FVD, he plans to finish the hardware upgrade by making the external cover, adding fans for internal cooling, and changing the gripper pad on the end effector.

4 Plans

In preparation for FVD, I plan to work with my team to continue with integration testing and debugging of the overall use case. We need to test, test more, test some more, debug, test, debug some more, debug, test, polish, test, debug, test, polish, then test the use case before FVD to improve performance, reliability, and safety. Specifically, I’m going to focus on improving reliability of resolved rate through imposing task space limits, singularity rejection, and inclusion of orientation DOFs for the goal position. I’m also going to pray to any powers that will listen that our FVD is successful.