Individual Lab Report - Progress Review 9

Autonomous Reaming for Total Hip Replacement

HIPSTER | ARTHuR

Parker Hill

Team C:

Parker Hill | Kaushik Balasundar | Anthony Kyu
Sundaram Seivur | Gunjan Sethi

October 13th, 2022

Carnegie Mellon University
The Robotics Institute

Contents

1 Individual Progress

1.1 End-Effector Updates

Previously with the end-effector we had a functioning system which could be actuated via a ballscrew and a reaming motor. This system utilized a design that integrated load cells into the end-effector to effectively measure the axial force applied to the acetabulum. For this update, we decided to begin moving away from using the load cells for our force validation and instead just measure the current applied by the ballscrew motor, as well as begin adding some necessary features to the end-effector such as fiducial markers and limit switches.

Changing away from utilizing the load cells allowed us to simplify our reaming motor adapter significantly, as essentially we could just utilize an L-bracket to adapt to the carriage. We decided to add angled connectors to this L-bracket to prevent bending in the component during reaming operations when there may be a lot of axial reaction forces. For the marker holder and limit switches we decided to add a side cover to the end-effector to adapt both. This cover would allow limit switches to be attached via M2 screws on one side and allow for fiducial markers to be compressed onto the other side by a secondary adapter piece. A further benefit of not of this side cover was that it improved the rigidity of our system in general, and as such we will likely utilize another side cover for our finalized design. The final design with these improvements can be seen in figure 1.

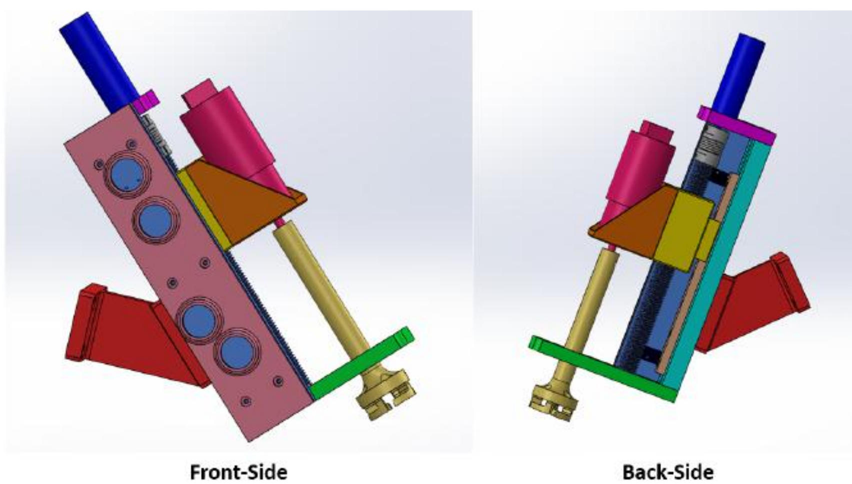

Figure 1: Updated CAD with Marker Holder and Limit Switches

With these parts updated in Solidworks we could then move forward with 3D-printing our new components. Most components were printed using our Ender 3 Pro in PLA and hand tapped using an allen key set, however importantly the reamer handle was printed in Tough PLA on the MRSD printers in order for better threads to form while hand tapping the set screw. The assembly process was relatively simple, and the finalized 3D-printed actuator can be seen in figure 2.

Figure 2: Updated Functional 3D-Print

1.2 Electrical System Integration

With the end-effector ready to be actuated, it came time to finally move forward with integrating the electrical system with the mechanical system. Previously we had used a Cytron MD10C, a custom PCB, an ATI force-torque sensor, and a Arduino Nano to actuate our end-effector, but this entire system had to be overhauled. We decided to use two Cytron MD10C motor controllers, an Arduino Mega, ATO load cells, limit switches, and current sensors for our electrical system. Currently only the Arduino Mega, Cytron MD10Cs, and limit switches and integrated.

The first step in integration was to elongate the motor wires for both the reaming motor and the ballscrew motor. As there were no good connectors that we could find in inventory, we decided to do so by soldering long AWG wires to the existing motors wires, and soldering jumper wires to the ends of those AWG wires for better integration into our circuits. While time-consuming, this was relatively easy and allowed us to extend our motor wires and interact with the motors easily. Next we had to solder wires to the limit switches and similarly attach long wires to them so that we can receive the signals from the limit switches at an Arduino under the table. With the wires soldered, we then wrapped all the wires in a wire sleeve and routed them along the arm, attaching the sleeve to the arm with zip ties, routing the wires to be exposed at the bottom of our Vention table.

With the wires exposed, Kaushik and I then brainstormed how to wire up all the components to the Arduino Mega and made a pin diagram for all the wires. This made it relatively simple to connect all the wires to where they needed to go, especially with the aid of a breadboard so that we could provide 5V and common ground to all components. The resulting wiring can be seen in figure 3.

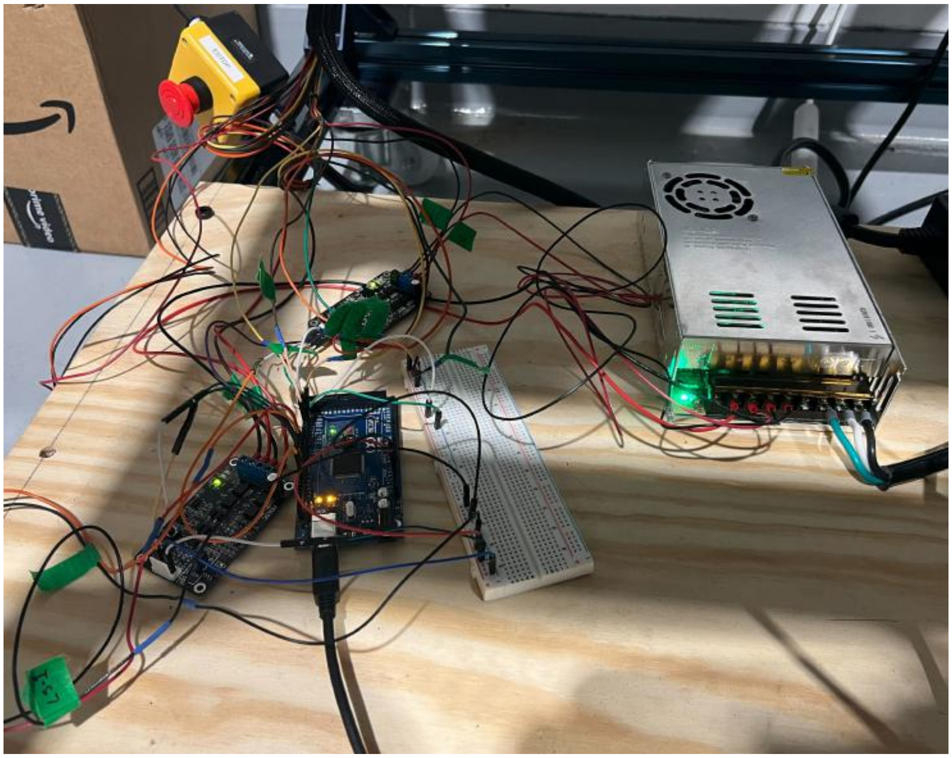

Figure 3: Current Wiring Under the Robot

With the wires exposed, Kaushik and I then began to write the Arduino code necessary to adapt to both motors and both limit switches. As I had previously written code which could allow the motors to be controlled in ROS via the Arduino, this was relatively simple, as we had to adapt the code to utilize both motor's encoders as interrupts and integrate the limit switches as interrupts which turn off both motors when pressed. With the code written, we are now capable of setting the velocities of both motors via ROS topics, and the ballscrew motor actuates until any limit switch is hit. With this verified, we are now capable of beginning to write more advanced control code for our end-effector.

1.3 User Interface

The final aspect of the project I worked on was the User Interface. I met with Gunjan to discuss her progress on the front-end of the user interface and discussed with her how to integrate with the watchdog and take information from the system and display it. We decided that we should receive booleans from the watchdog to monitor the health of different aspects of the system, and from those booleans we could make changes on the user interface as necessary. I then met with Sundaram to discuss how to expose these booleans on ROS topics, and we decided to write three custom messages which would be published on three separate topics for monitoring the health as reported via the watchdog. Based on this understanding I then created a simple python script which should subscribe to those topics and print out the health of all aspects of the system as they are reported. We have not tested this yet however as the user interface is still being developed separately from the rest of the ROS system and needs to be better integrated for testing to take place.

2 Challenges

2.1 End-Effector Updates

There were some challenges with the end-effector design as usual that led to reprints. The biggest problem we ran into however is that the way we are securing the reaming handle to the reaming motor is not stable enough for use in the system. Currently we are using a set screw, which is tapped through a hole in the 3D-printed reaming handle, to grab onto the reaming motor's shaft. This previously worked with our spring design, however it is not working with our current reaming handles as either the threads slip during the motor actuation (leading to the motor shaft spinning inside of the reaming handle) or the threads never actually allow for the screw to press against the shaft. We'll likely have to move to a coupling to solve this problem in the future.

2.2 Electrical System Integration

There were a couple challenges with the electrical system that should be highlighted. To begin, the Cytrons we received from inventory had some interesting problems. Last semester the Cytron we received had a four block terminal block soldered to it, allowing for the power supply wires and motor wires to be easily wired to the motor controller. All the Cytrons we received this semester did not have these terminal blocks, necessitating us to find viable terminal blocks in inventory which we could use. We could not find a four block terminal block, leading us to have to use two different two block terminal blocks, which led to some weird soldering and integration issues. Thankfully our solders were good and the Cytron works perfectly. We received a second Cytron with attached terminal blocks as well, but that one was just broken sadly. Another issue we faced was delays in receiving our current sensors. We intended to receive the current sensors and integrate them with the system so that we could have force feedback, but due to issues with ordering they were not ordered on time and arrived late. We plan on having these current sensors integrated for the next progress review.

2.3 User Interface

The only challenge with the user interface is that we did not have it integrated with the ROS system as a whole, and as such I could not verify that any of the code I had written was functional. This was challenging to me as I have not worked on ROS as much as the rest of my teammates, and as such I am uncertain if any of the code I have written would be functional and work as intended.

3 Team Work

- Anthony: Worked with Kaushik to set up the task-prioritization framework, creating several new classes based on the software architecture, further setting up the simulation environment, and finally testing the framework in simulation. Anthony also worked with Parker to design the end-effector marker mount, providing feedback on the design, and helping 3D print some parts. Anthony also helped Sundaram to debug some of the Watchdog Module code, providing suggestions for code structure and CMake. And lastly, Anthony helped collect data for reaming on the pelvis encased in ballistics gel.

- Gunjan: Continued development on the UI module. She setup the basic wireframe of the UI on Open3D. She then completed the Image Alignment tool development that is able to display multiple pointclouds and transform the implant pointcloud using UI-based controls. Further, she collaborated with Parker and Sundaram to facilitate the integration of the watchdog module with the UI. Finally. she worked with Kaushik to calibrate the new end-effector marker and test its detection and tracking.

- Kaushik: Worked with Anthony in setting up the task-prioritization framework and testing it in simulation. He assisted Parker with wiring electronics and programming the reamer end-effector. He assisted Sundaram in setting up the ballistics gel encasing for the pelvis. Finally, he post-processed raw surgery data and conducted frequency analysis of the vibrations during reaming to validate the use of Ballistics gel as a proxy for soft tissue.

- Sundaram: Worked on developing the watchdog module by setting up a ROSCPP node and successfully compiling the CMake file with the necessary dependencies. For this, he worked with the owners of all the subsystems to finalize the functionality of the watchdog and the features that need to be developed. He made a decision tree that helped with the development of the subsystem and rigorously tested the inputs and perception subsystem working. He also worked on creating the ballistics gel mold for testing the pelvis model. He worked with Kaushik and Anthony to collect data by reaming the pelvis model submerged in the gel and analyzing the results generated. He discussed with Parker the integration of the Watchdog module with the User Interface and assisted him with evaluating the performance of the 3D-printed end-effector.

4 Plans

4.1 End-Effector Quotes

Now that we have a functioning 3D-printed end-effector, we hope to move forward with getting some of the more key parts of the end-effector manufactured out of aluminum (and also design a plastic cover for the end-effector). As such we will need to redesign a lot of our components (while keeping a lot of the geometry the same) to be lighter and thinner so that our end-effector does not get too heavy. We plan to reach out to a variety of vendors once we have finished this redesign to receive quotes from them and determine which parts we should get professionally machined and which we should attempt to machine in the MRSD machine shop ourselves. We plan to reach out to Tech Spark, Proto Labs, Fictiv, Xometry, and a contact that Anthony has overseas for cheap manufacturing.

4.2 End-Effector Controls Integration

For the next PR we need the end-effector to be completely integrate with ROS and the controls to be finalized. This will require integrating the current sensors into the system for force feedback and some changes to the Arduino code and how we communicate with the Arduino via ROS. We hope to be able to send a command to the end-effector to start reaming at which point it will begin moving into the acetabulum and once a certain force threshold is exerted start the reaming motor. After the ballscrew motor has actuated a prescribed amount or the bottom limit switch is actuated, the end-effector will return a done command to ROS, allowing for the procedure to finish. Throughout the reaming operation we want the Arduino to stream the rpm of both motors as well as the measured axial force to ROS so that it could be displayed on the User Interface.

4.3 User Interface

Quite simply, the user interface needs to be integrated with the rest of the system, allowing for us to display information from the system. Furthermore, the front end of the user interface needs to be developed further to have more screens throughout the procedure.