HIPSTER

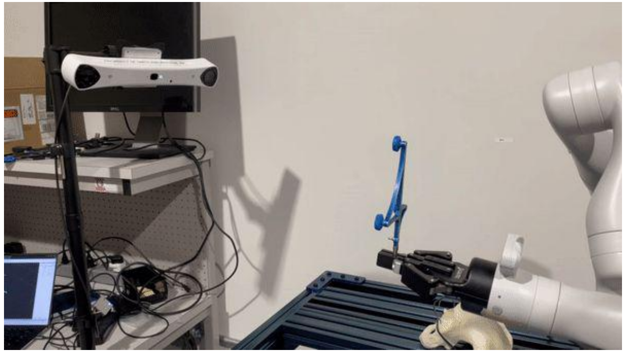
Autonomous **R**eaming for **T**otal **H**ip **R**eplacement **(ARTHuR)**

Progress Review - 3

Team C: Kaushik Balasundar, Parker Hill, Anthony Kyu, Sundaram Seivur, Gunjan Sethi

23 March, 2022

Previously



Validation of Test 4: Camera-ROS Integration Test

Hardware Setup

Schedule

Schedule			
Identifier	**Capability Milestone(s)**	**Associated Tests**	**System Requirements**
✔ **Progress Review 1** **2/16/2022**	- Test camera health and camera discovery via a ROS Node.	Test 1	M.F.1
✔ **Progress Review 2** **3/2/2022**	- Broadcast marker pose as a ROS transform & ROS topic - Validate the preliminary performance of the registration algorithm chosen	Test 2 Test 3 Test 4	M.F.1 M.F.3 M.N.1
Progress Review 3 **3/23/2022**	- Probe is able to be used to create a point cloud which can be visualized - Control method is capable of being used with robot manipulator virtually - Waypoint and trajectory generation working in ROS - Hardware verified for use in reamer assembly	Test 5 Test 11 Test 13 Test 17	M.F.1 M.F.2

Progress Review #3 Tests

- ✓ Landmark Capture Test — Perception and Sensing
- ✓ Waypoint/Trajectory Generation — Planning and Controls
- ✓ Position and Force Control in Simulation — Planning and Controls
- ✓ Reamer Motor Speed and Torque — Hardware

Further Updates

- ✓ Registration — Perception and Sensing
- ✓ Controls — Planning and Controls
- ✓ Simulation — Planning and Controls
- ✓ Hardware — Hardware

Progress Review - 3
Tests

Test 3: Landmark Collection Test

Goal: Read camera measurements from a ROS Node, identify the fiducial points with a pre-loaded geometry file and print the 6DOF marker pose, and store as a pointcloud for visualization on RViz.

Approach: Add functionality onto the previously developed camera_node to detect marker poses convert to a PointCloud2 message and publish as a topic.

Test 5 :	
Landmark Capture Test	
Objective	
Test the use of the registration probe to record fiducial landmarks on pelvis and test the ability to use Open3D to store the selected points as a pointcloud.	
Equipment	Atracys Sprytrack 300 Camera, Markers, Registration Probe MRSD Computer 2, Model Pelvis
Elements	Perception Subsystem
Personnel	Gunjan Sethi & Kaushik Balasundar
Location	NSH Basement
Procedure	
1. Use the registration probe to slide through the acetabuluar surface in the field of view of the camera. 2. Once done, run ROS script to visualize the captured points as Open3D pointcloud.	
Validation	
- The resulting visualization must be in the form of an Open3D visualization window. It must display the captured points that depicts the surface of the acetabular surface.	

Test 3: Landmark Collection Test

SNo.	Approach	Pros	Cons
1	Record **fiducial** 3D translation, closest to the tip of the probe	- Point closest to the tip of the pelvis	- Might not be more accurate compared to the marker pose.
2	Record **marker** 3D translation	- More accurate	- The transformation from point recorded and pelvis might be needed for better accuracy.

Test 3: Landmark Collection Test

Challenges:

- ➔ Obtaining the pose of the probe tip
  - ◆ Current algorithms in the SDK provide methods to only capture fiducials and marker positions; need a way to capture exactly probe tip position or learn a transformation between captured point and probe tip.
- ➔ Understanding sensor_msgs/PointCloud2.msg
  - ◆ Working with the incoming frames from the Atracsys SDK to correctly typecaste into PointCloud2 type.
  - ◆ Understanding the data array of the message.
- ➔ Testing against ground truth
  - ◆ Brainstormed to define tests to test the scale accuracy and effect of orientation on landmark collection.

Test 3: Landmark Collection Test: [Internal Test] Results

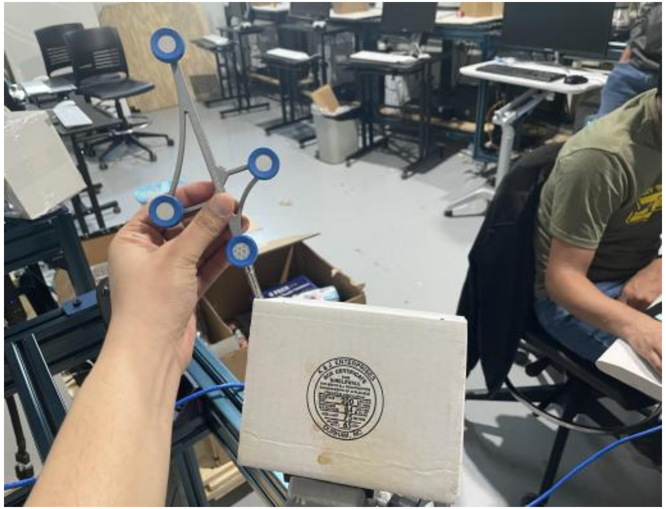
Goal:

- ➔ Test if landmark collection is at correct scale
- ➔ Understand effect of orientation of probe on landmark collection

Test: Obtain an object with known geometry. Record an initial point. Slide probe in one orientation towards an end-point along one dimension. Record the distance covered. Try various orientations.

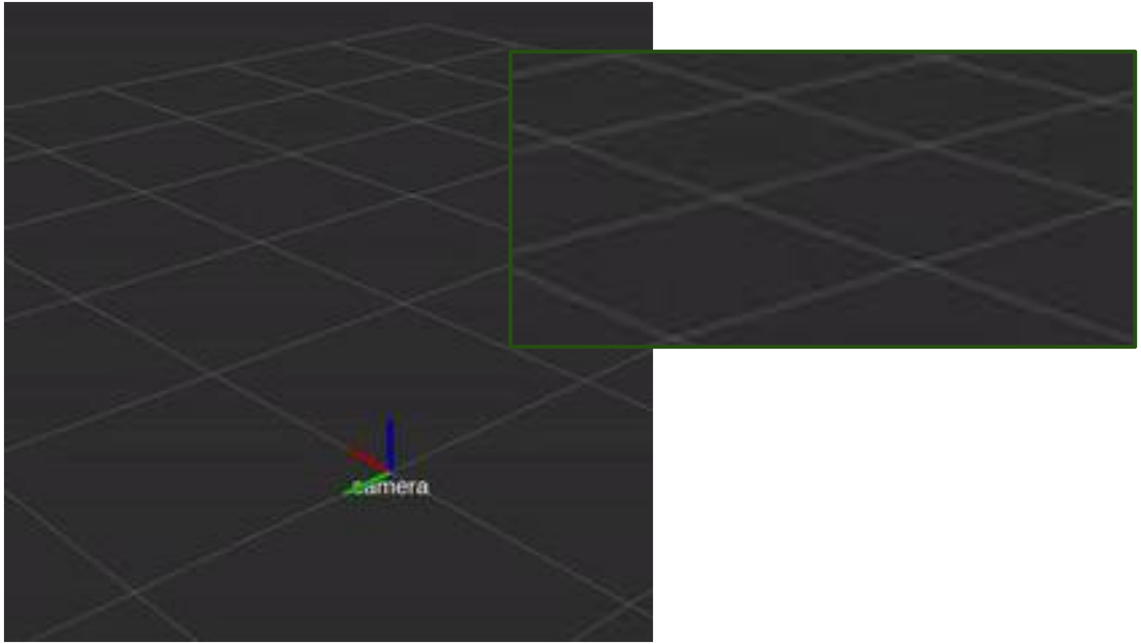
Results:

- ✓ Points up to 1-2 cm at scale + variations in orientation.



Landmark collection test on object with known geometry

Test 3: Landmark Collection Test: Results



Landmark Visualization on RViz

Pointcloud Collection using Registration Probe

Test 11: Waypoint/Trajectory Generation Test

Overview:

Goal: Test the generation of waypoints and trajectories using MoveIt! And verify that the arm moves along the trajectory in simulation and reality

Approach: Visual validation of trajectory following capability of Kinova Gen3

Test 11:	
Waypoint/Trajectory Generation Test	
Objective	
To test MoveIt to generate a trajectory from a start point(any point robot is left in space) to the end point	
Equipment	MRSD System 2, Arm, Markers, Reaming tool
Elements	Motion Planning
Personnel	Sundaram Seivur
Location	NSH Basement
Procedure	
1. Move manipulator close to the pelvis model using free motion mode 2. Mark current pose as start point for manipulator. 3. Run ROS script to invoke MoveIt to generate trajectory between start and end point.	
Validation	
- Check in simulation if arm is moving along generated trajectory. - Check visually if arm is moving along similar axis in reality.	

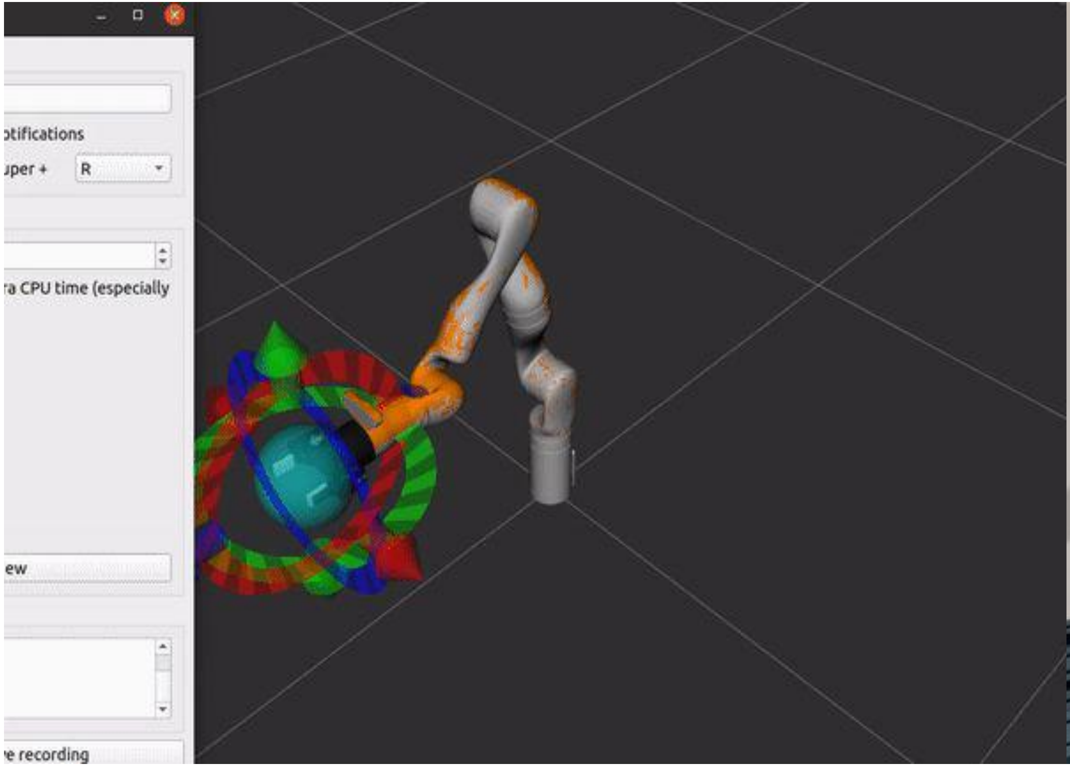
Test 11: Waypoint/Trajectory Generation Test

- Moved to using Pilz Industrial Motion planner instead of OMPL
  - Deterministic motion and repeatable trajectories
  - Essentially need only trajectory planning and not motion planning

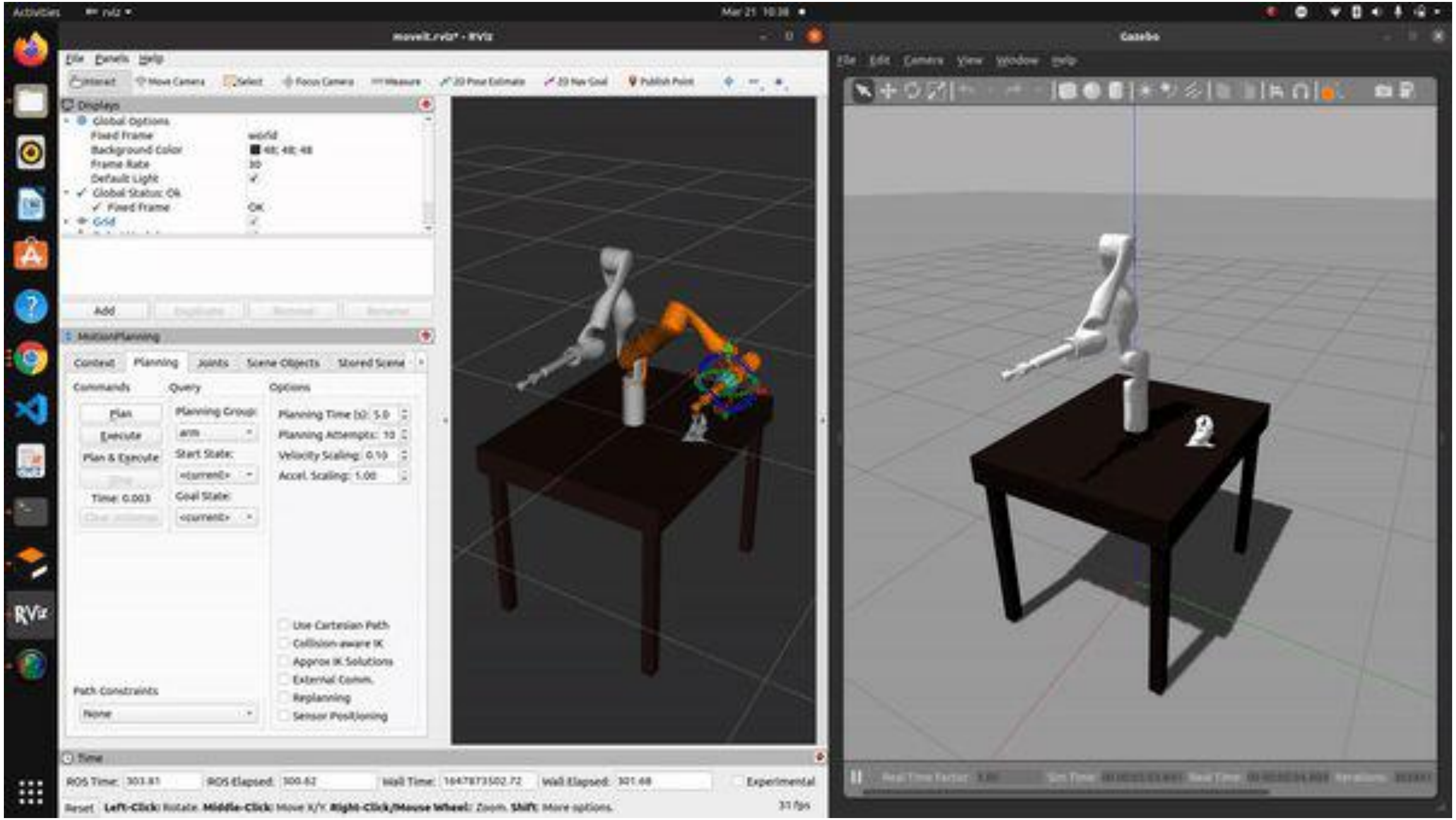
- Generated IKFast plugin to be used as IK solver instead of KDL
  - IKFast is an analytical solver in place of KDL which is a numerical solver
  - Repeatable and stable solutions

- Connected Gen3 via ethernet to validate if arm can communicate with external sources
  - Send random trajectories to be followed; generated using MoveIt!
  - Send joint state and cartesian pose command using a ROS node

Test 11: Waypoint/Trajectory Generation Test



Test 11: Trajectory using Pilz Industrial Planner



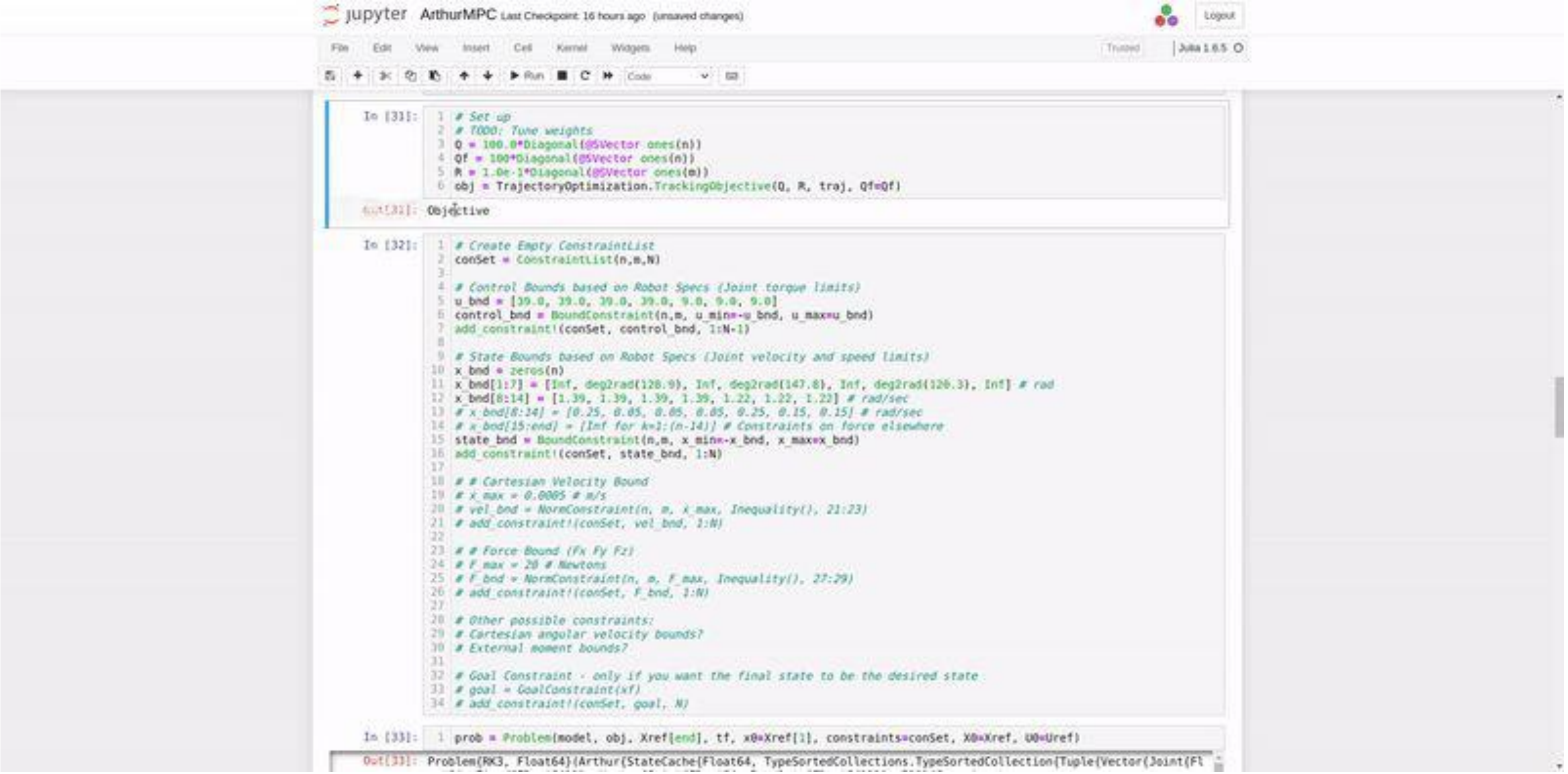
Test 13 (Revised): Position Control in Simulation Test

Goal: To test the initial capabilities of the MPC in following a trajectory given constraints (joint positions and velocities).

Approach: Develop MPC using pre-existing libraries such as ALTRO, TrajectoryOptimization, and RigidBodyDynamics in Julia

Test 13 :	
Position Control in Simulation Test	
Objective	
To test the ability of the MPC controller to move to desired positions without exceeding a joint position and velocity constraints.	
Equipment	System with Hipster Test Environment (MRSD Desktop 2)
Elements	Controls & Actuation Subsystem
Personnel	Anthony Kyu
Location	NSH Basement
Procedure	
1. Provide MPC with a desired trajectory containing joint positions and joint velocities. 2. Run the MPC solver and get joint torques along with corresponding joint positions and velocities. 3. Measure the error of the MPC final endpoint with the desired endpoint from the given trajectory.	
Validation	
- Pose error of the final endpoint of the trajectory should be +/- 2 mm and +/- 3 degrees.	

Test 13 (Revised): Position Control in Simulation Test



Test 13: Position Control in Simulation Test

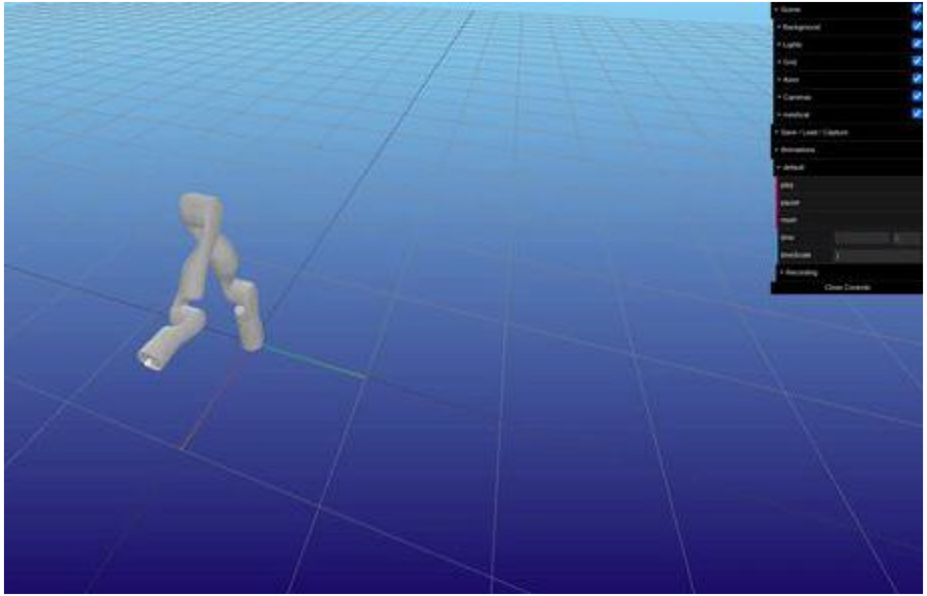


Figure: The MPC in Simulation

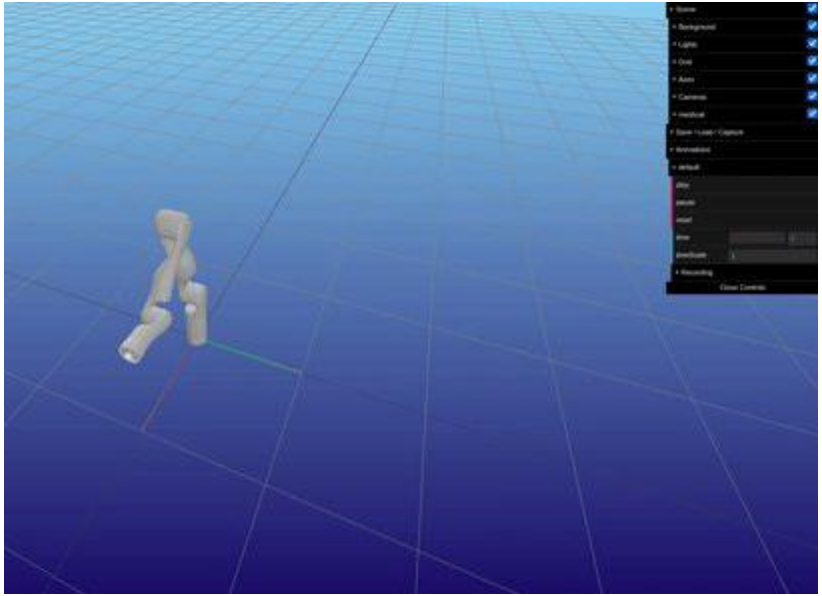
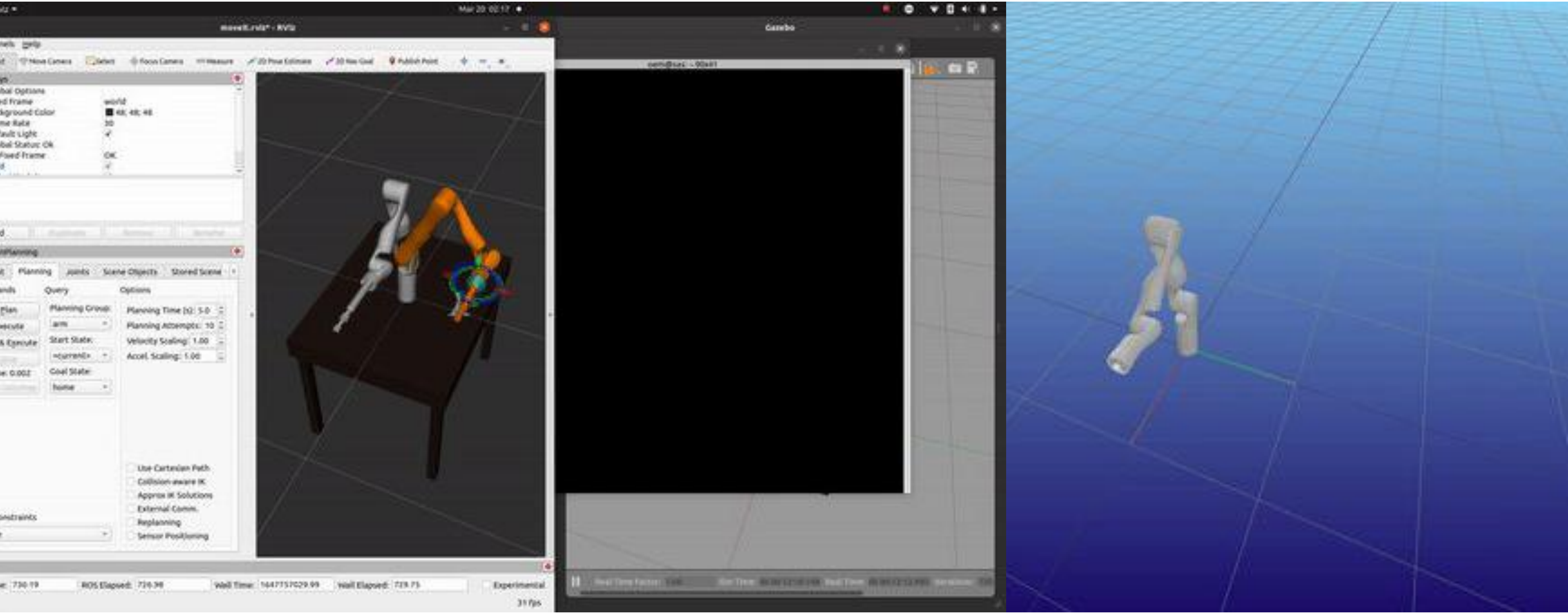


Figure: Desired Trajectory

Test 13: Position Control in Simulation Test



Test 13: Position Control in Simulation Test

Results:

✓ Positional Error < 2 mm

✓ Orientation Error < 3 degrees

Poses and Pose Error Relative to World Frame		
	Final Position (m)	**Final Orientation (RPY - degrees)**
Desired Trajectory	[0.214, 0.399, 0.436]	[89.764, -1.893, 91.168]
Model Predictive Controller	[0.214, 0.399, 0.436]	[89.764, -1.320, 91.166]
Norm of Error	0.000 mm	0.573 degrees
Joint State Error:	[0.00015959433561874015, -0.01370861054813316, -0.00020415288308361923, -0.006550863538767437, 0.004910735548274481, -0.0025631207105871745, -0.0039981932771855355, 1.1578955204986224e-5, 0.0025025376776104894, 5.748741903614427e-5, 4.2852715328442655e-5, -0.00028492385974444694, 2.2142163532343218e-6, 0.0001753074183847769]	

Test 13: Position Control in Simulation Test

Challenges:

➔ *Model Predictive Controller was challenging to get to converge*
  - Not much documentation to help solve issue
  - Convergence hinges a lot on initial guesses for desired input/torque trajectory
    - Initial guess is torque required for gravity compensation at every trajectory point

➔ *Tuning Q and R matrices of MPC*

Test 17: Reamer Motor Speed and Torque Test

Overview:

Goal: Test the torque and speed of the motor and gearbox, verifying its ability to output the necessary torque to ream the acetabulum

Approach: Utilize rough estimations to verify that the performance of the motor is satisfactory

Test 17 :	
Reamer Motor Speed and Torque Test	
Objective	
To test the torque and speed of the motor and gearbox and verify it's ability to output the necessary torque for the reamer to properly function to ream the acetabulum	
Equipment	System with Hipster Test Environment (MRSD Desktop 2), motor and gearbox from reamer assembly, power supply, video camera, long piece of wood, weights
Elements	Reaming subsystem of the hardware system
Personnel	Parker Hill
Location	NSH Basement
Procedure	
1. Hook up the motor to the power supply and increase the applied voltage to 24V 2. Using a video camera, measure the approximate no load speed 3. Turn off the power supply and attach a long piece of wood of a specified measured length to the motor shaft 4. Add weight to the end of the piece of wood and measure the applied torque, turn on the power supply and determine if the motor is capable of moving past an orientation where the wood is parallel to the floor 5. Repeat step 4, increasing weight each time until the motor is incapable of moving past the specified orientation, mark the resulting torque as the peak torque of the motor 6. Using the no load speed and peak torque, approximate the max torque at a speed of 400 rpm	
Validation	
- With a speed of approximately 400 rpm, the associated maximum torque of the system should be greater than 1 Nm	

Test 17: Reamer Motor Speed and Torque Test

Speed Test:

- Tested the no load speed and found that the rpm is ~600
- No load current was also verified to be 0.52 A
- Test was performed utilizing a video camera and counting the rotations in a 5 second period of time

Torque Test:

- Test performed using a custom motor attachment and a scale in lieu of not having access to a dynamometer
- Approximate stall torque at 4V and 5A found to be around 3.5 kgf-cm, which when scaled provides a potential stall torque of 14 kgf-cm
- With a proper power supply we are confident that the reamer motor will be able to provide the speed and torque we need to ream the acetabulum

A. Operating Conditions:

1	Operating Voltage Range	6~12	VDC	4	Operating Temperature	-10~+60	°C
2	Rated Voltage	12	VDC	5	Storage Temperature	-30~+85	°C
3	Rated Load	2.3	kgf-cm	6	Test Position	Horizontal	~

B. Electrical Characteristics:

1	Max. No-load Current	0.52	A	6	Max. Stall Current	20	A
2	No-load Speed	612±61	rpm	7	Insulation Resist.(500V)	20	MΩ
3	Rated-load Current	2.0	A	8	Dielectric Strength	250	VAC
4	Rated-load Speed	540±54	rpm	9	Motor Brush Type	Graphite	~
5	Min. Stall Torque	16	kgf-cm	10	Output Power at Max.Eff.	13	W

C. Mechanical Characteristics:

1	Gear Type	Planetary	~	7	Max. Shaft Radial Load	3.5	kgf
2	Gear Ratio	289/3969	~	8	Max. Shaft Runout	0.05	mm
3	Gear Material	Metal	~	9	Max. Shaft End Play	0.30	mm
4	Rated Tolerance Torque	10	kgf-cm	10	Bearing Type	Dual Ball	~
5	Moment. Tolerance Torque	20	kgf-cm	11	Net Weight	330±20	grams
6	Max. Shaft Axial Load	2.5	kgf				

Hardware Update: Reamer End-Effector

Update

- Got the reamer end-effector fully 3D printed and attached to the Gen-3

Challenges:

- The side piece contributes a lot of wobble into the system
- Need shorter screws or for the hole depth to be increased in the force-torque sensor adapter
- Too long, needs to be shortened

Future Work:

- 3D-print new components
- Make or receive shorter reamer handle

Figure: Reamer End-Effector

Hardware Update: Motor Control PCB

Update

- Finalized our motor control PCB

Challenges:

- Had to import, create, and edit many custom parts in Eagle
- Finding some parts in stock ended up being an issue + part elicitation is quite difficult

Future Work:

- Receive PCB and components
- Solder all parts onto the PCB
- Test PCB for efficacy

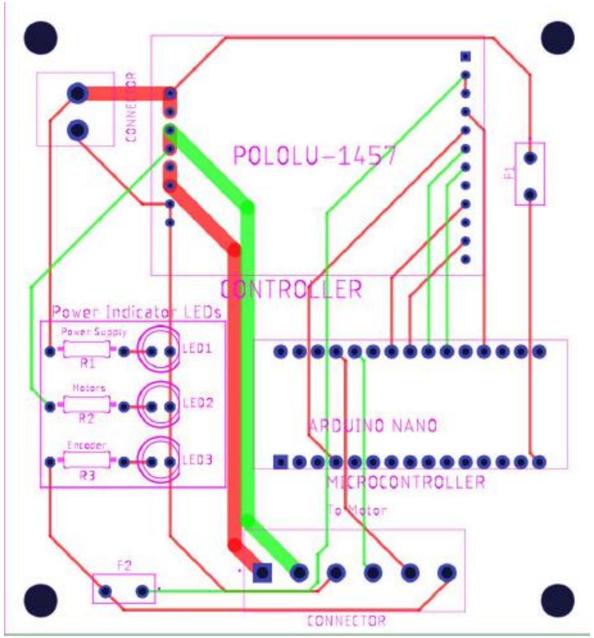


Figure: Motor Control PCB layers from FreeDFM

Plans : Progress Review 4

Progress Review #4 Tests

- ❑ Point Cloud Registration Test — Perception and Sensing
- ❑ Waypoint Generation Compensation — Planning and Controls
- ❑ Position and Force Control in Simulation and Reality — Planning and Controls
- ❑ Full Hardware-Test — Hardware

Perception Future Work

- Obtain probe tip pose from the marker geometry (upcoming discussion with sponsor today)
- Explore different publishing frequencies and number of landmark points collected
- Test registration using acquired Pointcloud and CAD model of Pelvis
- Explore manual correspondence matching using predetermined keypoints instead of using a feature detector such as FPFH
- Finalize a method to obtain initial transformation guess for ICP registration
- Evaluate quantitatively effectiveness of using ICP for cross-point sets registration

Motion Planning & MPC Future Work

- Read end-point transformation from perception subsystem and plan trajectory
- Validate error of generated trajectory and trajectory followed in reality using RPG package
- Publish trajectory directly to controller via a topic
- Provide cartesian states from trajectory generator to controller
- Implement additional states (cartesian states and wrench) into MPC
- Implement ROS Nodes that do MPC calculations
- Fully integrate Motion Planning & MPC

Hardware Future Work

- Assemble and test motor control PCB
- Mount power supply and motor control PCB to Vention table
- Redesign end-effector for improved rigidity and decreased length
- Verify full hardware setup efficacy

Thank you!

Questions & Discussion