Individual Lab Report
Weekly Progress

Brian Boyle
Team B: Team Space Jockey
Teammates: Nathaniel Chapman, Ardya Dipta Nandaviri, Songjie Zhong

ILR05
November 19, 2013

1. Individual Progress

My primary task since the last progress review was to generate a gait to move the robot forward. For simplicity, I decided to design a square-shaped gait whereby each segment lifts and places in strictly vertical motions, and extends or retracts in strictly horizontal ones. The resulting geometry for these motions could be described entirely with simple trigonometry. Given this, and the fact that we chose servo motors that exceeded our expected torques, I decided to ignore dynamics for the time being and focus strictly on inverse kinematics to generate joint trajectories. As a result, I was able to generate a gait very quickly, and our torque tests with the assembled prototype have made me confident that the robot will be able to achieve the prescribed poses without difficulty.

For exploring different variations of this gait, I added an “Operational Parameters” group of widgets to the GUI (See Figure 1). These new widgets control the vertical clearance during horizontal movement of segments, the maximum extension of the linear actuators, the delay between individual joint commands, and the resolution or granularity of joint commands during lift/place and extend/retract movements.

The completion of the basic “Square Gait” (See Figure 2) coincided with the completed fabrication of our second partial prototype (See Figure 3), made up of the two end segments. Unfortunately, this also coincided with Nate falling ill, so I took up the task of assembly. On my own I was able to put together most of the prototype, and when Nate returned we were able to finish it together in a little more than an hour.

Finally, after many disappointing attempts at molding our sticky footpads effectively, I came up with a new concept for molding, then designed and printed a proof of concept (See Figure 4) to demonstrate for the team. The resulting pad was at least as sticky as any one we’d fabricated before, but the primary advancement was in molding technique, which was both easy and effective in producing a smooth surface.

2. Challenges/Issues

The first challenge in adding gait functionality was designing an interface. I decided to partition the gait cycle into nine discrete steps which tackle the lifting, extension/retraction, and placing of each of the three segments. To make room for these buttons, I removed the canvas widget in the “Joint Configuration” pane that had been reserved for a simulation view, which is now being provided by the separate Rviz window. Integration of an Rviz simulation directly into the is preferable but not necessary at this time.

While the geometry of gait generation was not difficult, the initial implementation was kludgy and inefficient, so I then worked at generalizing and refactoring the code for organization, optimization and readability. This should also ease the process when I move gait generation to its own node..

Assembly was challenging mainly because Nate was inaccessible, so I only had the CAD assembly file and the first prototype to work from, but I can now say I thoroughly understand the entire mechanical structure of the prototype. The only error made during assembly was the stripping of a screw on a servo motor while changing its attachment, which we should be able to extract in the R.I. machine shop.

Thus far we've molded the foot pads by placing the foot face-down in a cast, resulting in a bond so strong that the toes snap right off the foot when we try to remove it. It occurred to me that this was partly a result of the adhesive on the cast walls resisting shear forces, and that we may be able to use this in our favor. My design adds a retaining wall around each toe, so that the resistance to shear forces help to hold the pad in place. A casting ring is snapped onto the toe, into which adhesive may be poured from above. After the compound dries, the ring can be removed. The resulting surface is much smoother than that of our prior results, and the retaining walls also make it much more resistant to peeling off the toe.

3. Team Work

While I was working on gait generation, the rest of the team was focused on producing the final prototype for this semester. Nate produced a new iteration of the end segment designs and laser-cut the chassis pieces, aided the final stages of assembly, and also developed a calibration sequence for the working actuators on the assembled prototype.

Songjie worked to develop and prototype the passive ball joint in each foot, with encouraging results. He also applied my foot molding technique to a new foot prototype (See Figure 5). The result, however, uses circular toes, which I had intended only for simplicity in rapidly designing my POC. In the coming week I intend to work with him to apply the retaining wall and casting ring concepts to wedge-shaped toes, which I believe have a better chance of peeling off gracefully.

Dipta's avenues for progress seemed limited by the fact that our B.O.M for electrical parts had not yet arrived. Nevertheless, he was able to test the connections on both our printed PD boards and found no problems, as well as assist Nate and me with the final stages of assembly. Additionally, he worked independently and with Songjie to perform some more exploratory foot molds.

4. Future Work

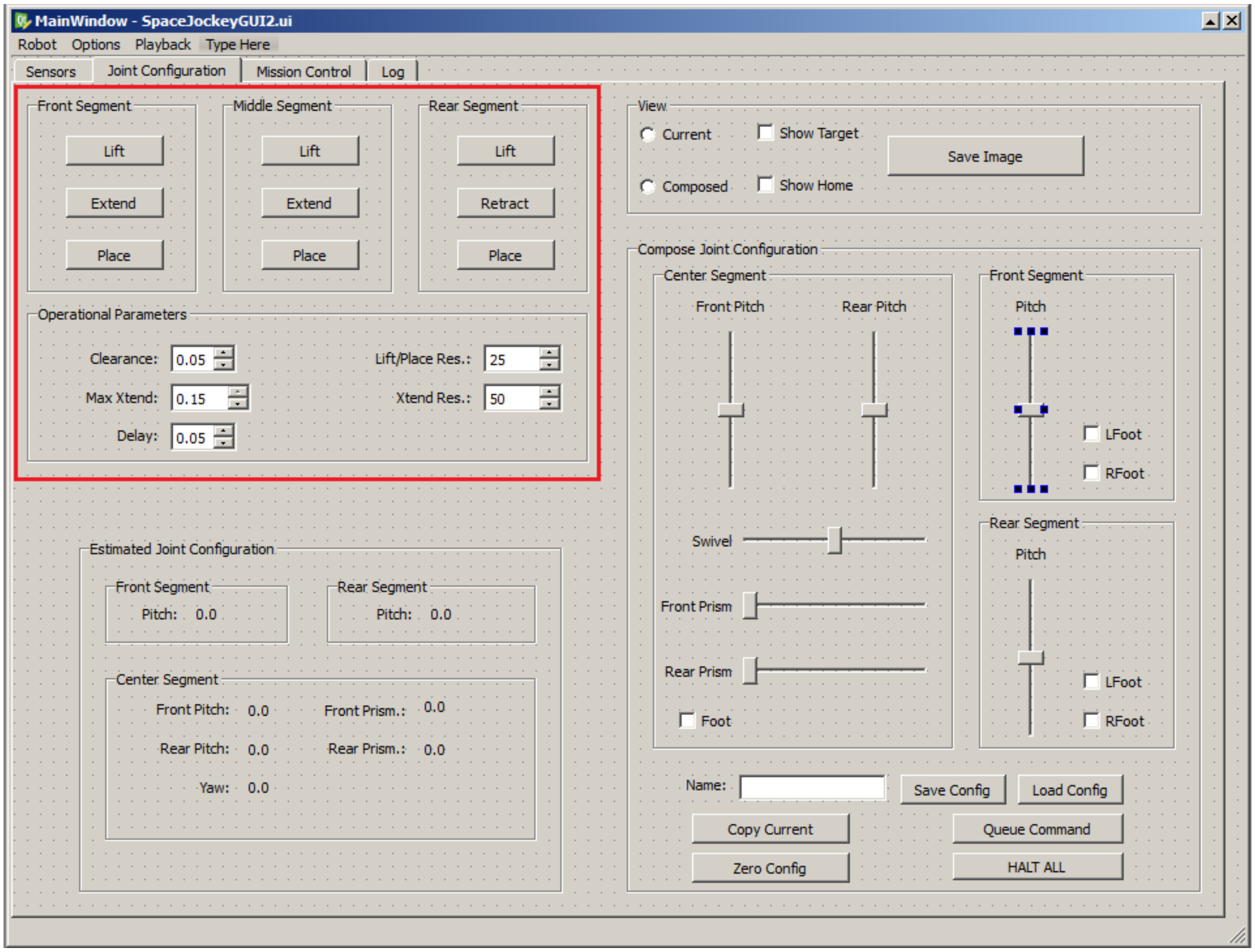
With a testable gait ready, my top priority is integrating my software with the robot's onboard computer to achieve our validation test goals. This will require setting up a ROS listener on the Arduino which will subscribe to the same topic used to pass joint commands between the GUI and RViz, and assisting the team wherever I can in the other aspects of final integration.

I expect there will be unforeseen issues from the onset, as is often the case in system integration, so I'll be working to establish this initial communication to the Arduino as early as possible and leave plenty of time for debugging. Beyond that, the actions described below are supplemental goals of mine, which I'll undertake if I have spare time to improve the control software.

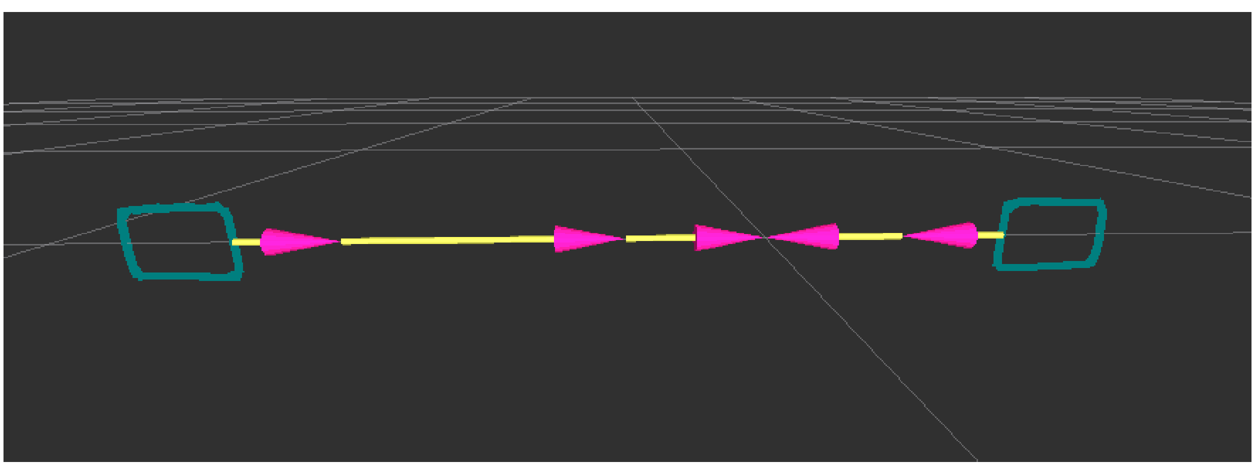
Presently, gait generation is performed in the GUI thread, making the GUI itself unresponsive. This renders the emergency "Halt All" button useless except when the robot is at rest. To remedy this, I plan to move gait generation to another node and also implement a scheduler or queue for joint commands.

I'd also like to develop a state machine to govern the set of controls available to the user. In the nine-step gait sequence, the user should only be able to move a single step forward or backward in the entire gait sequence at any time. Additionally, the user may wish to adjust the positioning of a lifted segment using the GUI's slider widgets. To enable this, only sliders for lifted segments should be available, and gait generation should be generalized so that motions can be performed from a variety of starting points for each segment.

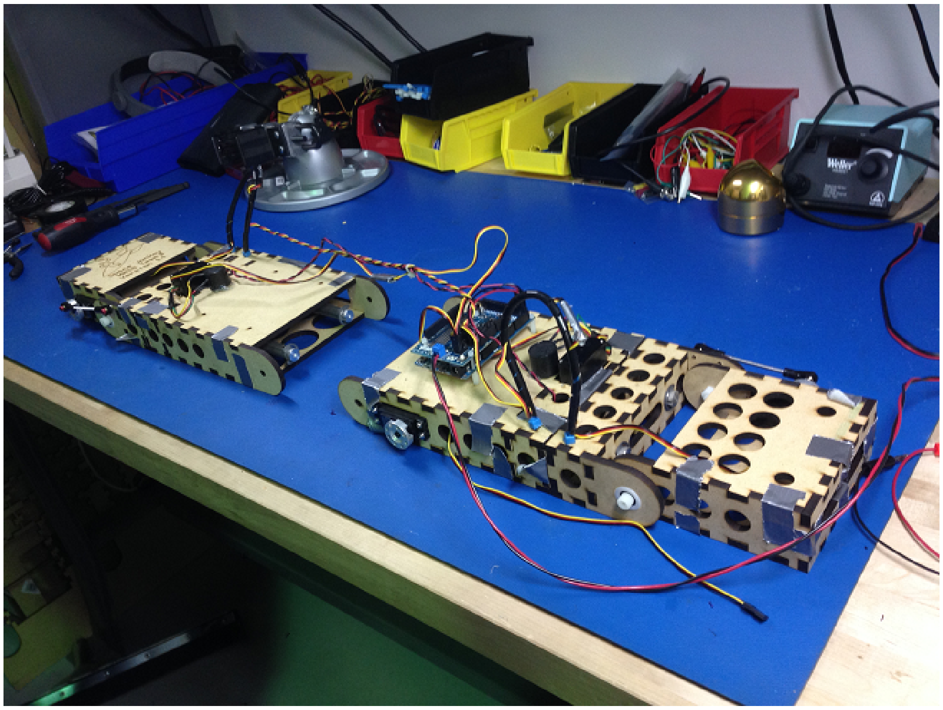
5. Figures



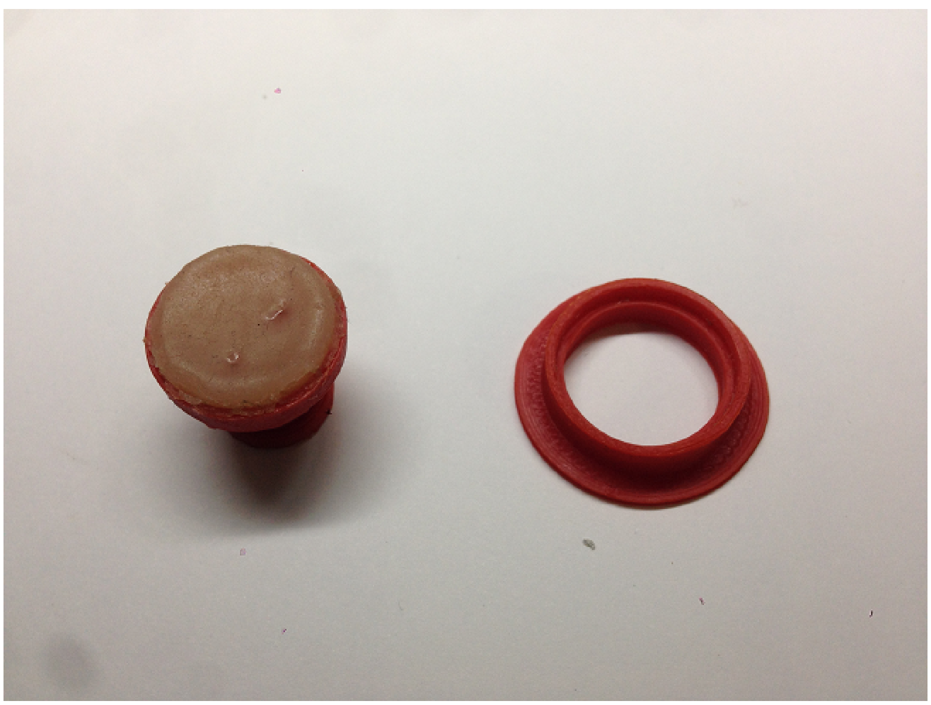
(Figure 1: GUI with new gait sequencing widgets)



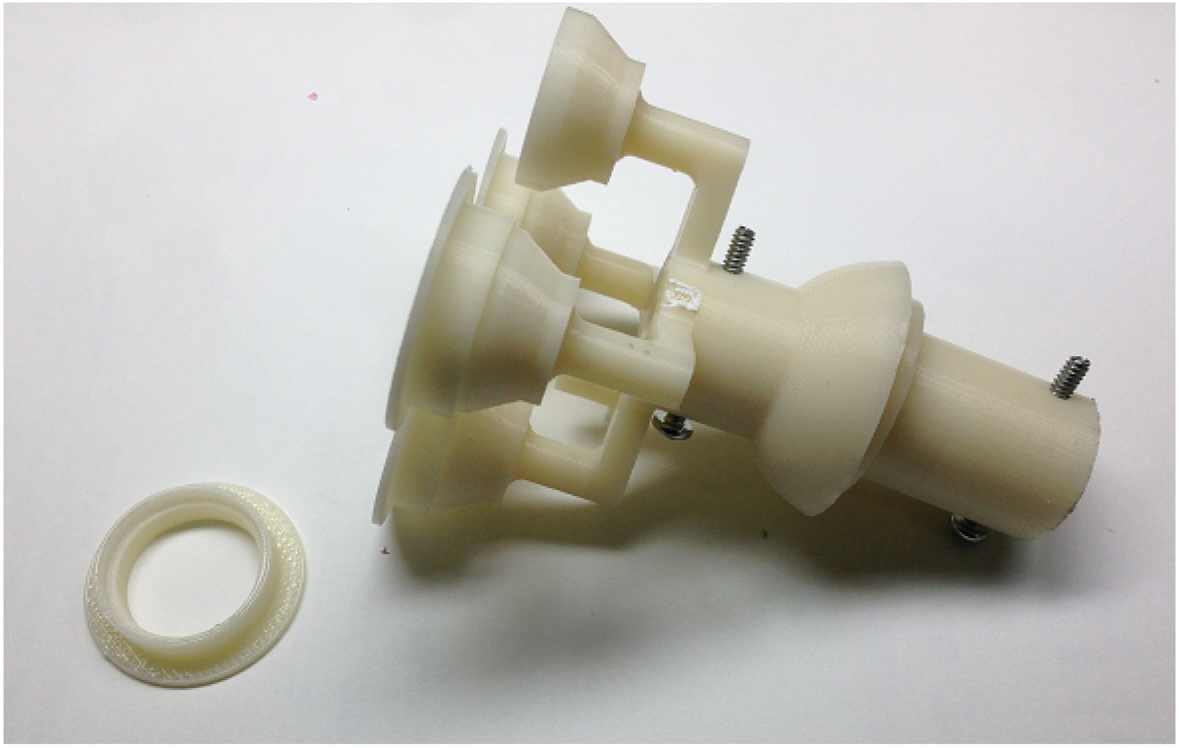
(Figure 2: Square Gait. The yellow lines with purple arrows represent the links and joints of the robot. Important to note is the blue rectangles, which show the trail of motion of the end segments during the Square Gate)



(Figure 3: Second prototype, with both assembled end segments)



(Figure 4: Proof of concept toe and casting ring)



(Figure 5: A new foot design, with working ball-joint and ring-cast feet)