

Jorge Anton Garcia

Team D – CuBi

*Teammates: Laavanye Bahl, Paulo Camasmie, Changsheng Shen
(Bobby), Nithin Subbiah Meganathan*

ILR06

Sept. 11, 2019

Individual progress

This week I worked on planning the semester, cleaning up the code, fixing the control issues we were having last semester, resetting odometry and performing the timing analysis.

Cubi Technical

On the technical-side, one of our biggest risks is not being able to pick up 15 objects in 15 minutes given the following assumptions:

- 20m^2 room (shown below)
- Viewing angle of 90° (45° to the right and left)
- Viewing distance of 1.2m

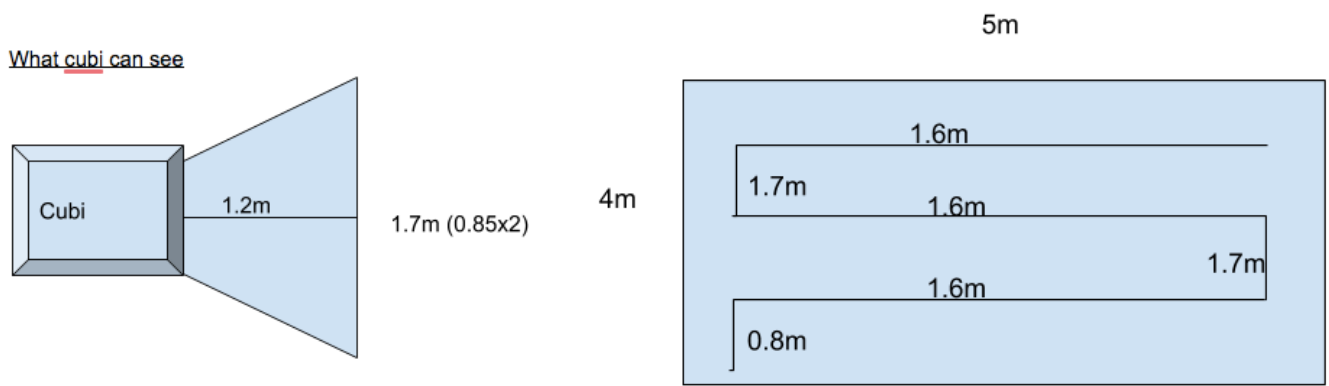


Figure 1. How cubi could traverse room and what it can see at all moment in time.

So assuming a $3\text{m} \times 4\text{m}$ room with no obstacles, it needs to go horizontally 3 times ($5/1.7$). To explore the it needs to follow the path shown above. Remember, cubi can see 1.2m in front, so it does not need to go 4m horizontally only 1.6m ($4 - 1.2\text{m} - 1.2\text{m}$). If we pick up 15 toys and have to drop of each toy to the box = 23.2 min. This latter measurement includes obstacle avoidance, picking up and dropping objects, going to the box, going to pick up the object, and missed attempts. I created an excel with parameters we can tune to estimate the time it will take given different assumptions. Based on these, I was able to estimate improvements we could achieve by working on specific subsections. These can be found below:

- Doubling cubi's speed can save: 4 min
- Halving the time to pick up an object saves: 3.5 min
- Halving the time to drop off an object saves: 3.5 min
- Picking up two toys at once can save: 10 min

We need to analyze how difficult it is to pick up two toys at once. Maybe we can do so for only the smallest toys and can save 5 minutes there. This would allow us to relieve the pressure on speeding up cubi. If it's not possible without making too many changes, then we need to focus most efforts on making cubi quicker and very agile at doing what it has to do.

I also worked on resetting odometry. Whenever Cubi sees an AR tag with our realsense, it knows where the box can be located and hence reset its odometry. Right now, the function just uses the position of the AR tag as the 0 position for the newly reset odometry. We will need to add a transform, so cubi knows where to drop off the toys and does not crash into the box. There is a wait that limits the time between odometry resets. In the future we may also want to have cubi stop when resetting odometry.

CuBi Project Management

I ported all of our previous project management tools into notion. Before we were using Slack, Trello, Whiteboard images, and Whatsapp for division of work and communication. This resulted in very sporadic planning streaks, action items being lost between meetings, lack of transparency, and poorly integrated subsystems. I used Notion for my startup and we jumped in our ability to finish tasks and track risks.

Notion acts like a large excel in which you can save pre-defined views. These views can be different filters or sorting options in each table, viewing the table as a canban similar to Trello, or in calendar format. Soon there will be a gant chart!

Task	Team Divi...	Project	Due Date	Status	Assign	PR
Project Management Review	Coursework		Sep 04, 2019 3:44	Done		7
General Planning 1	PM		Sep 05, 2019	Done	J Jorge Anton	7
Lab cleanup 2	Coursework		Sep 09, 2019	Done	J Jorge Anton P Paulo Can	7
Define Software Interface 3	SE-General		Sep 09, 2019	Verification	Laavanye Bahl J Jorge A	7
Freeze and clean up git documentation 2	SE-General		Sep 10, 2019	Done	Laavanye Bahl	7
Take dimensions CuBi and Caster for attachm	Mobility		Sep 10, 2019 2:00	Done	P Paulo Camasmie	
Controls Issues from Last Semester	Controls		Sep 11, 2019	To Do	J Jorge Anton	7
Dropping issues from last Sem OPEN	Planning		Sep 11, 2019	To Do	J Jorge Anton	7
Localization Issues from Last Semester	State Estimatio		Sep 11, 2019	To Do	J Jorge Anton	7
Simple 2D Lidar SLAM 2	State Estimatio		Sep 11, 2019	Done	C Changsheng Shen	7
ILR 06	Coursework		Sep 12, 2019	To Do	J Jorge Anton P Paulo Can	7
Task 1 PR #7 1	Coursework		Sep 13, 2019	In Progress	J Jorge Anton	7
clean the mobility controls code	SE-General		Sep 13, 2019	To Do	J Jorge Anton	
Prepare for meeting	Other		Sep 14, 2019	To Do	J Jorge Anton P Paulo Can	
Add tickets to notion 1	PM		Sep 15, 2019	To Do	J Jorge Anton N Nithin Me	7
Create structure/code framework which reser	SE-General		Sep 15, 2019	To Do		
Consider Transitioning from Behavior tree to s	SE-General		Sep 16, 2019	To Do	J Jorge Anton	7
Acceptance test for master merge	GIT		Sep 16, 2019	Not Started		

Figure 2. Screenshot of this weeks tickets.

Comments in tasks allow for good documentation and clear status updates per item. We are able to estimate the amount of hours each task will take allowing us to roughly estimate how much work everyone is putting in. In every meeting, we check what tasks have been completed and this keeps everyone accountable. It helps us keep track of risks too.

Challenges

Technical

The biggest technical challenge I worked on was that cubi would rotate back and forth, as if it were dancing, when trying to rotate 180°. To go to each point and align itself in the desired direction at point cubi followed three steps. It would first orient itself towards the goal, it would then drive towards the goal, and then orient itself again (with a different angle) towards the desired final orientation.

I found that it was a combination of three different issues:

- Wrap around when calculating difference between transforms
- Using thresholds between each of the three stages. For example, take the case where cubi is at a threshold. Before the threshold cubi is far from its goals, so it would try to orient itself and go towards the desired waypoint. When it passes a threshold its objective is to orient itself to the final orientation. However, since cubi is at the threshold, when trying to orient itself towards the final orientation, it would exit the threshold and want to orient itself the other way towards the waypoint. Then it would get back in the threshold and it would try to perform the final orientation again. The dance was caused by entering and exiting the threshold.
- To make matters worse, we used cubi's tray as the frame we wanted to align with the goal. When cubi turns, the tray is the farthest part from the center of rotation (40cm away) causing it to not rotate in place. Because of this, whenever we had to rotate 180°, the tray would always bounce around from inside the threshold to outside of it.

The fix was to use two different thresholds. One smaller one which caused cubi to align almost perfectly at the waypoint and one larger one, so that cubi would not exit it when performing the final orientation.

Team Work

Nithin: I worked directly with him to create more granular tasks to plan out the semester efficiently.

Paulo: He analyzed our current wheels and determined that our traction issues are most likely caused by the small caster wheels. He created a preliminary design for a larger caster wheel. This is going to make it easier to create good controls.

Laavanye: Cleaned up our GIT, designed a process of how to get code merged in master, and cleaned up the vision code.

Bobby: Worked on 2D SLAM using Hector SLAM. He also cleaned up the teleoperation and manipulator control code.

Plans

I will finish planning the end of the semester. I was blocked by some team decisions we will be making this weekend. I will integrate the odometry reset with the rest of our code. Finally, I will get more familiar with the vision portion, so I can help Laavanye.