

# INDIVIDUAL LAB REPORT #7

10/01/2020

## AWADHUT THUBE

Team G - The Pit Crew

Team Members: Alex Withers, Justin Morris

## Individual Progress

For progress review 8, our team had two major goals. One was to add the edge detection state to the webots simulator, and the other was to create a pit capturing control sequence for gathering data of the pit. The summary of both these tasks is given below:

1. Add edge detection state to the sim: This task involved achieving basic brinkmanship functionality in the simulation and integrating it with the previously developed navigation state machine.
2. Creating a pit capture control sequence: The skeleton for this was developed during the spring semester and that was further refined during the build up to this progress review.

My work for this progress involved development of the edge detection state and integrating it with the navigation state machine. During the last progress review, we were able to add a stereo camera pair on the simulated rover and also published the raw point cloud onto a ROS topic. I started by accessing the raw point cloud published by the simulated camera and transforming it into the base\_link reference frame. The transform between the camera frame and the base\_link frame is static and is defined in the world file that the simulator loads. Once the cloud was obtained in the right frame of reference, I moved on to develop the code for using a heuristic to determine the presence of a pit edge. We basically use the number of points in the pointcloud to determine the presence of a pit edge.

I also developed a separate node which would publish the rover twist messages to command it's motions. Once the rover reached a highway point, it would receive twist messages from this node instead of the global planner. The rover thus moves closer to the edge of the pit. The rover moves forward with a velocity of 5 cm/s. Once it detects a pit, it recedes with a velocity of 10 cm/s. After it moves back a certain distance, the rover reaches safe territory and the brinkmanship node transfers the motion control to the global planner. Thus, the rover uses the twist messages published by the global planner to move back to the highway point

The information from the pointcloud is relevant just to inform the rover of the pit edge. Hence, this information is just used to generate a boolean variable which indicates the presence of a pit edge. I created a helper class which stores polls the pointcloud and makes the decision regarding the presence of a pit edge using it. The object of this class stores the boolean value which informs the state machine to transfer control to the brinkmanship node.

Alex helped me integrate this piece of code with the navigation state machine that he had been working on. Together, we were able to execute the entire pipeline using predefined highway waypoints along the circumference of the pit.



Fig: Rover Near The Brink

### Challenges:

While working on the desired goals, I faced two major challenges. One challenge was to actually run the simulation fast enough to allow myself to unit test the developed code. The simulation ran very slowly on my personal computer with a real time factor of about 0.2. To get around this problem, I cut down some visualization/rendering aspects of the simulation. First, I tried turning off the shadows in the simulator and then ran the entire simulation without visualizing the scenes. However, these modifications had little improvement on the real-time factor. Eventually, we ended up placing the rover close to the pit edge and tested the pipeline starting close to a highway point. This really helped to test the developed code quickly.

My second major challenge was integrating the code with the navigation state machine. Alex had been working on this part of our system for a significant amount of time and knew the exact hooks for integrating both modules. He helped me integrate and later test the code.

### Team Work:

Alex Withers: Alex helped me to get familiarized with the webots simulator. He had the most experience in working with the simulator and is also managing our project GitHub repository. He also helped me integrate and test the brinkmanship node. He also started work on our targets for the next progress review, i.e. integrating RASM in our system for local planning. Alex also recorded all content for the actual presentation during progress review 8.

Justin Morris: Justin was responsible for making changes to the previously developed pit capture sequence. He made four major changes which included, parameterizing the code to accommodate changes in camera parameters and also generating different panoramas for different tilt angles of the rover's stereo camera pair. He also modified the system to generate panoramas online instead of saving all images and post processing them for panorama generation.

Future Plan:

Now that we have achieved the basic brinkmanship functionality, we would be looking to improve the heuristic used to determine the presence of a pit edge. Specifically, we aim to convert the simulated point cloud into a triangle mesh and use the mesh to not only determine the presence of an edge but also estimate the distance to the pit edge. Our other goals include making progress towards integrating the RASM local planner into our system.