# PROGRESS REVIEW 9

Astha Prasad

Team F: ADD_IN

ILR #8

February 25, 2016

Teammates:

**Daniel Berman**

**Nikhil Baheti,**

**Ihsane Debbache**

# Introduction

For Progress Review, the primary software tasks that were tackled were as follows:

- Check the function that compute's R angles to fix the problem of spurious 'NaN' values
- Switch from absolute angles to relative angles
- Implement a collision check function

## The NaN issue

After the previous PR, it was Dan and I noticed that for certain printer states, the R angle was turning out to be 'NaN'. This occurrence can be seen in the image below:

```
G1 X97.498 Y75.229 Z5 R4.3987 E2.2106 F900
G1 X97.511 Y75.078 Z5 R4.9206 E2.2144 F600
G1 X102.436 Y75.078 Z5 R90 E2.3402 F600
G1 X102.421 Y75.213 Z5 R6.3402 E2.3402 F7800
G1 X102.421 Y75.213 Z5 RNAN E0.34018 F2400
G92 X102.421 Y75.213 Z5 RNAN E0 F2400
G1 X97.328 Y75.171 Z5 R-269.5275 E0 F7800
G1 X97.328 Y75.171 Z5 RNAN E2 F2400
```

**Figure 1 : GCode showing NaN values**

It was observed that this occurred only when the X and Y coordinates of two sequential G-Codes remained the same. The velocity vector is calculated using the equation:

```
v = (p(1:2)-cp(1:2))./norm(p(1:2)-cp(1:2));
```

where 'p' and 'cp' are the XYZ coordinates of the current and the next printer states respectively. As can be seen, the denominator evaluates to zero in the case mentioned above which resulted in the 'NaN's. This case was handled smoothly by adding an 'any(isnan(v))' check on the velocity vector. If it evaluated to true, the R angle was set to zero.

## Absolute to Relative angles

Nikhil firmware currently works on the assumption that it receives relative angles. Hence, an additional variable called 'r_relative' was added to the PrinterState class to store relative angles along with absolute ones that were being computed earlier. The above was achieved by simply subtracting the absolute angles of two sequential PrinterStates. The function that writes the GCode into a .gcode file can be changed to write absolute or relative values as required.

## Collision Check

The collision check functionality was added to check cases such as those illustrated in the figure below:

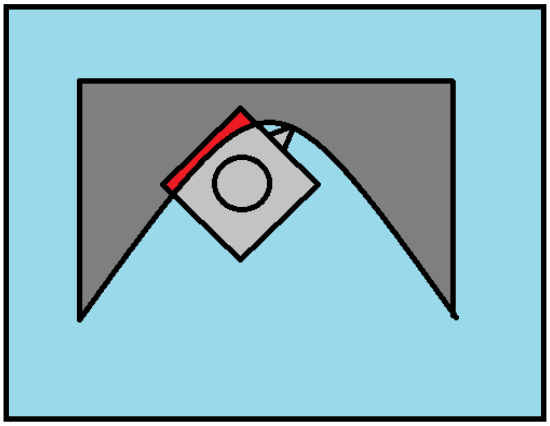

**Figure 2 : Occurrence of Collision**

The figure above shows the top view of the print bed. The area in blue shows the part of the print that the dark grey COTS item rest on. In this particular case, we can see that the nozzle is in collision with the COTS part itself because of its convexity. This is exhibited by the red portion of the nozzle. In such a case, the function will alert the user that the COTS part is likely to collide with the nozzle. As a result, the user might have to redesign the part/the COTS item to suit the printer's capabilities.

Our project requirements dictate that we shall be including only one COTS item and that the part will be cylindrical or rectangular prism shaped. As a result, our standard implementation will not require this collision check function as our path planning algorithm is robust enough to plan a collision free tool path for the specified requirements. However, if we decide to scale up our project scope to include convex COTS items or multiple COTS items, this function will play a vital role in checking off the feasibility of a designed part.

The collision function was implemented as follows:

1. Obtain the convex hull of the nozzle (simplified by the use of 5 points, 4 to show the corners of the heat block and one to show the tip of the nozzle protruding from one of the sides). These vertices were generated for each and every Printer State, using the XYZR coordinates to translate and rotate the nozzle to mimic its position at every line of GCode.
2. Obtain the projection of the COTS part in the XY plane. In order to move the COTS part to the coordinate frame of the GCode, the user specified origin and rotation angles were used.

3. For each PrinterState post insertion, the Collision Check function checks if any of the vertices of the nozzle are inside the projection of the COTS part in the XY plane.
4. If a collision occurs, alert the user.

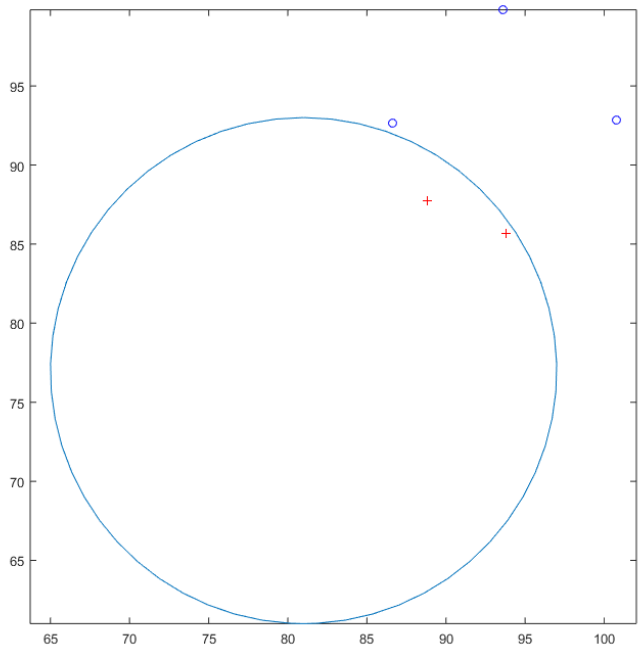


Figure 3 : Nozzle is in collision with the COTS part

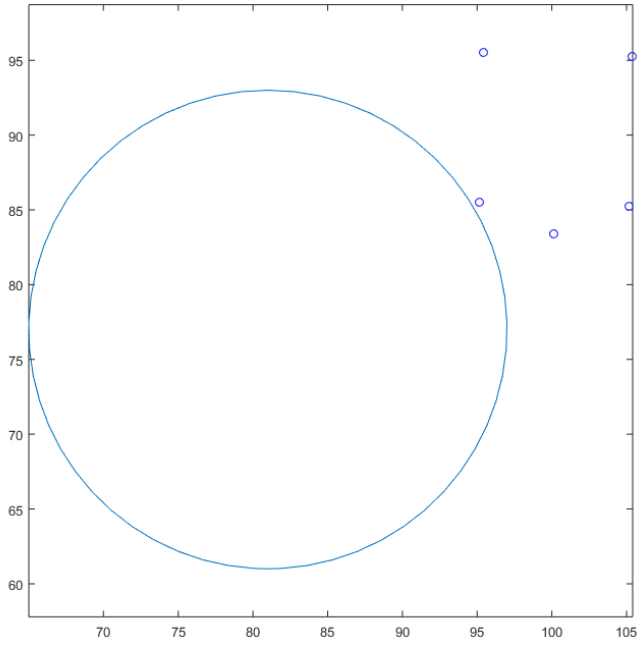


Figure 4 : Nozzle is not in collision with the COTS part

## Note:

This above case was forcibly created to display a scenario where the nozzle and COTS item are in collision by moving the COTS part into the nozzle's path. In our current implementation with decided COTS parts, the collision check function does not return any collision.

# Challenges

The biggest challenge for me this week was managing time. It turned out to be an extremely hectic week with assignments of most subjects aligning with a mid term. The team in general is waiting on the jamming issue to be fixed so that we can start our testing process.

# Teamwork

Remaining goals for Progress Review 9 were distributed as follows:

### Ishane Debbache

Ihsane worked on getting the encoder mounted onto the printer, and attempted to carry out some test prints. However, the clogging of the barrel was a constant problem. He will work on managing the heat transfer within the nozzle more efficiently.

### Nikhil Baheti

Nikhil worked on the firmware goals. He had help from Dan to find the bug that was causing the issue with the E1 port. After that, he worked on building the final version of the firmware that controls all motors.

### Dan Berman

Dan and I discussed how to trouble shoot the existing errors in the code and how to implement collision checking effectively. After some great inputs from him, I proceeded to write the code, while he focused on mounting his nozzle and testing it.

## Future Goals

As we are already generating the GCode that the printer will run, the next steps for the software team involve adding functionality to the GUI and beginning work on taking it to a web platform. As the firmware is also achieving completion in the next few weeks, our team's goal is to focus our efforts towards reaching a point where we can all test our printer whenever we have time. As a result, our primary goal for the coming PR is to be able to print around COTS items.