

ILR06

1/28/16

TEAM F: ADD_IN

Dan Berman

Astha Prasad

Ihsane Debbache

Nikhil Baheti

Individual Progress

In the fall semester we focused on developing prototypes of the highest risk portions of our project. This semester we are focusing on refining the prototypes we developed in fall into final versions and integrated all components of the system.

Since returning from winter break I have focused primarily on software development and assisting Ihsane with the design and manufacture of a new nozzle and mounting hardware for the rotary joint components. In fall semester Astha and I demonstrated that we can control Slic3r (an open source 3D printer slicing software) from MATLAB, developed a path planning algorithm to generate G-Code for the rotational axis of our printer, created visualization tools, and created a simple user interface for selecting an insertion layer and pausing the print at the appropriate height. All of these modules are currently separate, and implemented as MATLAB scripts which requires user expertise to modify. For this reason, we have devised an object oriented framework which can incorporate all of these algorithms, manage shared data between them, and provide a single interface for a user, GUI, or web application.

In the first two weeks of the semester I created a MATLAB class definition for 'ADDIN' including function prototypes for all the key methods needed to implement the algorithms described above. I also developed additional classes for internal data structures to store the state of the printer throughout time and track the location and geometry of COTS items. Using the ADDIN class definition Astha and I were then able to implement all of the algorithms developed in the fall semester within the new ADDIN class. A summary of the properties and methods of the ADDIN class is shown in Figure 1.

Property Summary

COTSArr	Array of all COTS items
PlotArr	Array of [X;Y;Z;R] (absolute) points to plot
StateArr	Array of all printer states
arrowLength	
flag_PlotArrValid	
flag_rValid	Flags
plotInterval	Plotting Control

Method Summary

	addCOTSItem	
	computeRAxis	
	displayGCodeBasic	
	displayGCodeTemperature	TODO
	getCOTSLocs	
protected	getVal	re = [prefix, '-?d?.?d*'];
	insertPause	TODO
protected	parseCmd	Define each gcode command with regular expression
	plotCOTSLocations	
protected	readGCodeFile	Find the number of lines of gcode, 'L' (Not being used at the moment)
	writeGCodeFile	

Figure 1: ADDIN Class properties and methods

In addition to software development I have also been helping Ihsane with mechanical design. I've advised him on good practices for developing the torque transmission interface between the hollow shaft stepper motor and the extruder nozzle. In addition, I helped him to select appropriate materials which I will be machining per his design next week.

I've also developed a new design for our extruder nozzle which will hopefully help us to reduce the overall footprint of the nozzle. The primary reason our original nozzle design was so large was because we incorporated the stock printer's cartridge heater. This approximately $\frac{1}{4}$ in diameter x 1 in long heater added significant volume to the nozzle's heat block. In my new design I have forgone using a prebuilt heater and instead sourced a spool of nichrome wire from which we can hopefully wind our own heater coil directly onto the extrusion nozzle. Based on the wire resistance, heater geometry, and our 19v power supply I have calculated the heater to produce at most 50 Watts, comparable to the stock 40W heater. Using SolidWorks I performed a simple thermal analysis which indicates the nozzle operating in the correct temperature regime, however it is unknown how the nozzle will actually perform during printing where flowing plastic may induce significant thermal transients. A CAD model and thermal analysis of the nozzle design is shown in Figures Figure 2 and Figure 3

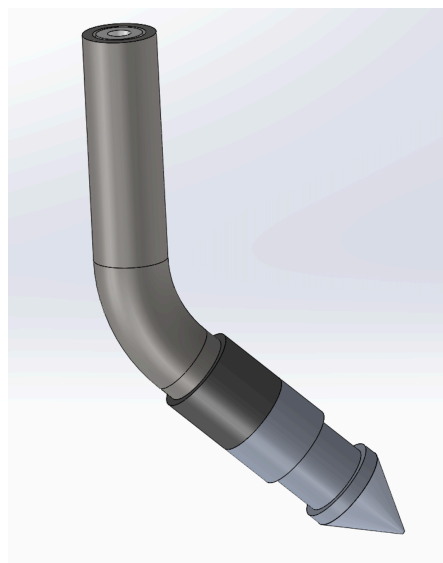


Figure 2: New low-profile nozzle design

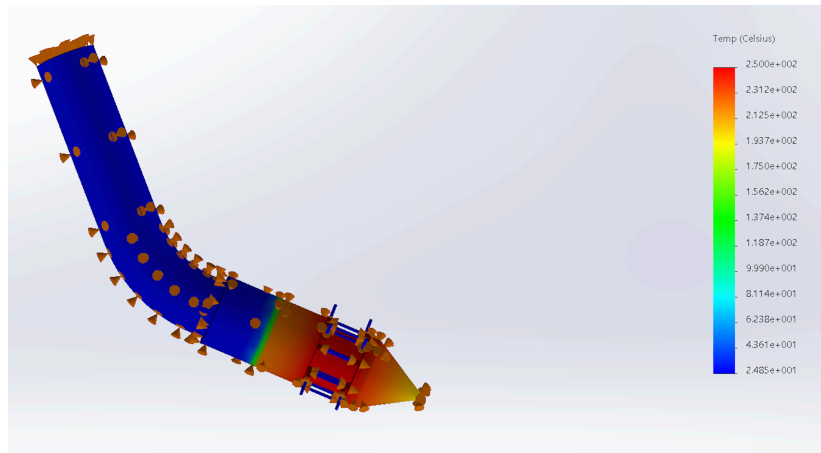


Figure 3: Thermal analysis of new nozzle design. The nozzle is simulated to be in free convection, ($T_{\text{ambient}} = 25\text{C}$) with 50 Watts of heat applied at the center of the heating coil

Challenges

Since we have only just begun the semester no major challenges have yet presented themselves, however there are some minor shortcomings which we are working to solve.

At the end of the fall semester we made the decision to purchase a second printer to accelerate testing. We have a second hollow shaft stepper motor, but did not procure a second slip ring. Thus we are working with the slip ring manufacturer to procure a second one so we can assemble the rotary joint on the new printer. The manufacturer claims to have the slip rings in stock, but last semester it took multiple weeks before it was received.

We are currently aiming to have the full system integrated (rotary joint operational, software functional, and firmware modifications complete) by February 3rd. On Tuesday it was discovered that an electrical malfunction on the printer's controller board may prevent micro stepping of the rotary joint. This would limit the accuracy and potentially prevent smooth rotation of the rotary joint. It appears that this issue is a hardware failure which will be solved by swapping in our spare controller board but this has not yet been tested.

Finally, we have encountered difficulty in choosing a suitable standard to for our project class presentation. Despite the proliferation of 3D printers, regulators have not yet produced any standards pertaining to additive manufacturing or 3D printing. UL however has developed a website addressing additive manufacturing and list a variety of standards which may be applicable depending on the printer's application (home, office, medical, etc). Ultimately, we have chosen to use IEC 62368 which covers traditional printers and low voltage business equipment. For the purposes of the class exercise this will suffice, however I was hoping to find a standard which is more specific to 3D printing.

Teamwork

Astha Prasad and I worked together to develop the software architecture and begin populating the methods. She constructed the functions for G-Code reading and writing which interface with the PrinterState class I created for storing time dependent printer state information.

Nikhil Baheti worked on developing the custom firmware for the printer. He has gained a great understanding of the original firmware's architecture (no small feat!) and successfully implemented the same axis synchronization algorithm on the rotary axis used for the printer's original X Y and Z axes. He also uncovered a potential hardware failure in the controller board which will require further investigation in the coming week.

Ihsane Debbache has worked to devise the mounting hardware for the hollow shaft stepper motor and slip ring, as well as the necessary shafts and connectors to transmit torque from the motor to the nozzle. He has also created a CAD design for a new revision of our modified nozzle which is less risky than the completely new nozzle design I created. He has also been responsible for purchasing, including all of the parts for both new nozzle designs, the slip ring, and a second 3D printer.

Future Plans

In the coming weeks my primary focus will be developing a 'demo version' of the software allowing all team members to easily create, modify, and visualize G-Code specific to our printer. Using the framework Astha and I have begun, I will be creating a simple GUI (using MATLAB's GUIDE software). The interface between the GUI and the ADDIN class will form the API which will later be used to create a web-based version of our software. A mock up of the GUI is shown in Figure 4.

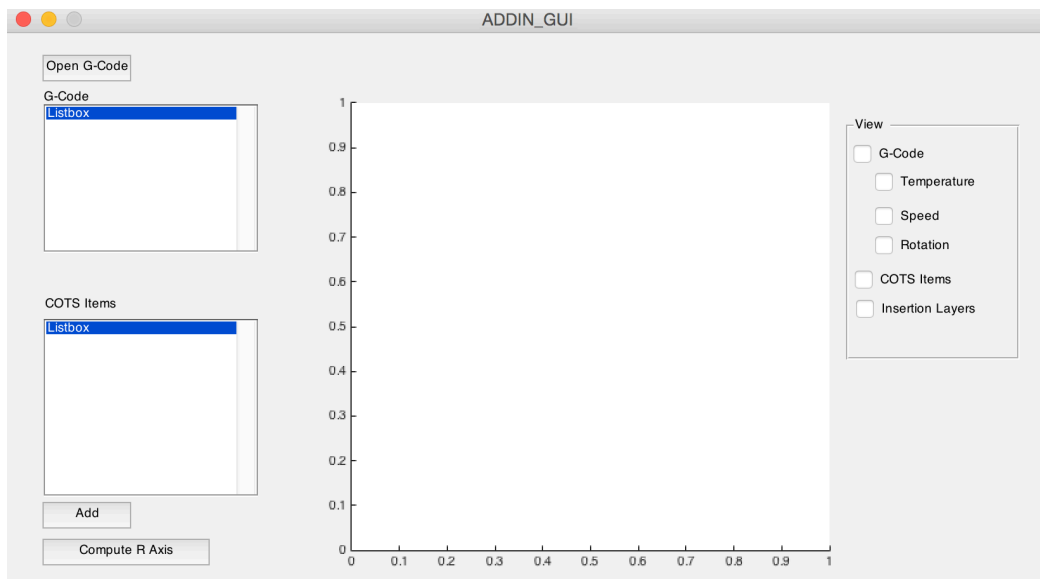


Figure 4: ADDIN GUI Mockup

Since Ihsane does not have shop access, I will also be responsible for manufacturing both nozzle designs and the mounting hardware for rotary joint. I expect there will be significant hurdles encountered with my new nozzle design and thus will likely need to spend significant time testing, tuning, and iterating the design. I am particularly interested to see if the PID loop used to control the heater is able to operate quickly enough to compensate for the reduced thermal mass as compared with the current heater design.