5-2016

Dynamic Extruder Control for Polymer Printing in Big Area Additive Manufacturing

Alex Christopher Roschli
University of Tennessee - Knoxville, aroschl1@vols.utk.edu

To the Graduate Council:

I am submitting herewith a thesis written by Alex Christopher Roschli entitled "Dynamic Extruder Control for Polymer Printing in Big Area Additive Manufacturing." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Electrical Engineering.

Benjamin J. Blalock, Major Professor

We have read this thesis and recommend its acceptance:

Syed K. Islam, Brett G. Compton

Accepted for the Council:
Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Dynamic Extruder Control for Polymer Printing in Big Area Additive Manufacturing

A Thesis Presented for the
Master of Science
Degree
The University of Tennessee, Knoxville

Alex Christopher Roschli

May 2016

To my parents and brothers who have provided me tremendous support throughout my education.

ACKNOWLEDGEMENTS

I would like to thank my advisor Dr. Benjamin Blalock for all his help with my research. He has provided me countless hours of support and encouragement to help make this possible. Without his encouragement, I may not have continued my education into graduate school and stopped with just a Bachelor’s Degree.

I would like to thank my committee members Dr. Islam and Dr. Compton for the time they spent reviewing and editing my thesis.

I would like to thank Dr. Brian Post for mentoring me in control systems and guiding me through my research. Without Dr. Post, this research could not have happened.

I would like to thank the Department of Energy for their support of the Manufacturing Demonstration Facility where all of this research took place.

Lastly, I would like to thank my coworkers at ORNL who have been very gracious in their support of my research. Thank you Lonnie Love, Pete Lloyd, Randy Lind, Brad Richardson, Greg Dreifus, John Rowe, Matt Sallas, John Lindahl, Vlastimil Kunc, and Fletcher Blue.

ABSTRACT

Big Area Additive Manufacturing (BAAM) is 3D Printing on a large scale and can be used to create structures on the scale of cars and houses. Scaling up 3D printing to BAAM size meant fundamentally altering the traditional fused deposition modeling process by switching from a filament to a pellet material feed. This meant switching from a stepper motor extruder to a servo driven screw extruder. While increasing the throughput of the system, this new extruder increases the overall complexity. Effective control of the system is paramount to the success of BAAM enabling it to effectively scale in speed in the way it scales in build size. If the extruder can be quickly and accurately controlled, then a dimensionally accurate part can be printed. This paper will focus on the control of a single screw polymer extruder with zone controlled heating. State space control methods will be applied to shape both acceleration and deceleration of the spindle during extrusion so that a consistent bead can be produced at variable speeds.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE
INTRODUCTION

3D Printing

Additive Manufacturing, more commonly known as 3D Printing, is a manufacturing process that involves successively adding material to grow a part, as opposed to traditional, subtractive manufacturing processes that take away material, like cutting and machining. Additive Manufacturing allows for construction of a 3-Dimensional part using 2-Dimensional layers. Each of these layers is a thin cross-section of the actual part, making the additive manufactured part an approximation of the actual part. A benefit of 3D Printing is that complexity is free. In contrast to traditional manufacturing processes where more features lead to more machining operations and therefore cost and time, the effort required to manufacture a part is directly proportional to the amount of material required and the deposition rate. Nearly anything that can be designed can be printed. This allows for new and unique structures to be built that could not be made with traditional manufacturing. 3D Printing has been around since the 1980s, but has typically been limited by small build spaces, slow printing speeds, and high material costs.

Big Area Additive Manufacturing

Big Area Additive Manufacturing, more commonly known as BAAM, is 3D Printing amplified. The original prototype BAAM gantry system, still in use today, had a build space of 8’x8’x8’ and could only output 10lbs/hour of material (still a factor of 100 larger than most 3D Printers). Since then, BAAM has grown in

every way such that it is significantly bigger and faster than traditional desktop 3D Printing. Table 1.1 shows a direct comparison between desktop 3D printing and BAAM. The speed is almost three times faster, volume over three thousand times larger, and flow rate over two thousand times greater.

Table 1.1. Desktop vs BAAM 3D Printing

	Desktop 3D Printer	**BAAM**
Build Length	~8in	240in
Build Width	~8in	96in
Build Height	~8in	72in
Build Volume (in^3)	512	1658880
Print Speed	~4in/s	~11in/s
Mass Flowrate	1lb/day	100lbs/hour

In addition to being bigger and faster, BAAM is more diverse in the materials that it can print with. This is because BAAM uses a screw based extruder that takes in pellets as a feedstock as opposed to desktop printers which use a stepper motor to feed in filament. The screw-based extruder allows for printing with composite materials, such as carbon fiber reinforced ABS (acrylonitrile butadiene styrene), as well as high temperature materials, such as PPS (polyphenylene sulfide). Using injection molding pellets also reduces the cost. Pellets average about one to five dollars per pound while filament is about sixty to one hundred and fifty dollars a pound. BAAM requires purchases of pellets in 1000lb Gaylords while traditional desktop printers use filament from 1kg spools.

Machine Setup

The current machine hardware is based on a collaboration between Oak Ridge National Laboratory and Cincinnati Incorporated with both sharing ownership of the machine components. The machine, nicknamed Bertha, offers a build volume of 8’x20’x6’. Bertha is currently installed in the Manufacturing Demonstration Facility at ORNL, which is located in the Hardin Valley area of Knoxville, Tennessee.

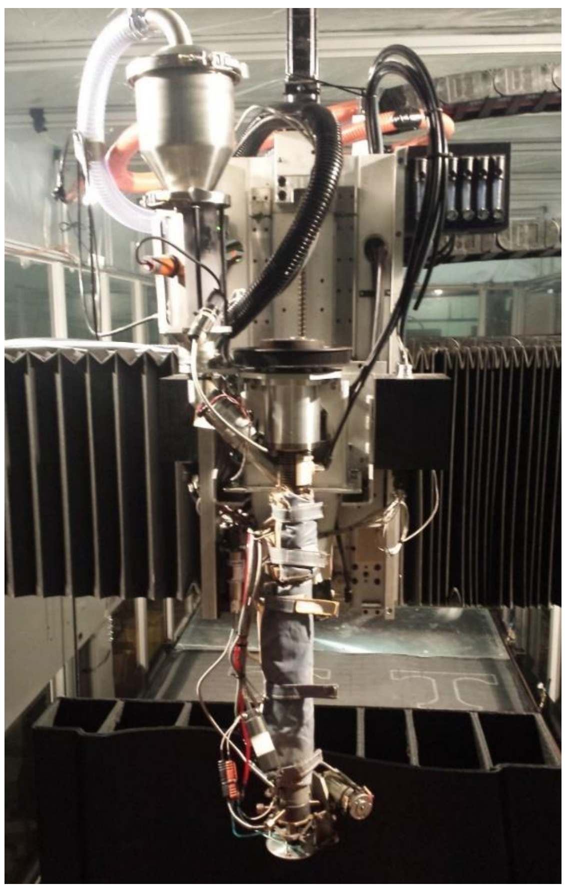


Figure 1.1. Current Extruder on Bertha

Extruder

The extruder (shown in Figure 1.1) for this research is a single screw extruder with independent zone controlled heating. Plastic micro pellets are first dried in a desiccant hopper system then fed to the extruder with a vacuum line. The pellets feed in at the top of the screw/barrel of the extruder and get pushed down through the heat zones until they are fully melted for extrusion. The screw is belt driven by a 3.5HP servo motor (the belt is visible at the top of the extruder in Figure 1.1). The screw is custom made from a case-hardened alloy. Heating for the screw chamber is done in five zones, with the fifth zone being the tip. The heating elements are resistive band heaters that heat the outside of the extruder barrel.

Thermocouples are used to measure the temperatures at each zone as well as a thermocouple placed in the melt stream to measure the extrusion temperature. A pressure sensor is also present just above the tip so that backpressure in the barrel can be measured during extrusion. A set screw is located near the end of the barrel that can be adjusted to change the backpressure.

Attached around the nozzle is a device developed by ORNL known as the tamper. This device oscillates a platen up and down during printing, striking the top of the molten plastic. The tamper helps increase z-strength of printed parts by increasing the surface area of the layers in contact and it also helps keep all material in-plane. Often times during printing, material will build up or overfill and the tamper helps maintain a flat top surface for the next layer to be printed over.

Gantry System

The extruder is mounted on a four-axis gantry system. This system is what moves the extruder to trace out each layer of the part. The gantry used for this research was developed by Cincinnati Incorporated for laser cutting. The gantry uses two linear motors in a master slave configuration to control movement on

the X-axis, one linear motor for the Y-axis, a ball screw for movement in the Z-axis, and six ball screws connected with line shafts to a single servomotor for moving a table workspace along the W-axis. The machine is capable of traveling at speeds greater than ten thousand inches per minute in the X and Y directions.

Oak Ridge National Laboratory took possession of the gantry, without the laser cutting hardware, and mounted an extruder in place of the laser cutter. A hopper-style dryer for materials handling has also been added to the system.

The Extrusion Problem

When scaling up 3D printing from typical desktop size to BAAM size, persistent problems have been observed in the extrusion. The transition from a filament fed extruder run with a stepper motor to a pellet fed servo driven screw extruder has greatly increased the system complexity and resulted in significant control issues. Pellet flow is not nearly as consistent as filament feeding and as a result, is harder to control. Furthermore, the behavior of the polymer is highly nonlinear and very sensitive to both temperature and shear rate. Often the errors are exacerbated in the dynamic regimes, when the extruder is started and stopped. With a stepper motor filament extruder, the stepper can reverse directions to retract the filament up into the barrel to stop extrusion. With a screw extruder, the barrel fills with molten plastic which behaves as a non-Newtonian fluid. Spinning the screw the opposite direction to stop the flow of plastic is not nearly as effective as retracting the filament in a desktop system. Another problem with reversing the direction of the screw is that molten plastic could be fed into cool areas of the barrel where plastic melt is not intended. This retracted plastic will cool and harden causing a disturbance of flow or blockage in the extruder.

In addition to the problem of not being able to stop plastic flow, a screw based extruder feeding pellets cannot maintain bead width and thickness as

consistently. This becomes much more of a problem when the extruder accelerates or decelerates, such as starting/stopping a print move or traversing around a corner. This thesis will focus on tuning the extruder for a more consistent bead width during printing in the dynamic regimes, i.e. periods of rapid acceleration and deceleration.

CHAPTER TWO
BACKGROUND

This chapter will focus on the background information of 3D Printing and polymer extrusion. The first section will offer a brief history of 3D Printing and where Big Area Additive Manufacturing fits in. The second section will focus on polymer extrusion, specifically for screw based extruders.

Brief History of Polymer 3D Printing

3D Printing began in the early 1980s with the invention of stereolithography (SL). Charles Hull is credited with being the inventor of this new process that uses a laser to cure a plastic resin into a 3D Printed part [1]. Hull received a patent for this technology in 1986 and subsequently brought the technology to market with his new company 3D Systems. Following the invention of SL by Hull, was the invention of Selective Laser Sintering (SLS) in 1989 by Carl Deckard. The patent for SLS was licensed to DTM Inc a company that was bought by 3D Systems. SLS is very similar to SL; however, instead of forming a part from a liquid resin, SLS uses a powdered material.

While 3D Systems was working on SL and SLS technologies, a company called Stratasys, led by Scott Crump, began work on Fused Deposition Modeling (FDM). Stratasys received a patent for this technology in 1992. FDM is an extrusion-based approach to making thermoplastic parts. FDM works by adding plastic in layers one right after another. This is typically done by feeding in a plastic filament that is melted in a hot end and extruded through a nozzle.

Big Area Additive Manufacturing (BAAM) is an FDM based process that began at Oak Ridge National Laboratory in 2013. Every aspect of traditional FDM machines was scaled up to allow for the creation of BAAM which is capable of

making parts as large as an entire car. BAAM uses pellets as feedstock in place of the filament used by traditional FDM machines. This exchange not only saves on material cost, but allows for the use of a screw based extruder where volumetric flow rates can exceed 2400in^3/hr. The current generation of BAAM has a print volume of 960ft^3, while the average consumer grade FDM machine is just 0.3ft^3.

All three of the mentioned technologies: SL, SLS, and FDM are all still in use and are the most popular methods for polymer additive manufacturing. Of the three, FDM is the most common because it is the least expensive machine to produce, and thermoplastic materials are readily available and offer good mechanical properties. As a result of this low cost, small scale FDM machines are very common and can be purchased locally at stores such as Walmart. This is also the reason that FDM was scaled to create BAAM. A large machine, such as the one in this research, is cost effective using FDM but is not as feasible using the other two technologies.

Polymer Extrusion

Extruding polymer with a screw based system is not a new concept. Injection molding, invented in the 19th century, primarily uses screw based extruders to eject molten plastic to fill a mold cavity. Research on extrusion for injection molding is often focused on maintaining consistent polymer quality to ensure the molded parts are uniform. The following subsections will discuss the existing research that has gone into perfecting polymer extrusion and how these ideas can or cannot be used for additive manufacturing.

Control Designed for Injection Molding

For injection molding, an extruder is typically turned on and run at a constant speed until the mold is full, then turned off. In alternative approaches, the screw

produces a “shot” of material in a heated reservoir, which is then injected into the mold. In additive manufacturing, the extruder travels on the head of a robotic manipulator. It is not always turned on at a constant speed; it must respond to the dynamic motion of the system. The speed must be constantly changing as the gantry moving the extruder has to accelerate and decelerate to follow specific contours. In addition to changes in gantry speed, the extruder must also be able to account for different print moves. The gantry speed may remain constant, but the extruder will need to either accelerate or decelerate to fill a large or small void. This acceleration of the extruder needs to be smooth so that a consistent bead of plastic is maintained without any starvation events. While the main concern of injection molding is maintaining a constant extrusion temperature for the duration of the filling of the mold, in AM, maintaining a consistent temperature is important for uniform layer to layer bonding, but it’s not nearly as important as consistent flow because inconsistent flow could result in a broken bead, or misplaced bead, which could lead to a failed print.

Control Based on Temperature and Pressure

In [2], Previdi, Savaresi, and Panarotto developed a system for controlling a single screw plasticating extruder based purely upon temperature and pressure feedback. Their extruder had seven temperature zones. Each zone was heated individually and had its own thermocouple for temperature monitoring. Of these seven zones, four were in the nozzle and three were inside the barrel. In addition to the seven zones, an eighth thermocouple was placed in the extruded plastic about ten centimeters from the nozzle. One pressure sensor was implemented where the material mated with the metal. To control their extruder, a 0-5v signal was sent to the motor which corresponded to a 0-200rpm spindle speed. Figure 2.1 the proposed feedback control system based on temperature and pressure.

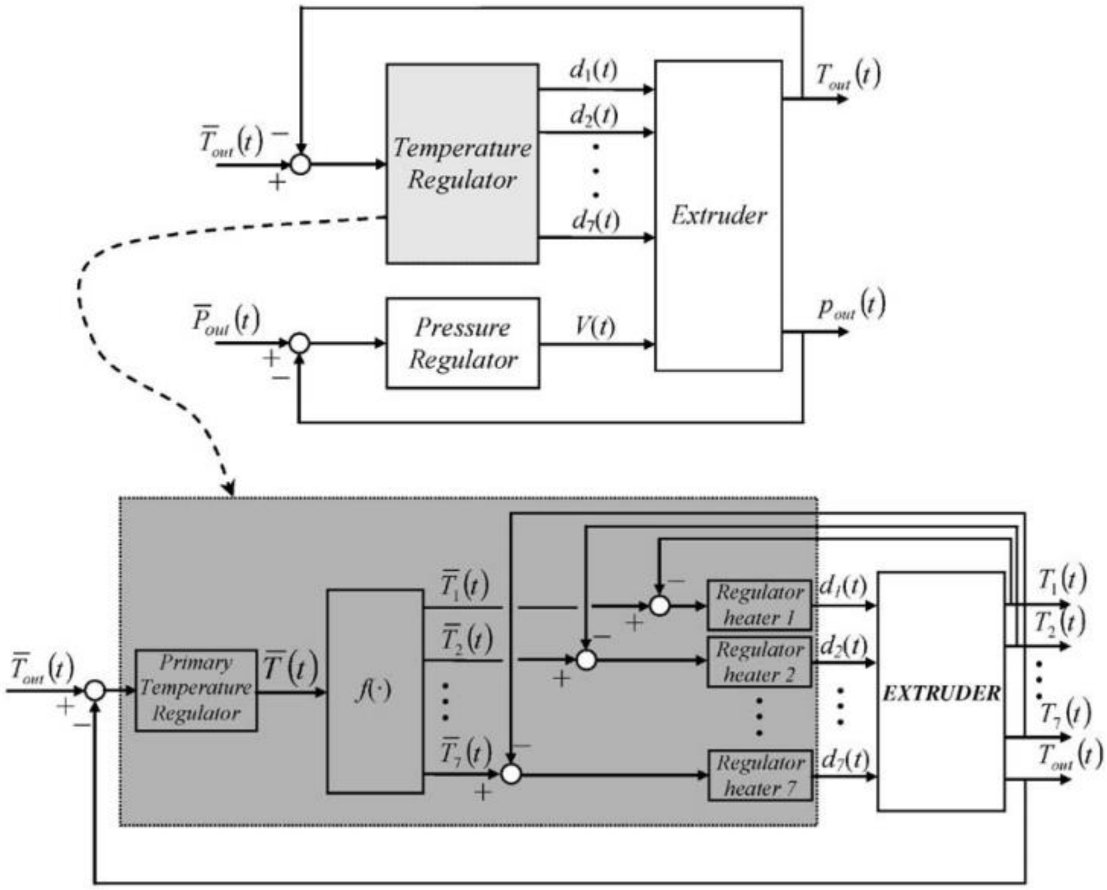


Figure 2.1. Previdi Feedback Control System

They were able to use the real-time feedback from all seven zones as well as the output temperature and pressure to maintain a consistent output temperature and pressure. A near-perfect control of output temperature and pressure is great for additive manufacturing in that it will help regulate consistency from layer to layer in a part. If every layer is printed at the exact same temperature and pressure, then the part will have a very consistent layer to layer bond strength. The ability to control the temperature of each layer to increase the layer to layer bond strength would be very helpful. In addition to

aiding the bond strength, a consistent temperature and pressure could help create a more uniform surface finish that requires little or no post-processing.

How Extrusion Pressure Affects Plastic

In [3] and [4], Costin, Taylor, and Wright demonstrate the control of plasticating extruders where they studied the effect that pressure had on the plastic output. Their goal was to study the plastic quality by changing the input composition and by varying screw speed. The plastics to be used for the research were three types of polyethylene from Union Carbide.

For their research, they used a 38mm Killion extruder with a 24:1 length-to-diameter ratio. The barrel had four PI controlled temperature zones and the die, or nozzle, had one additional heat zone. The melt temperature was measured by placing a thermocouple in the die. Pressure was measured in the melt-pumping region of the screw with a pressure transducer. Extruded plastic was water cooled and then air cooled before feeding into a take-up mechanism. Below is a schematic illustrating this extruder setup [3].

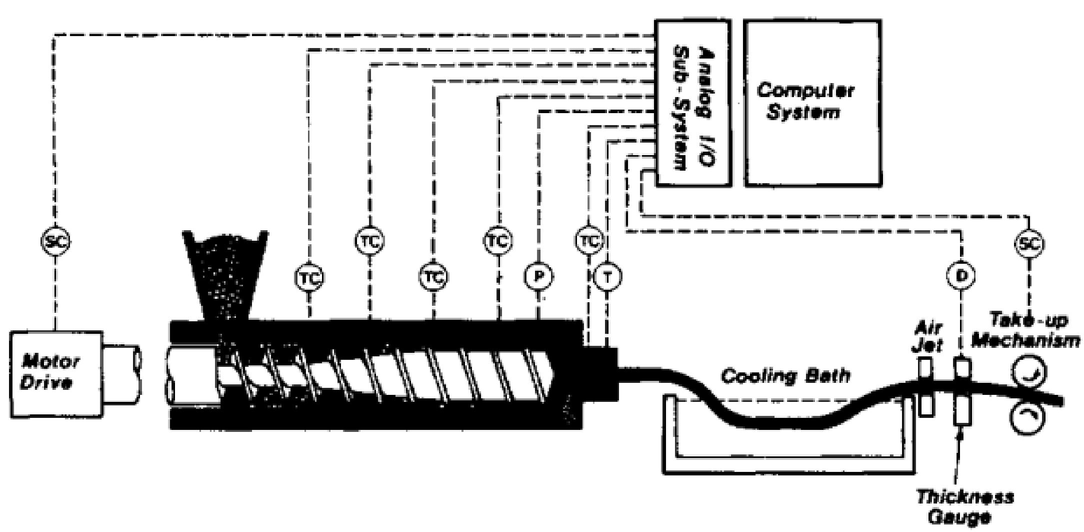


Figure 2.2. Costin Extruder Schematic

In their research, they were able to create transfer functions to correlate the relation of screw speed and pressure. Equations 1 and 2 show transfer functions for temperature and pressure for a screw speed change of 6.5r/min. In addition to these transfer functions, they created a time-series model of the extrusion to get an accurate estimate of the noise (Equation 3). Using these models, they were able to successfully regulate the pressure using a digital PI algorithm, even with varying polymer quality. The ability to regulate pressure of the extruder in the face of changing polymer quality is very important for additive manufacturing. A consistent pressure can help keep a constant flow rate meaning that layer after layer, the extruder will perform the same and result in a uniform outer part surface. When printing large parts, as is often the case with BAAM, it is possible to go through an entire batch of material and start in on another mid-print. No two batches of material are alike, so control of the extruder needs to be implemented to adjust for this change in flowrate.

$$P(s) = \frac{76.}{0.25s + 1} U(s) \quad (1)$$

$$T(s) = \frac{0.19}{15s + 1} U(s) \quad (2)$$

$$\nabla_s P_t = \frac{\omega_0(1 - \omega_x z^{-x} - \omega_{x+1} z^{-x-1})}{(1 - \delta z^{-1})} U_{t-1} + \frac{(1 - \theta_1 z^{-1} - \theta_2 z^{-2})}{(1 - \varphi_1 z^{-1} - \varphi_2 z^{-2})} a_t \quad (3)$$

where the parameters and their 95 percent confidence bands are:

ω_o = 70.7 ± 3.4 kPa/motor power
ω_s = 0.711 ± 0.064
ω_{s+1} = 0.521 ± 0.065
δ = 0.281 ± 0.039
θ_1 = 1.134 ± 0.154
θ_2 = −0.295 ± 0.163
φ_1 = 1.722 ± 0.079
φ_2 = −0.869 ± 0.079
∇_s = $1 - z^{-s}$
s = 60/TACH$_t$/T_s (TACH$_t$ is in r/min)
T_s = sampling time = 0.5 s.
x = integer truncation of s

Smith, Tortorelli, and Tucker wrote two papers ([5] and [6]) discussing the control system of polymer extrusion. Part of their research involves changing the die size and thickness to minimize pressure fluctuation and maintain a consistent exit speed of the polymer. Both of these are important for additive manufacturing. Changes in pressure during extrusion can be directly correlated to bead quality. Too low of pressure means lower flow rate and a non-uniform bead. High pressure is dangerous to the system and can cause the polymer to degrade, or failure of the equipment. Another aspect of their research involved maintaining a consistent residence time. This is also important for additive manufacturing because residence time (the amount of time the polymer is inside the heated extruder barrel) affects polymer quality. If the residence time is too high, the polymer degrades. If it is too low, the polymer doesn't fully melt and the resulting output could have unmelted pellets, resulting in poor layer to layer adhesion.

Extruder Step Response

Tadmor, Lipshitz, and Lavie developed a dynamic model for a plasticating extruder [7]. Their model encompasses the entire process of polymer extrusion, going from pellets in a hopper to molten plastic ejecting from the extruder. This is important because a similar system is used by BAAM, where the pellets are stored in a dryer/hopper and fed via a vacuum line to the extruder where melting and extrusion occurs. The goal of their research was to study the transient response of the extruder. In their research, they found that pressure gradients and melt rates can be calculated with steady state models. They also found that adding a valve at the die to control flow rate was a bad idea because of the pressure spikes it can cause. Equation 4 gives the equation for approximating pressure drop at the die.

$$G = K_D \rho_m \frac{\Delta P_D}{\mu_D} \qquad (4)$$

μ_D = mean viscosity at die

G = flow rate

ΔP_D = pressure drop across die

ρ_m = density of the melt

K_D = constant dependent on die geometry

CHAPTER THREE

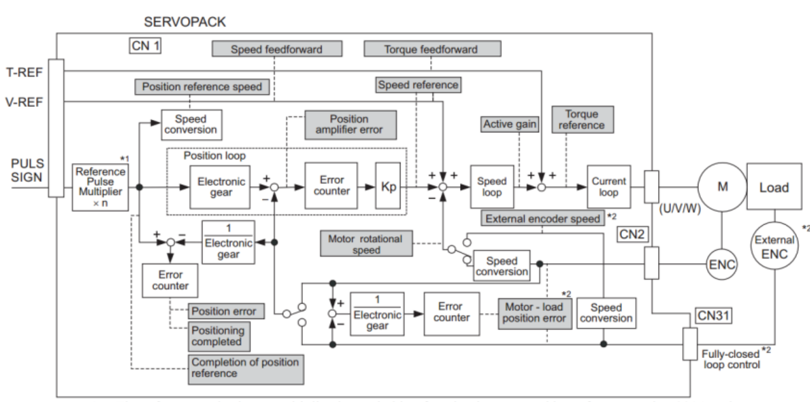
METHODS FOR TESTING

Initial Control System

The servo motor used by the extruder on Bertha is a Yaskawa Sigma-5 Series. The initial control system for this servo motor was developed by Yaskawa Electric Corporation. This generic control was designed to be a best fit for most use cases and did not work effectively to control the extruder during 3D Printing.

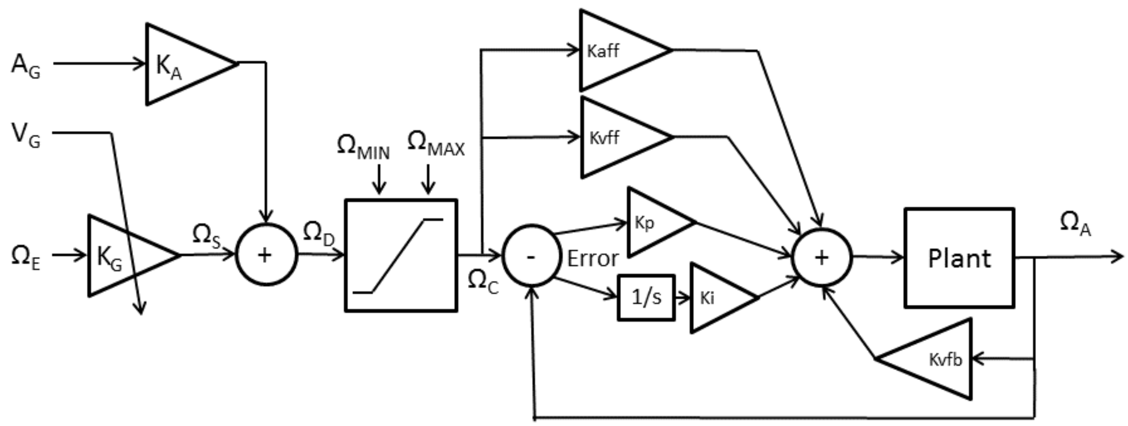
Block Diagram

The control diagram for this servo motor has been mapped out by Yaskawa and can be seen in Figure 3.1 [8]. All shaded boxes represent analog output signals that can be monitored while the servo motor is running, or during extrusion of polymer. Figure 3.2 shows a block diagram designed for this thesis that shows how all control values factor in.



*1. The reference pulse input multiplication switching function is supported by software version 001A or later.
*2. Available when the fully-closed loop control is being used.

Figure 3.1. Yaskawa Block Diagram



A_G – Gantry Acceleration
V_G – Gantry Velocity
Ω_E – Gcode Programed Extruder Feedrate
Ω_S – Servoed Screw Speed
K_G – Override Gain (Based on Gantry Velocity)
K_A – Acceleration Feedforward Gain
K_p – Proportional Gain
K_i – Integral Gain

Ω_{MIN} – Minimum Screw Speed = 0 RPM
Ω_{MAX} - Maximum Screw Speed = 400 RPM
Ω_D – Desired Screw Speed
Ω_C – Commanded Screw Speed
Ω_A - Actual Screw Speed
Kaff – Acceleration Feedforward
Kvff – Velocity Feedforward
Kvfb – Velocity Feedback

Figure 3.2. New Control Block Diagram

Default Values

When BAAM started with this new extruder setup (circa May 2015), the extruder servo motor input commands were being shaped with default values (see Figure 3.1 for a list of all variables and their values). These default values were provided by the programmer/distributor of the Programmable Logic Controller (PLC) used by Cincinnati Incorporated for their Laser Cutters and BAAM. This company is Delta Tau Data Systems. With these default settings, the extruder was not able to accurately print parts at print speeds of ten inches per second with spindle speeds up to four hundred revolutions per minute on a 0.3” nozzle because the produced bead width was inconsistent.

Table 3.1. Default Control Values

	Default Value
Kp (proportional gain)	0.01
Ki (integral gain)	0
Kvfb (velocity feedback)	0.6
Kvff (velocity feedforward)	0.285
Kaff (acceleration feedforward)	1

As seen in Table 3.1, there are five main variables to be controlled from the firmware to change how the extruder operates. Initially, the integral gain was turned off, or set to zero. Integral gain allows the previous errors to effect the future control by summing the error and multiplying it by the gain value. When it is off, the controller does not take into account the previous error.

Inconsistent Bead Widths

The extrusions with the default settings had many issues, the most significant was an inconsistent bead width. The default settings did not allow the extruder to quickly and accurately accelerate and decelerate as needed which caused significant fluctuations in bead width, or a non-uniform bead. The gantry could change speeds much more dynamically than the extruder, which caused print failures. If the extruder can’t accelerate fast enough, then not enough plastic will be extruded for the bead to adhere to the previous layer, and it will be removed by the motion of the extruder and the bead tension. If the extruder cannot slow down fast enough, such as when going around a corner, then the extruder will deposit too much plastic causing issues with overfilling.

At starting points, the extruder didn’t accelerate nearly as quickly as the gantry. This caused thin sections or beads that didn’t stick and would be dragged around by the extruder. At stopping points, the extruder couldn’t decelerate fast enough and would leave a large amount of extruded material and the end of a

bead. Both situations are not desirable. Furthermore, if the extruder were starved of material, the control would destabilize the system resulting in violent vibration of the extruder.

All of these problems compounded, resulting several failed (>$1000) builds. Overfilled sections caused the outer beads to be pushed out resulting in a loss of dimensional accuracy. Thin sections could fall off the side of the part meaning that the next layer would also fail to adhere. Infill sections would typically be overfilled risking machine damage if any high spots got in the way of the future trajectories causing collisions between the gantry and the part. These failed prints motivate this research into effective tuning of control systems for AM polymer extrusion systems. Since these systems operate with no process or supervisory feedback, it is imperative that an appropriate amount of plastic is extruded, at the correction locations during all dynamic printing regimes.

Plastic Flow

The raw material that feeds into the extruder is solid plastic pellets. For this research, the material is Carbon Fiber Reinforced ABS. Where the composite material is twenty percent carbon fiber by weight. These pellets feed in at approximately 145°F and get extruded at approximately 480°F. Carbon Fiber ABS is an amorphous plastic, and the resulting molten plastic is a non-Newtonian fluid. This means that the viscosity depends primarily upon the shear rate.

If the molten plastic was a Newtonian Fluid, it could easily be modeled by a linear system where the viscosity is constant at a given temperature regardless of the speed of the screw (which is proportional to shear rate). This is because a Newtonian fluid's relationship between stress and strain is linear. Because it's non-Newtonian, a more complex control model must be developed and dialed in for the specific plastic being extruded.

Control Values

This research will be focused on tuning the control values available in the PLC. These controls can be divided into three types. These types are the PID Control, feedback, and feedforward. Each of these terms will be explained below.

PID Control

PID Control is a style of feedback control that attempts to minimize the error by adjusting the three variables “P”, “I” and “D”. Proportional gain is the “P” part of PID Control, with the “I” term being integral, and the “D” being derivative. The proportional gain is based on the present error value. Integral gain is for past values, and derivate gain is for future values. The proportional gain contributes most of the output state change in comparison to the integral and derivative gains. Initially, BAAM only used the proportional gain and ignored the integral gain, thus was just a P Controller. The PLC does allow for an integral term to be added, making it into a PI Controller.

A higher proportional gain will lead to faster acceleration and deceleration, but could cause the system to be unstable. An unstable system will oscillate above and below the commanded value. For BAAM, this means that extruder will be constantly accelerating and decelerating. If this happens, the bead width will not remain consistent and will continually change between being too thick and being too thin, both of which are bad outcomes for the produced parts.

In contrast, a small proportional gain leads to slower response time, or slower acceleration and deceleration times. This is great if the error is small, but if the error is large, the system could be slow to reach the commanded value. If the system doesn’t reach the commanded value quickly, then there is risk of the bead not reaching steady state as it slowly gets thicker as the machine accelerates and then slowly gets thinner as the machine decelerates.

Because the integral gain term is time based, it depends on past error values, the duration of the error is an important factor. The integration of this error effectively sums the instantaneous error over time and then multiplies this error by the integral gain term. This means that a large accumulated error can cause overshoot. Overshoot occurs when the actual value exceeds the commanded value. With BAAM, a large overshoot means over extrusion or under extrusion. In the case of overshoot while accelerating, the extruder spindle will be spinning faster than commanded and dispense excess plastic. If this happens when exiting a turn, the bead could be very thick at the exit and then neck down as the feedback starts to minimize the error. If overshoot happens while decelerating, the extruder could shut off too early and plastic would stop flowing before the gantry stops moving. This will result in a gap between the beginning and end point of a bead.

Feedback

Feedback is the use of the output state to manipulate the input state to achieve a desired result. Without feedback, the system would be an open-loop controller meaning that a measure of the system output is not used by the controller [9]. In the control of BAAM there is only one feedback term to focus on, velocity feedback. This means the actual machine velocity is fed back into the controller as an input to shape the commanded velocity to try and get the actual velocity to match the commanded velocity.

Feedforward

Feedforward is manipulating the output state using the input state that has not been fed through the controller. This means that the output state will be modified in a pre-defined way based on a prediction of how the controller will shape the input signal into the output signal, i.e. the control signal is not influenced by a measurement of the system output. With feedforward control, early warnings can be sent to the control system [9]. However, this also means that errors in

command input don't get corrected by the controller and directly affect the output. BAAM control has feedforward terms for acceleration and velocity. Velocity feedforward is good for decreasing time for short moves (command response time), but can cause overshoot. Acceleration feedforward can compensate for the overshoot of velocity feedforward and also allows for high loop gains. Effective command shaping reduces the effort required by reactive controllers allowing for tighter control through increased feedback gains.

Extruder Dynamics Simulation

For the purpose of simulating the PI Controller used for the extruder, a Matlab model was created. This model is based on the actual screw used on the BAAM machine and accounts for the damping that occurs from the plastic in the barrel. The damping factor can be adjusted to demonstrate different plastics which have different viscosities with a higher viscosity correlating to a higher damping factor.

For this Matlab model, several transfer functions were used. The first, is the simple PI controller transfer function seen in Equation 5 [9]. The next is the transfer function of the plant which accounts for the moment of inertia of the screw and the damping factor, Equation 6. Equation 6 also includes Equation 7, the moment of inertia for a closed cylinder. These equations together give the open loop transfer function seen in Equation 8. From this transfer function, the poles and zeros can be plotted as seen in Figure 3.3. Finally, the closed loop transfer function is shown in Equation 9. The closed loop transfer function shows the effect of a step response, Figure 3.4.

A high value of Kp causes the step to reach max value quickly while a low value causes the step to reach max value much slower and may also cause oscillations, Figure 3.5. A value of Ki that is too high will cause significant

overshoot and oscillation, as seen in Figure 3.6. The effects of low and high damping can be seen in Figure 3.7.

$$PI\ Transfer\ Function = K_p + \frac{K_i}{s} \quad (5)$$

$$Plant\ Transfer\ Function = \frac{1}{Is + b} \qquad I = \frac{mr^2}{2} \quad (6,7)$$

$$Open\ Loop\ TF = \frac{sK_p + K_i}{s(Is + b)} \quad (8)$$

$$Closed\ Loop\ TF = \frac{sK_p + K_i}{Is^2 + s(b + K_p) + K_i} \quad (9)$$

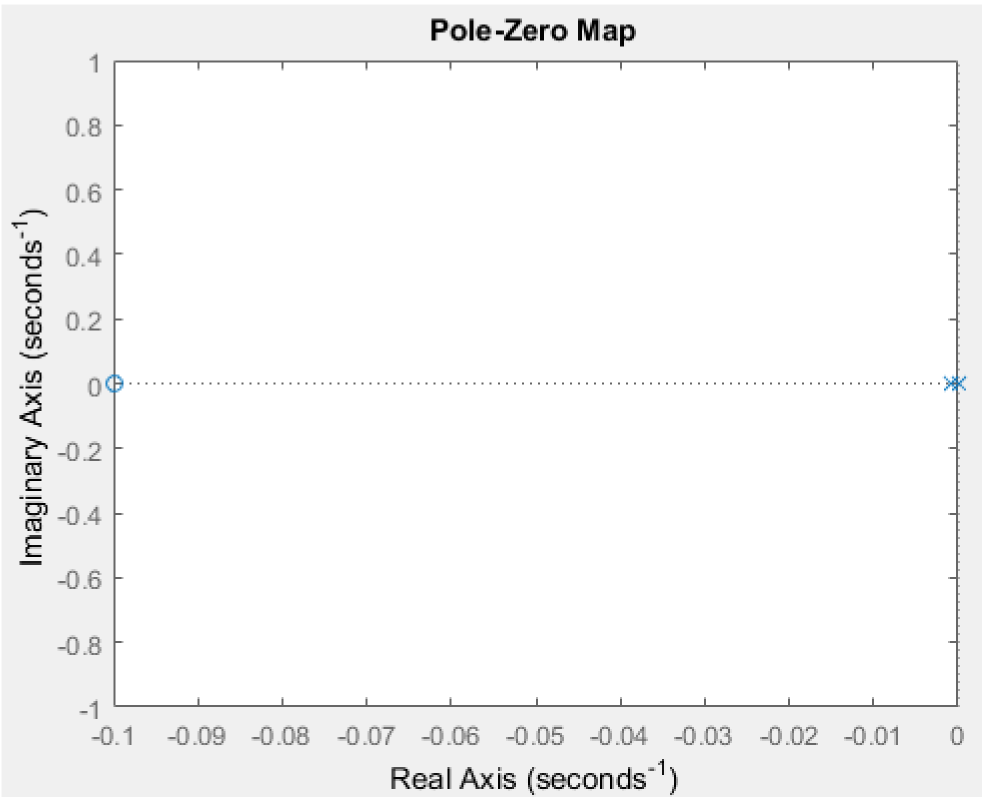


Figure 3.3. Poles and Zero of Open Loop Transfer Function

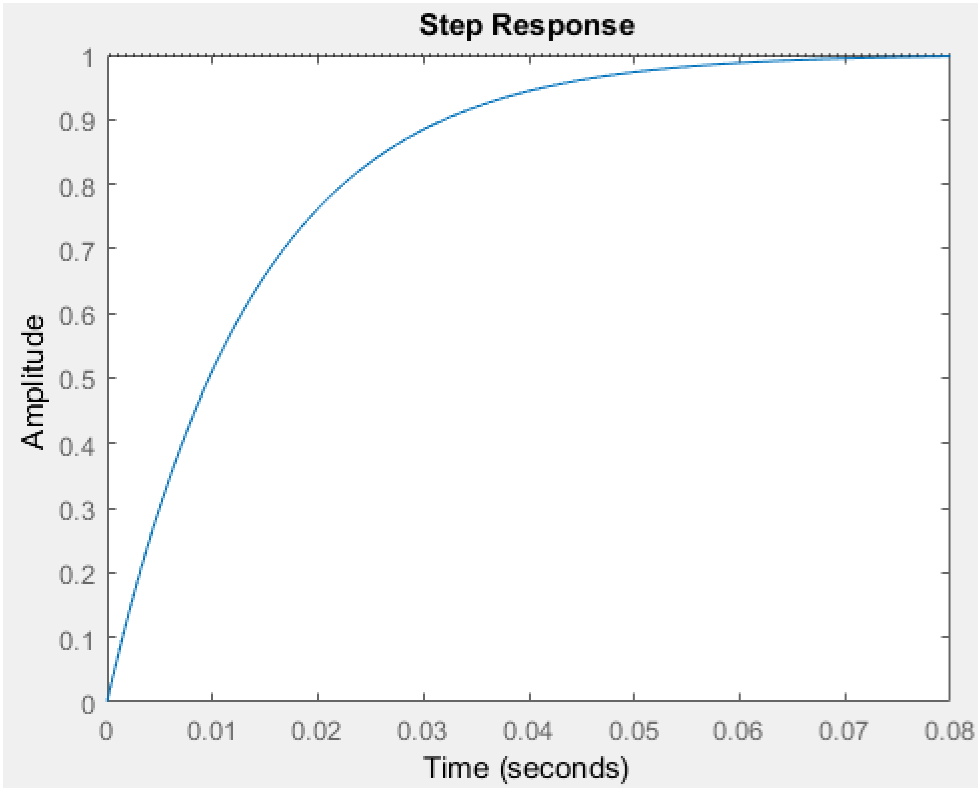


Figure 3.4. Step Response of Closed Loop Transfer Function

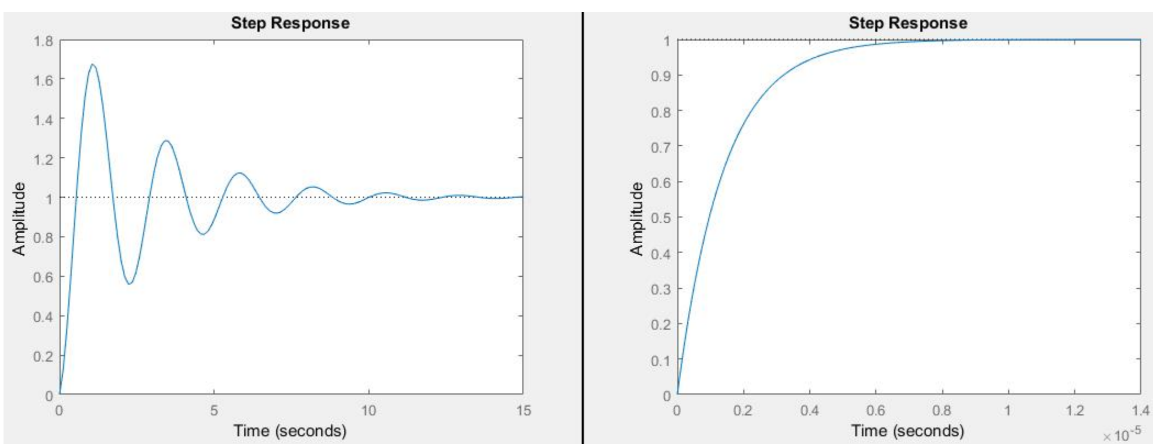


Figure 3.5. Step Response for Low and High Values of Kp

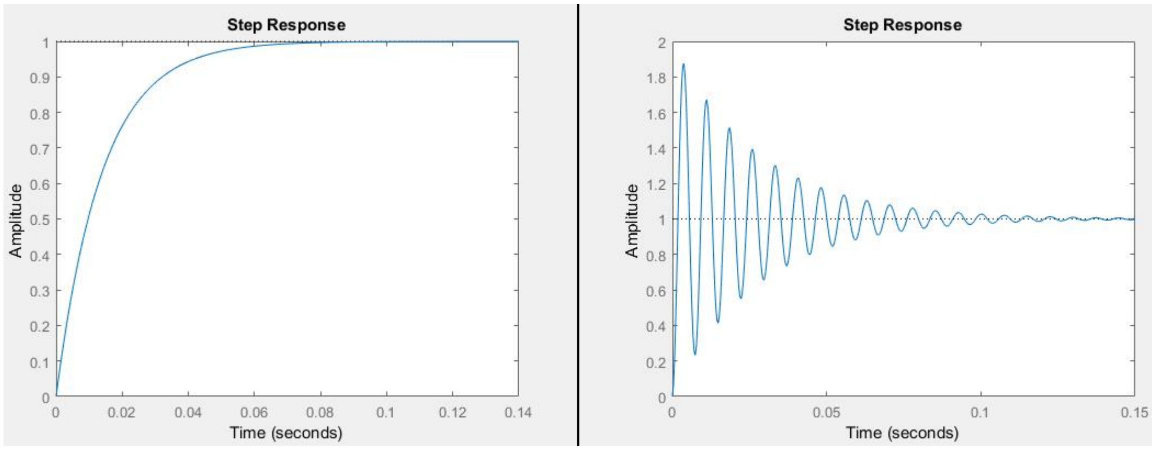


Figure 3.6. Step Response for Low and High Values of Ki

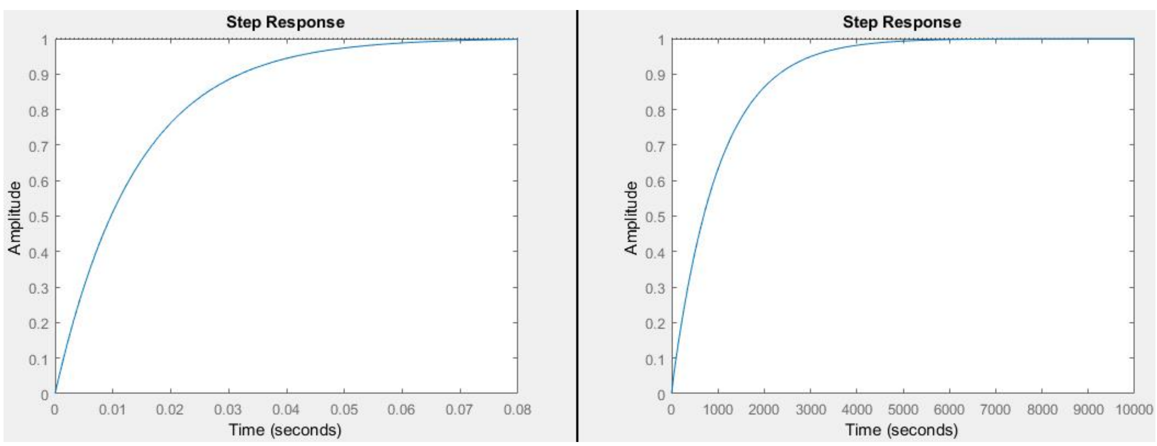


Figure 3.7. Step Response for Low and High Values of Damping

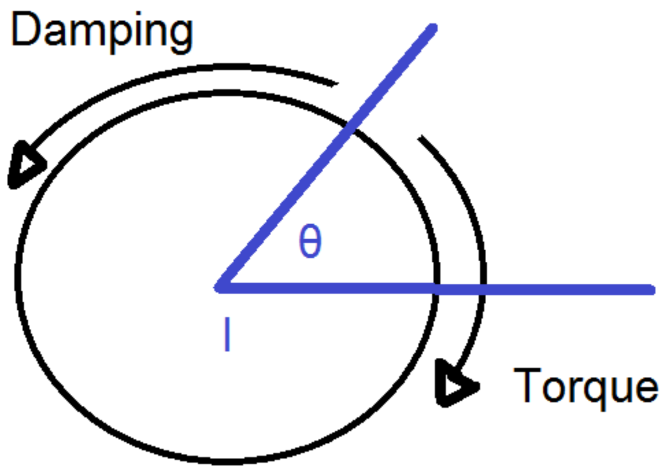


Figure 3.8. Free Body Diagram of Extruder Screw

Figure 3.8 shows the free body diagram used to create equations 5-9 and the subsequent Matlab model. The large circle represents the end of the screw which is being turned by the torque, and opposed by the damping force. The moment of inertia is represented by I, and θ represents the screw position.

The results from this simulation will be helpful in determining how to tune the extruder for an optimal bead width. From the simulations, Kp cannot be too high and Ki cannot be too low or else oscillations will occur which will cause fluctuations in bead width.

Single Bead Wall Test

To calibrate the extruder and test the effect of each control variable, a single bead wall test print was created. A single bead wall means that the extruder will

make just one pass, or closed loop, for the print. This specific closed loop is in the shape of a rectangle with three rounded corners. The width of the bead at various positions will be measured with calipers and recorded. During all tests, the pressure, extrusion temperature, environment temperature, bed temperature, and pellet lot will remain the same. There are five individual build sheets inside the printer where parts can be made. For consistency, all prints will be done on sheet one.

The Design

The single bead wall test is a one layer part where the extruder makes just one closed loop pass to complete the part. This test is a good representation of several types of print moves that the machine will see during the printing of a typical part. The starting and stopping point (colored pink in Figure 3.11) is at a right angle corner. Starting from that corner, the extruder travels down (+x direction on the machine) towards a one inch radius corner, then a five inch radius corner, before hitting a ten inch radius corner and returning to the start/stop point. Between each corner is a straightaway that allows the gantry/extruder to accelerate to top speed and then decelerate before changing direction.

Each turn, or corner, is of a different radius because each will cause a different amount of acceleration and deceleration for the extruder. A turn, such as the radii in this print, is not programmed as one single move. The GCode only employs straight line moves. This means that a radius must be approximated by tens, if not hundreds, of short line segments. This causes the gantry system to constantly accelerate and decelerate between points as is traverses the corner.

The long sides are forty eight inches long and the short sides are twenty-four inches long. An overhead drawing of this part is shown in Figure 3.9 and a look at the solid model is seen in Figure 3.10. A GCode representation is

depicted in Figure 3.11, and the actual GCode used for printing can be seen in appendix A.1.

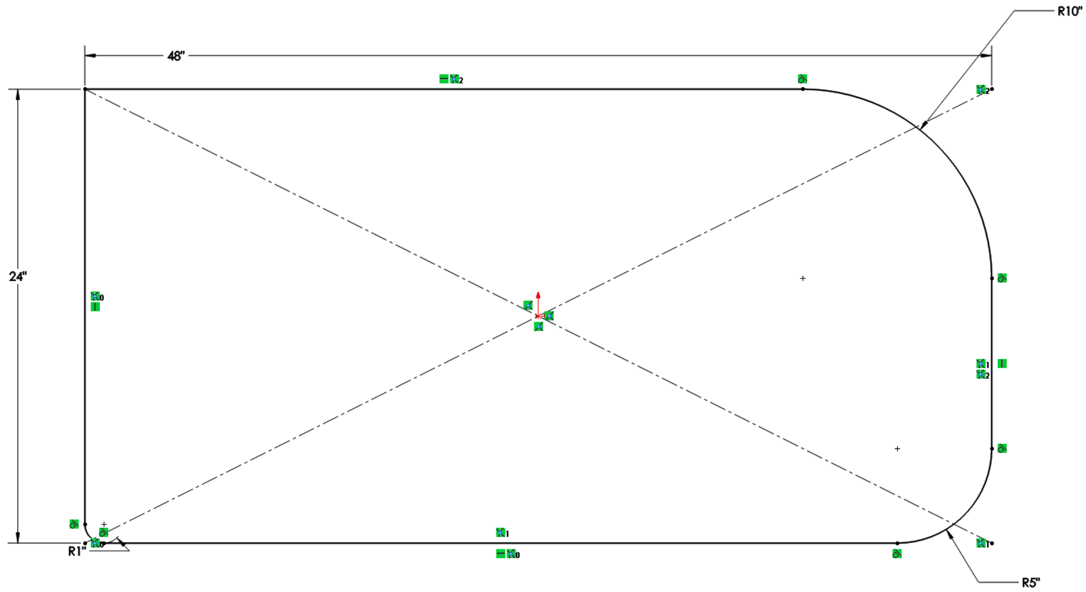


Figure 3.9. Single Bead Wall Rectangle Drawing

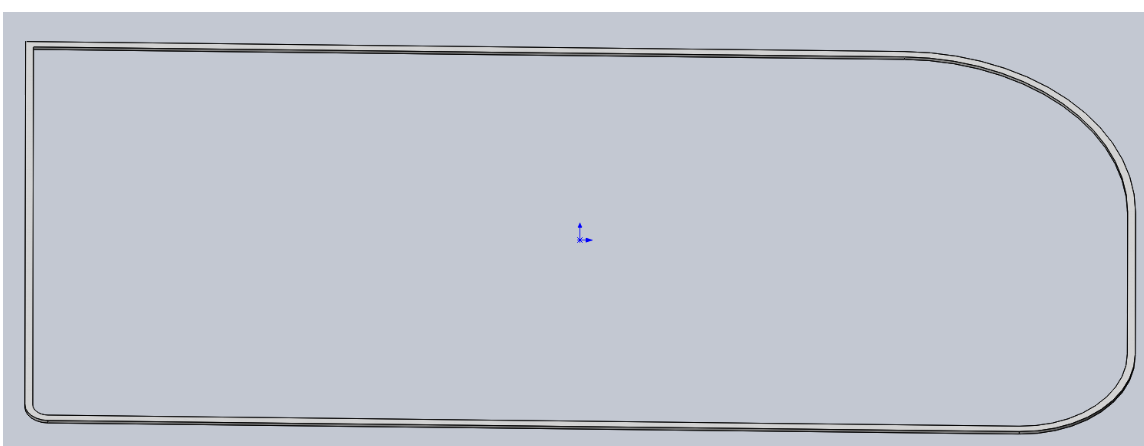


Figure 3.10. Solidworks Part Visualization

Figure 3.11. Single Bead Wall GCode Visualization

CHAPTER FOUR
EXPERIMENTAL RESULTS

This chapter details all results obtained from the performed research. Some of the evaluated tuning strategies performed well and others did not work well at all. As an example, the proportional gain could only be set so high or the current limit would be exceeded resulting in a drive fault. This high gain could be used at very low speeds, but made the extruder vibrate during startup because of the significant overshoot in response to a step command in speed.

Thirty-five total tests were conducted of the single bead wall rectangle. With each subsequent test, the values of the control variables were changed as seen in Appendix A.2. A picture of one printed rectangle can be seen in Figure 4.1. After printing, each piece would be numbered and then marked at all measurement locations with a sharpie. Appendix A.3 provides a list of all measurement points with descriptions about the locations. There are twenty-two total measurement locations for each print where the bead width would be measured and recorded. The measurement points were divided into two main categories, each having three subcategories. The two main categories are radii and straightaways. The radii are three arcs, or corners, that were printed with radii of 1”, 5”, and 10”. The three straightaways are the straight paths leading up to each measured radius. Each radius or straightaway was measured at the start and end point with a few additional measurements in the middle depending on the total length of the move. The statistical data from these 3-5 bead width measurements for each move was used to compute the average, variance, and standard deviation for each print type and for the overall print so that each instance of the specimen could be directly compared. All of the raw bead width measurements data are documented in Appendix A.4.

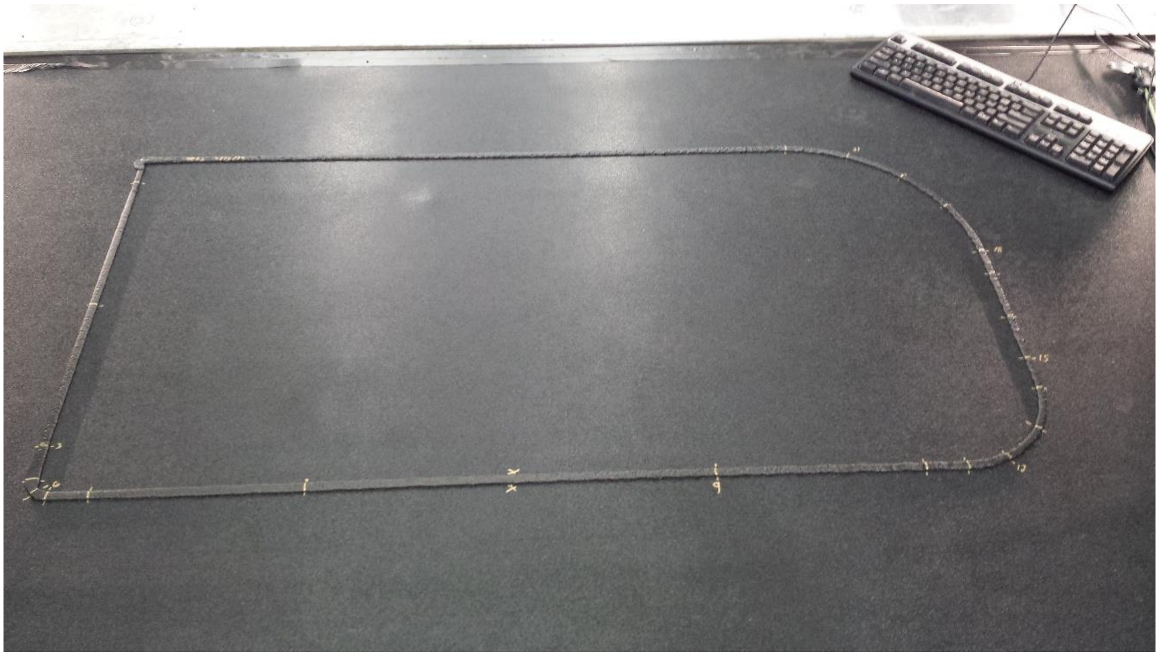


Figure 4.1. Completed Test Print with Marked Measurement Locations

The results will be broken down into four main sections. The first will focus on a comparison of all thirty-five tests and focus on average bead width, variance, and standard deviation. The second will be an analysis of how each variable effects the printing and sensitivity of the system performance to the parameter variance and will include a subsection for each variable. The third section covers repeatability of results and the fourth will compare the final results to the initial results to demonstrate the stability of the system.

Comparing All Tests

The raw data for all thirty-five tests can be seen in appendix A.4. The following subsections will discuss average bead width, variance, and standard deviation. They will show comparisons between straightaway moves and radius moves. In

all graphs, test number twenty-eight appears to be an outlier and that something in the data might have been entered incorrectly; however, this is not the case. The reason the numbers for this test are so abnormal is because of the very low proportional gain. This test will be explained in further detail in the section about Kp.

Average Bead Width

The bead width was measured in twenty-two places for each print (see appendix A.3 for locations). From this data, the average was computed for straightaways, radii, and the entire print. Figure 4.2 shows a graph comparing the straightaway, radius, and overall average for all thirty-five tests. The average bead width of all 770 test points is 0.4478”. The average of all radius points is 0.4511” and for straightaway points is 0.4444”.

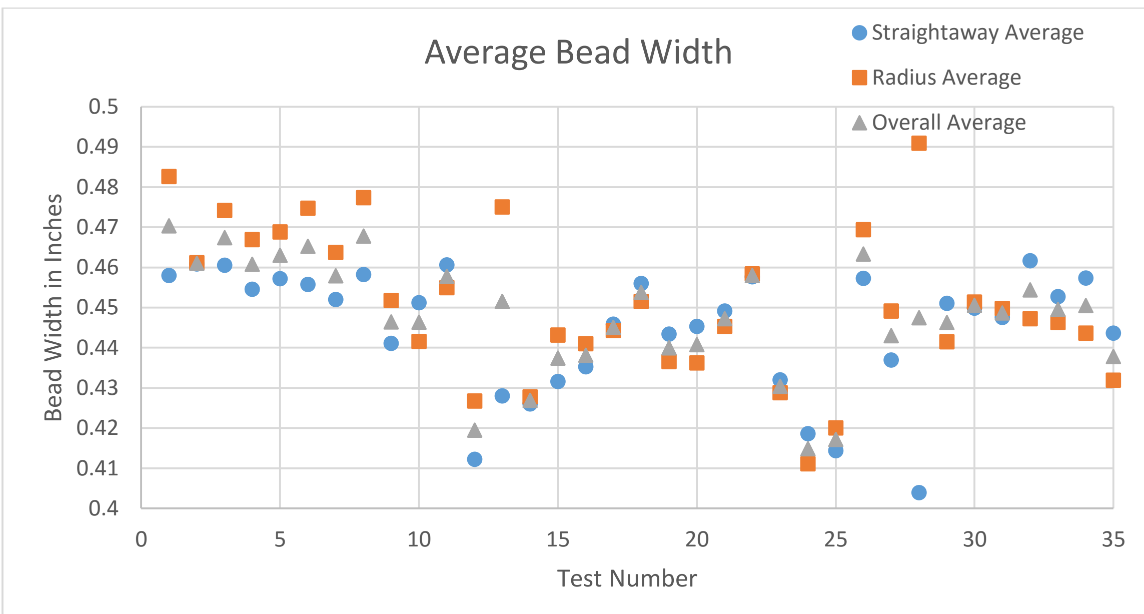


Figure 4.2. Average Bead Width

As seen in Figure 4.2, the average bead width varies significantly throughout the testing process. For most tests, the average bead width during the radius moves is larger than that of the straightaway moves. This is caused by the slowing, or deceleration, of the extruder to enter the corner and then having to quickly accelerate back out of the corner. The straightaways have a smaller average bead width because the machine is traveling at a higher speed during these moves. The commanded print speed for the rectangle was 10.8inches/second with the spindle, or extruder, turning at 400rpm.

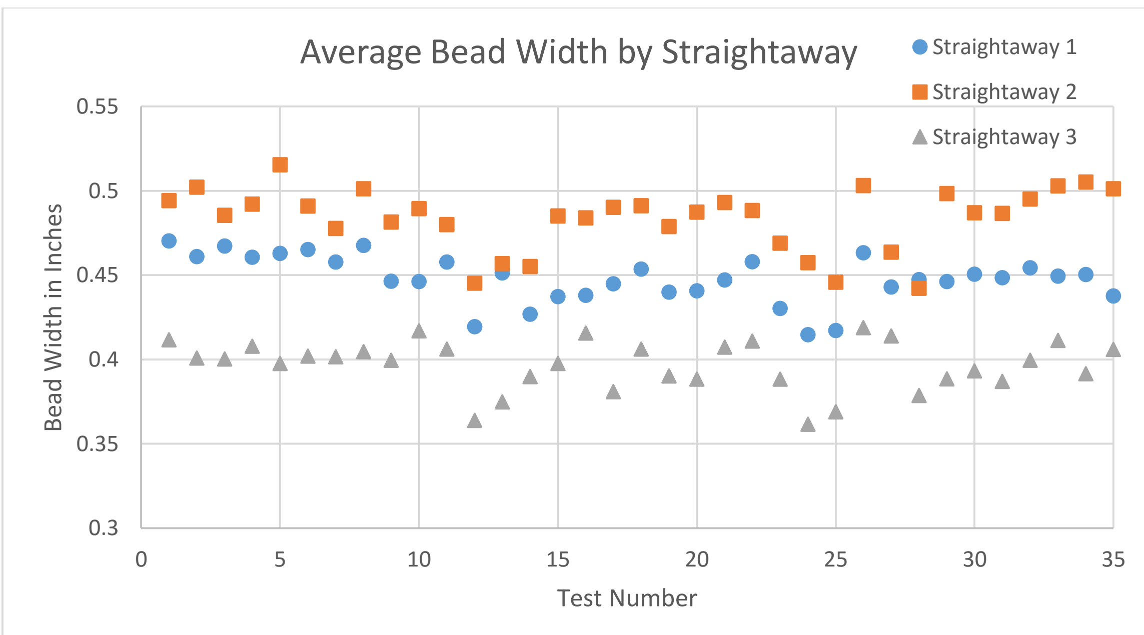


Figure 4.3. Average Bead Width for Straightaways

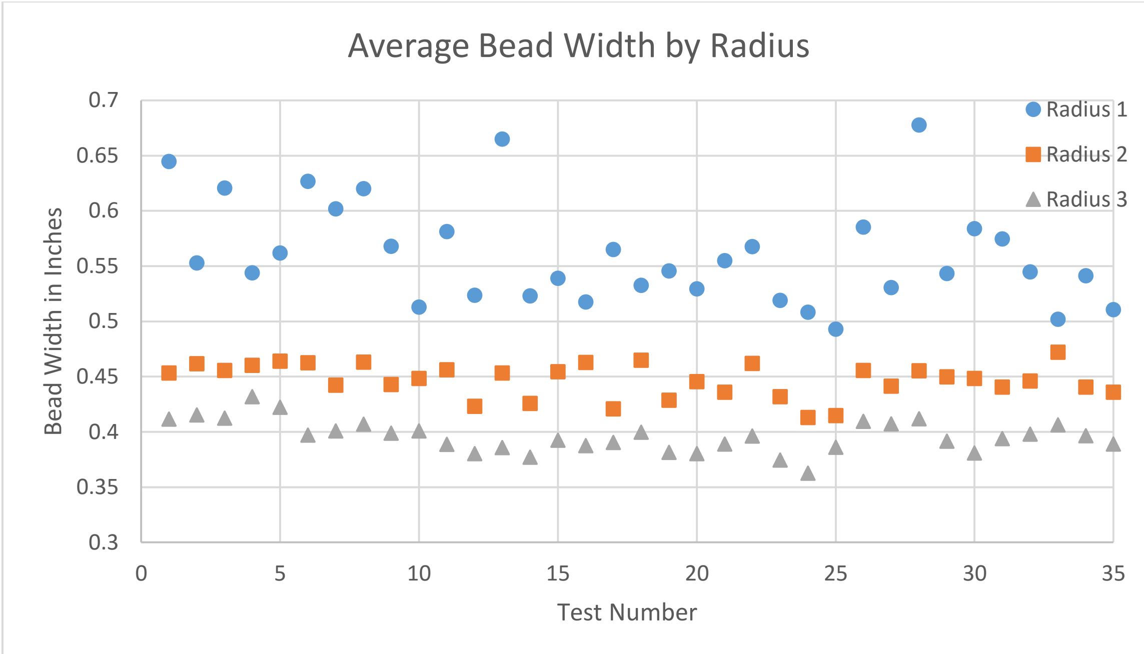


Figure 4.4. Average Bead Width for Radii

Figures 4.3 and 4.4 above breakdown the bead width averages by specific straightaway and radius. Straightaway two is the longest straightaway, meaning the machine spends the most time at actual print speed giving the extruder time to reach top speed and maintain a steady state. Because of this, straightaway two has the largest bead width average. Straightaway three is the shortest, and is where the narrowest average bead width occurs. The opposite effect happens with radius moves. Radius one is the shortest, at just one inch. This requires the most rapid acceleration and deceleration of the gantry which is quicker than the extruder can react. Because of this, the average bead width during this radius is quite large. Radius three is the longest radius, closely resembling a short straightaway (like straightaway three) in the way the machine accelerates and decelerates, which results in it having a narrower average bead width.

Variance and Standard Deviation

Variance is used to show how far the numbers are spread out. A small variance is most ideal and is the goal of this thesis, to minimize variance during both radius and straightaway moves. Variance for the radius moves is larger than for straightaways due to more fluctuation in bead width from the rapid acceleration and deceleration. Radius three is most similar to a straightaway due to the radius being made up of several short line segment moves. Because of this, radius three has the least variance while radius one has the most variance. The short distance of radius one allows very little time for the acceleration and deceleration of the extruder, resulting in a fat bead entering the corner and a narrow bead exiting the corner.

The standard deviation shows how spread out the measurements are from one another. It is the square root of the variance, meaning that the standard deviation plots should resemble the variance plots. A standard deviation of zero implies that all the numbers are the exact same. Figure 4.5 shows the standard deviation for all tests, all radius, and all straightaway moves.

As previously mentioned, the standard deviation for test twenty-eight is significantly higher than any of the other tests. This is because the acceleration and deceleration are very slow, allowing for a large error to propagate in the sluggish controls. This will be explained in detail in the section about Kp, which is the cause of this problem. As with the variance and bead width average, the standard deviation is larger for the radius moves than it is for the straightaway moves. This is caused by the repeated quick accelerations and decelerations, which result in inconsistent velocities when cornering.

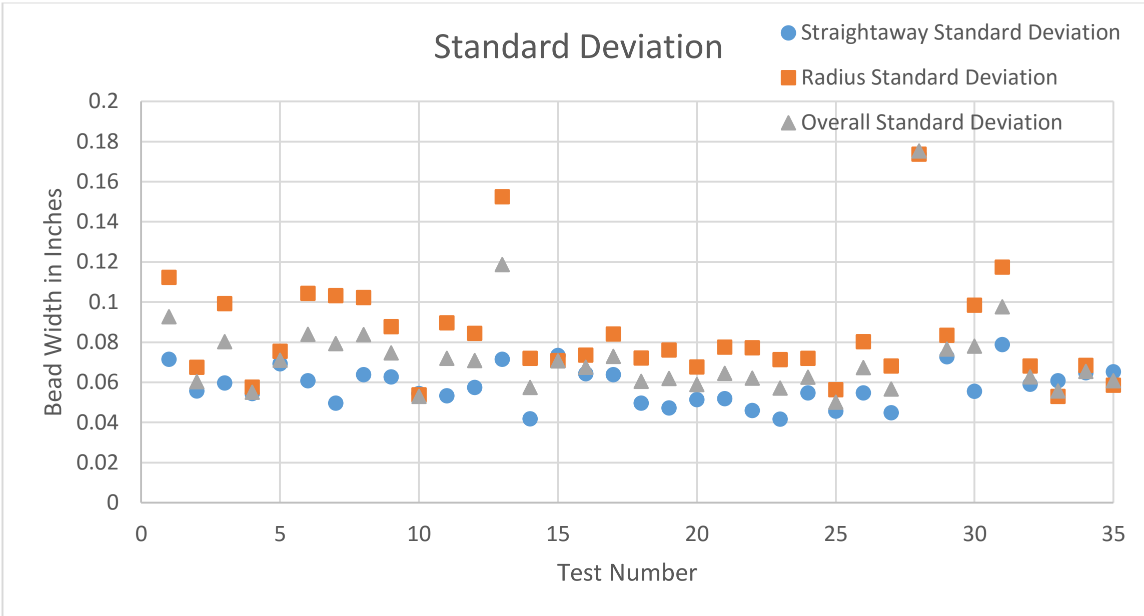


Figure 4.5. Standard Deviation for all Tests

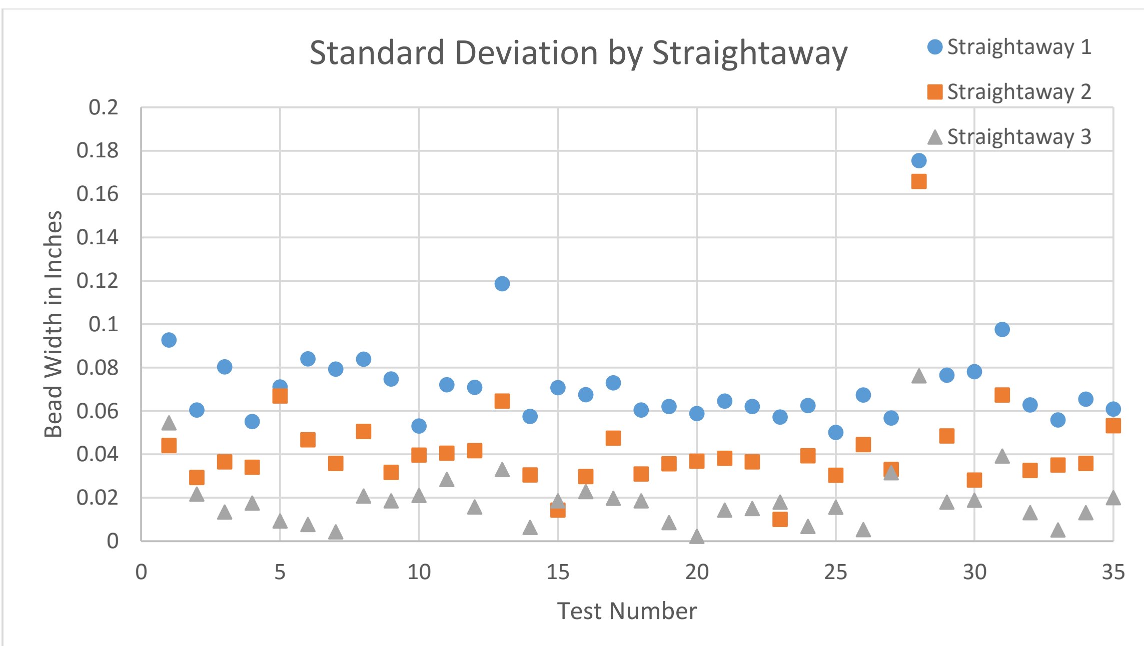


Figure 4.6. Standard Deviation for Straightaways

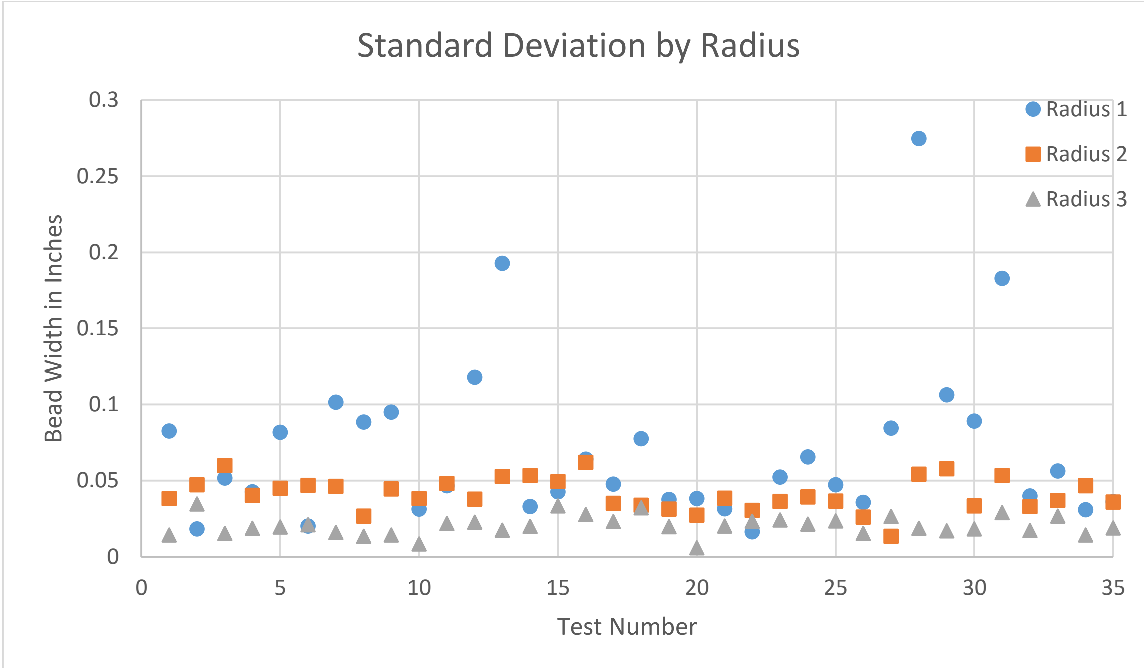


Figure 4.7. Standard Deviation for Radii

Looking at Each Variable

As demonstrated in the previous section, thirty-five total experiments were performed to show how much the bead width can vary and just how inconsistent it can be given changes in motor tuning. This section segments the results into each of the five specific variables and focuses on the contribution of each variable to the desired output control, bead width. The plots for these sections will be focused only on the variance. Average bead width is not as important as variance is for tuning in these parameters.

Kp

Kp is the proportional gain. Figure 4.8 illustrates how the change in Kp affects radius and straightaway variance. A small value of Kp causes slow acceleration

and deceleration. On test twenty-eight, this meant that a value couldn't be measured at point one because the extruder had not accelerated fast enough to build the needed pressure to output plastic. It also meant that entering the radii, specifically the first one, the bead was very wide as the extruder had just started to slow down when the gantry had slowed to the proper speed. This had the opposite effect on the exiting the radius. At the exit, the extruder couldn't accelerate fast enough to keep up with the gantry, so the bead was stretched very thin. Radius one for test twenty-eight can be seen in Figure 4.9.

When testing out Kp, it was found that the highest usable value of Kp was 0.01. Any value larger than this was too high of a gain and caused the extruder to vibrate and fault the drive. This is because a high proportional gain causes the servo motor to accelerate and decelerate very quickly which results in overshoot. Values of Kp larger than 0.01 could be used, but only at extruder speeds of less than 400rpm and would still cause the extruder to jerk at startup.

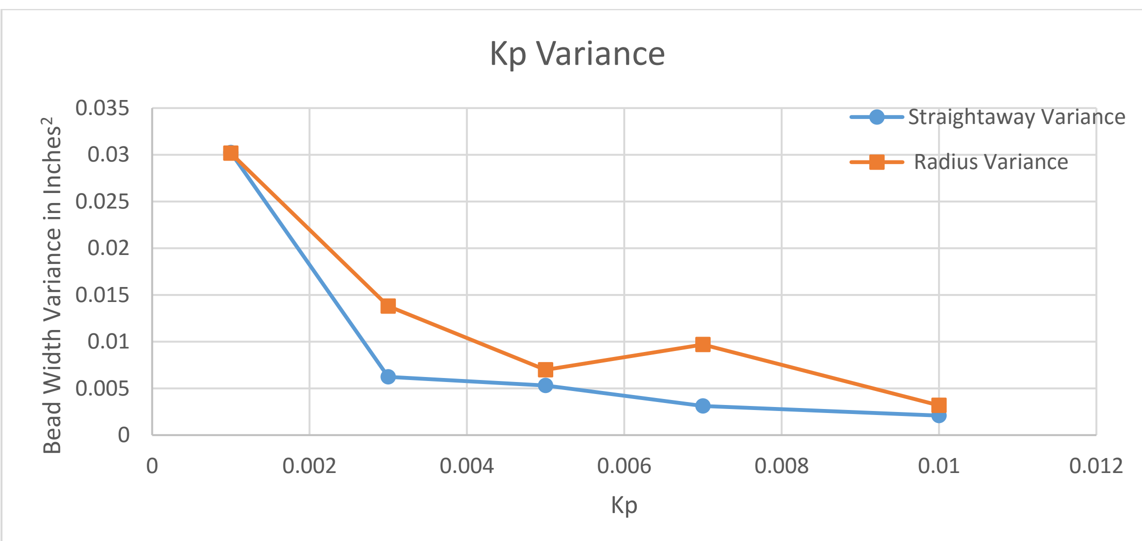


Figure 4.8. Kp Variance

Figure 4.9. Radius 1 of Test 28

Ki

Ki is the integral gain. Changing the value of Ki had very little effect on the straightaway variance, but did impact the radius variance. Initially the integral gain was turned off, which caused the largest variance for straightaways. Overall, changing Ki didn't have a drastic effect on the variance since its effect is based on the integral of the position error, and with a sufficiently tight control the errors are small and therefore it has little influence. Figure 4.10 illustrates how the change in Ki affects radius and straightaway variance.

Ki was also limited to a maximum value of 0.01, like Kp. Again, this was because such a large gain would cause the extruder to overshoot on startup resulting in oscillation of the extruder. This faulted the drives and caused the program to pause so as not to cause damage to the machine.

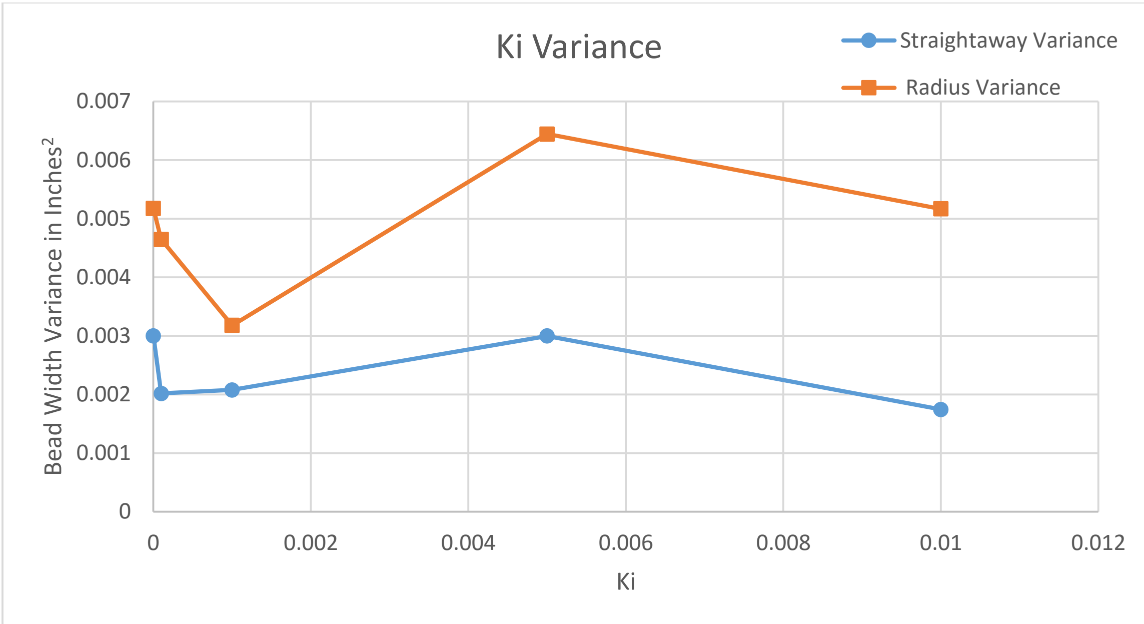


Figure 4.10. Ki Variance

Kvfb

Kvfb is the velocity feedback. Changing the value of Kvfb also had little effect on the straightaway variance, but did impact the radius variance, especially at high values of Kvfb. The initial value of Kvfb was 0.6, but the best determined value was 0.4. Increasing Kvfb increased the radius variance. This is because the velocity error when traveling around a radius is larger than that of a straightaway. Kvfb is multiplied by this error and compounds the problem, producing a larger variance. Figure 4.11 illustrates how the change in Kvfb affects radius and straightaway variance.

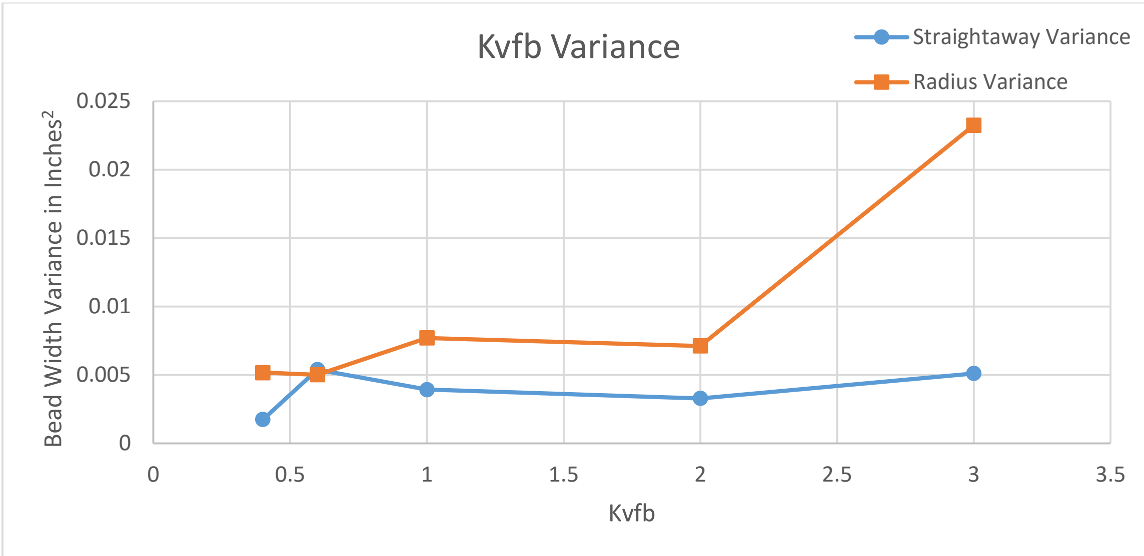


Figure 4.11. Kvfb Variance

Values of Kvfb below 0.4 would not run. This is because a system without feedback becomes unstable. Feedback is used to measure what is going on in the present state of the system so that the system knows how the plant is functioning.

Kvff

Kvff is the velocity feedforward. Increasing this feedforward made variance increase for both radii and straightaway moves. Without velocity feedforward enabled, the gantry system would not extrude plastic, it would vibrate because the system was so unstable. This explains why the straightaway variance decreased as Kvff approached 0.01 and then increased as it approached zero. Figure 4.12 illustrates how the change in Kvff affects radius and straightaway variance.

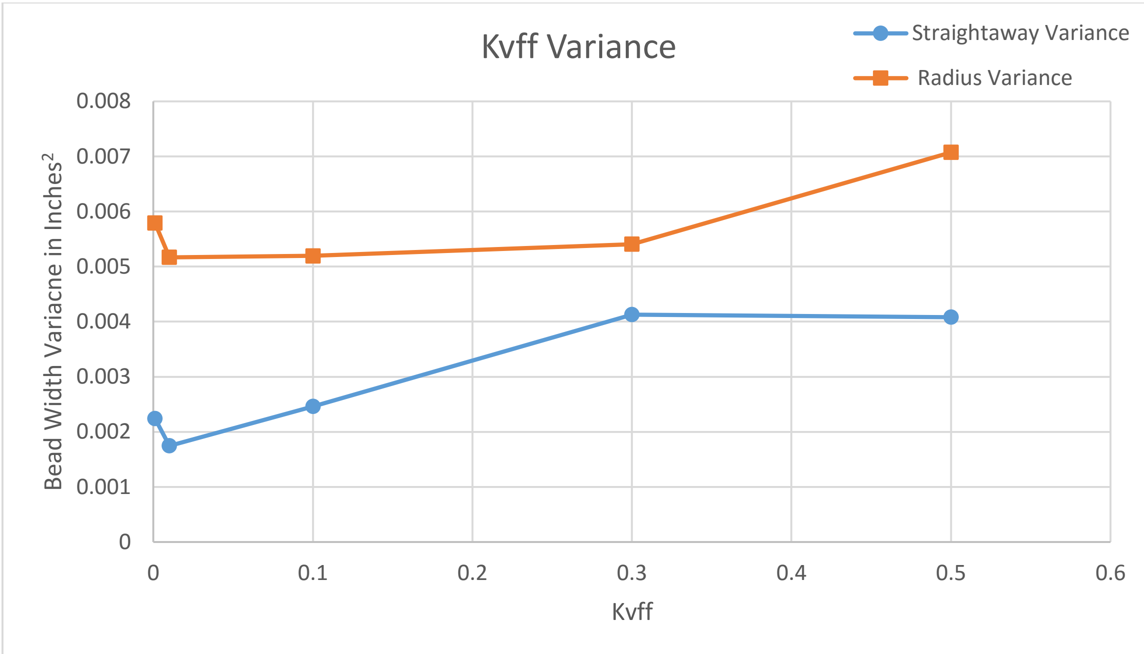


Figure 4.12. Kvff Variance

Kaff

Kaff is the acceleration feedforward. Changing Kaff resulted in little noticeable effect on bead width. Through testing, the radius variance had a range of ~0.0015in^2 and the straightaway variance had a range of ~0.001in^2, both very small. Values of 0.25 and 0.5 produced nearly identical results, with 0.5 being only slightly better. Figure 4.13 shows what little affect changing Kaff has on radius and straightaway variance.

A high value for Kaff gives a more consistent bead width because the acceleration is smoother. The downside to this is that the initial bead width, at point one, is often narrow. This is because the extruder doesn't accelerate to steady state from a dead stop as fast. This delayed startup time can be compensated for in the ORNL Slicer by increasing the on delay time and initial pump speed. The initial pump speed is the speed that the extruder is set for at

the start of a print move while the gantry is commanded to remain in place. After the duration of the on delay, the gantry begins moving as the extruder is commanded to accelerate to printing speed. This feature was implemented to make sure that the starting point of each bead was securely anchored to the build sheet.

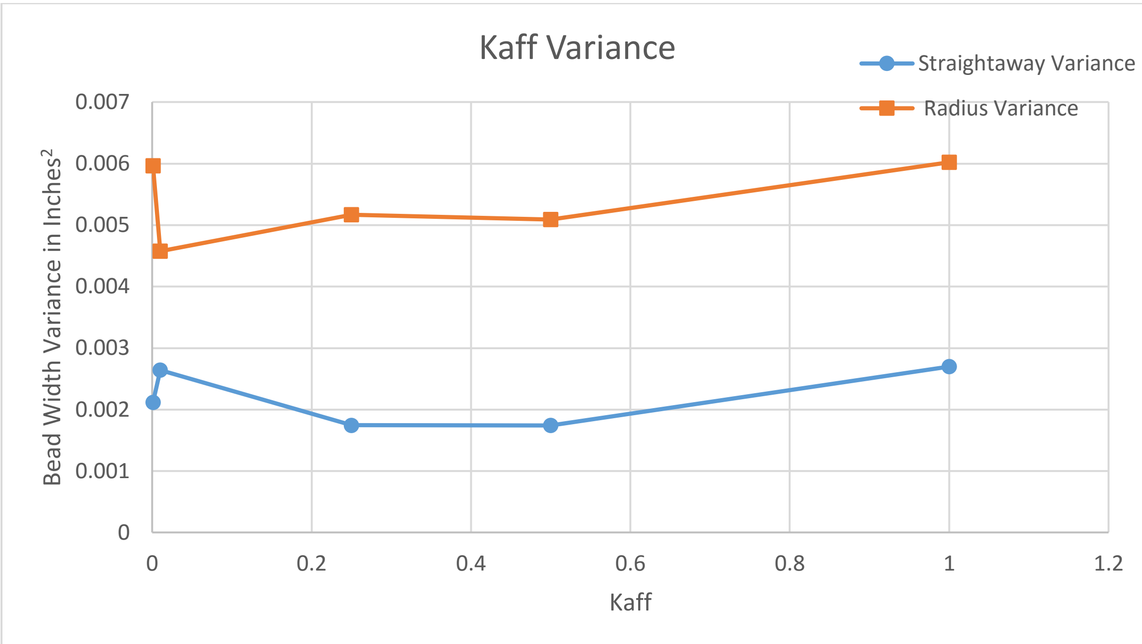


Figure 4.13. Kaff Variance

Repeatability of Results

To ensure that the data wasn't achieved through coincidence, the values determined to be best were tested five times and compared to ensure that the results didn't change. The following plot compares the variance for all five tests.

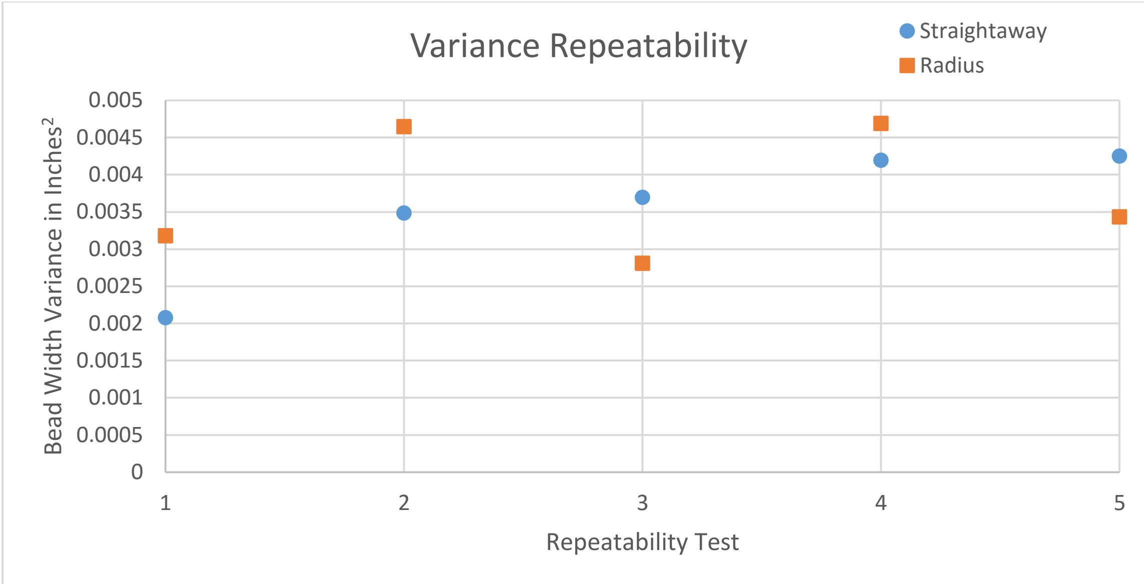


Figure 4.14. Repeatability of Variance

Figure 4.14 shows that the variance from test to test is quite minimal. There appears to be a large drop-off in radius variance from test two to test three, but the difference in variance is less than 0.002in^2. This variance corresponds to a standard deviation of 0.045in, only 10% of the mean bead width. A difference in variance so small is negligible and should be considered measurement error.

Improvement over Original Values

The original control values, used in test one, are compared to the average of all five repeatability tests. The differences can be seen in Table 4.1. Overall, the variance was significantly decreased for both straightaways and radii print moves. This also corresponds to a lower standard deviation.

Table 4.1. Variance Comparison

	Average of Repeatability Tests	**Original Control Values**	**Percent Change**
Radius Variance	0.003752065	0.012613855	-70.25441
Straightaway Variance	0.003541527	0.0050992	-30.5473942
Overall Variance	0.003561305	0.008593751	-58.5593681

CHAPTER FIVE
CONCLUSIONS

This thesis presents a method for testing and tuning screw based extruders for polymer extrusion in additive manufacturing. Five control values are configured to minimize variance in bead width both during straightaways and around corners. Some of these values did not have a large effect on straightaway variance, in comparison to their effect on radius variance. This was to be expected because the more serious variance problem occurs during radius moves where the extruder must rapidly accelerate and decelerate. Some of these controls had limits to what their value could reach without leading to instability which resulted in faults of the controller. For example, the proportional and integral gains (Kp and Ki respectively) could not exceed a maximum level of 0.01 or the servo motor drive would trip, meaning too much current was requested. This is because these large gains resulted in the servo being commanded to accelerate at rates faster than it was capable of achieving. Turning off, or setting control values to zero, such as the feedback, made the machine unstable. This was very visible because the machine would begin shaking as it was trying to accelerate and decelerate simultaneously causing it to oscillate about the commanded value.

Ultimately, the five control values were tuned in to provide average bead width variances of just 0.003542in^2 and 0.003433in^2, for straightaway and radius moves respectively. This is a decrease of 36.1% for straightaways and 108.3% for radius moves resulting in an overall variance decrease of 82.8%. The final five values can be seen in Table 5.1. These values represent the minimum variance over all values tested. Further testing could be done to find the absolute minimum for each variable. The value of Kp ultimately remains unchanged, but this thesis demonstrates the effects of increasing and decreasing this value.

Table 5.1. Initial and Final Control Values

	Initial Value	Final Value
Kp	**0.01**	**0.01**
Ki	**0**	**0.001**
Kvfb	**0.6**	**0.4**
Kvff	**0.285**	**0.01**
Kaff	**1**	**0.5**

Future Work

This work to optimize screw based extruders for maximum consistency of bead width is just beginning. Tuning the extruder acceleration values is only one part of the problem. Bead width variance is also affected by extrusion temperature, extrusion pressure, and pellet geometry. Furthermore, all of these factors are strongly correlated to one another. The current Matlab model detailed in this paper only incorporates two of the control variables but can be expanded to account for all control variables as well as extrusion pressure and temperature. A more advanced model that takes all these factors into account will allow simulations of new materials and new flowrates.

While this bead width optimization is continuing, there will also be research to increase mass flow rate. The system used for this thesis is only capable of 100lbs/hr of material (the initial system at ORNL only did 10lbs/hr), but the industry continues to demand a machine that is an order of magnitude faster. This large increase in flow rate will not only allow parts to be printed faster, but it will make very large parts possible. Parts that are on the order of 100ft long aren’t feasible just yet because the time to print one layer of a part that size is measured in hours, instead of minutes like with BAAM. A layer time that exceeds twenty minutes causes the part to cool too much which prevents successful layer bonding. Without a strong layer to layer bond, part delamination will occur. The

field of Big Area Additive Manufacturing is just emerging and there is still a considerable amount of research and development work needed to bring it to the mass market.

REFERENCES

[1] Ferry P.W. Melchels, “Celebrating three decades of stereolithography,” in *Virtual and Physical Prototyping* Vol. 7, No. 3, 2012, pp. 13-175.

[2] Fabio Previdi, Sergio M. Savaresi, and Angiolino Panarotto, “Design of a feedback control system for real-time control of flow in a single-screw extruder,” in *Control Engineering Practice 14*, 2006, pp. 1111-1121.

[3] M.H. Costin, P.A. Taylor, and J.D. Wright, “On the Dynamics and Control of a Plasticating Extruder,” in *Polymer Engineering and Science Vol. 22, No. 17*, 1982, pp. 1095-1106.

[4] M.H. Costin, P.A. Taylor, and J.D. Wright, “A Critical Review of Dynamic Modeling and Control of Plasticating Extruders,” in *Polymer Engineering and Science Vol. 22, No. 7*, 1982, pp. 393-401.

[5] Douglas E. Smith, Daniel A. Tortorelli, and Charles L. Tucker III, “Optimal Design for polymer extrusion Part I: Sensitivity analysis for nonlinear steady-state systems,” in *Computer Methods in Applied Mechanics and Engineering 167*, 1998, pp. 283-302.

[6] Douglas E. Smith, Daniel A. Tortorelli, and Charles L. Tucker III, “Optimal Design for polymer extrusion. Part II: Sensitivity analysis for weakly-coupled nonlinear steady-state systems,” in *Computer Methods in Applied Mechanics and Engineering 167*, 1998, pp. 303-323.

[7] Z. Tadmor, S.D. Lipshitz, and R. Lavie, “Dynamic Model of a Plasticating Extruder,” in *Polymer Engineering and Science Vol. 14 No.2*, 1974, pp. 112-118.

[8]Yaskawa Electric Corporation, MANUAL NO. SIEP S800000 45K: 209 2014

[9] Franklin, Gene F., J. David. Powell, and Abbas Emami-Naeini, “Basic Properties of Feedback,” in *Feedback Control of Dynamic Systems*, Upper Saddle River, NJ: Pearson/Prentice Hall, 2006, pp. 1,186-187

APPENDIX

A.1 Single Bead Rectangle GCode

```
(ORNL-Cincinnati Part File)
(Filename: Single Bead Rectangle Test.nc)
(GCODE Flavor: Cincinnati-BERTHA)

(Slicing Parameters)
(Nozzle Diameter: 0.30in)
(Printer Base Offset: -14.6900in)
(Default Extrusion Width: 0.335in)
(Downskin Count: 0)
(Upskin Count: 0)
(Number of Insets: 1)

(SAFETY BLOCK - ESTABLISH OPERATIONAL MODES)
T1 (EXTRUSION TOOL)
M11 (EXIT SERVOING MODE)
M66 L0.1770 (SET DEFAULT ACCELERATION VALUE)
M16 (DEFAULT SPINDLE ADJUSTMENT)
G20 (PROGRAMMING IN INCHES)
G53 (ABSOLUTE COORDINATE SYSTEM)
G17 (XY PLANE SELECTION)
G25 (SPINDLE FLUCTATION DETECTION OFF)
G40 (TOOL RADIUS COMPENSATION OFF)
G43 (TOOL HEIGHT OFFSET COMPENSATION NEGATIVE)
G50 (SET AXES SCALE FACTORS TO 1)
G50.1 (PROGRAMMED MIRROR IMAGE CANCEL)
```

```
G96 (CONSTANT SURFACE SPEED, BY VARYING SPINDLE SPEED)
G69 (COORDINATE ROTATION CANCEL)
G64 (NORMAL CUTTING MODE)
G90 (ABSOLUTE VALUE DISTANCE MODE)
G94 (INCHES/MINUTE FEEDRATE MODE)
G1 F120 (SET INITIAL FEED RATE)
M0 (WAIT FOR USER)
G4 P0.25 (Dwell)
G1 F7.0866 W0.0000 (Set Initial Table Height)

(MATERIAL LOAD - INITIAL PURGE)
M69 S1 (PURGE)
M0 (WAIT FOR USER)

(Start Build)
(Layer count: 1)
(BEGINNING LAYER: 1)
G1 F59.0551 Z-14.2100 (TRAVEL-Move Extruder off table)
G1 F7.0866 W-0.1500 (Moving Table)
M66 L0.1593 (SET ACCELERATION VALUE)
G1 Z-14.2100 F59.0551 (TRAVEL-Lift Tip For Travel)
G0 X207.1666 Y18.9166 (TRAVEL)
G1 W-0.1500 (TRAVEL)
G1 Z-14.6900 (TRAVEL-Lower Tip)
(TYPE:VOLUME)
M66 L0.2655 (SET ACCELERATION VALUE)
M12 (PERIMETER SPINDLE ADJUSTMENT)
M64 L0.70 (Turn Tamper ON)
M60 L1000 P10 (TURN FEED SHAKER ON)
```

M11 (Turn OFF Extruder Servoing)

M3 S300.0 (Turn Pump on)

G4 P0.05 (Dwell)

M3 S400.0 (Change Pump Percentage)

M10 (Turn ON Extruder Servoing)

G1 F649.6063 X229.9941 Y18.9166 (VOLUME-WALL_OUTER)

G1 X230.1235 Y18.9268 (VOLUME-WALL_OUTER)

G1 X230.3774 Y19.0093 (VOLUME-WALL_OUTER)

G1 X230.5879 Y19.1622 (VOLUME-WALL_OUTER)

G1 X230.7367 Y19.3671 (VOLUME-WALL_OUTER)

G1 X230.8333 Y19.7694 (VOLUME-WALL_OUTER)

G1 X230.8333 Y61.7441 (VOLUME-WALL_OUTER)

G1 X230.7733 Y62.5059 (VOLUME-WALL_OUTER)

G1 X230.5964 Y63.2432 (VOLUME-WALL_OUTER)

G1 X230.3062 Y63.9439 (VOLUME-WALL_OUTER)

G1 X229.9097 Y64.5911 (VOLUME-WALL_OUTER)

G1 X229.4174 Y65.1674 (VOLUME-WALL_OUTER)

G1 X228.8411 Y65.6597 (VOLUME-WALL_OUTER)

G1 X228.1939 Y66.0562 (VOLUME-WALL_OUTER)

G1 X227.4932 Y66.3463 (VOLUME-WALL_OUTER)

G1 X226.7559 Y66.5233 (VOLUME-WALL_OUTER)

G1 X225.9941 Y66.5834 (VOLUME-WALL_OUTER)

G1 X217.0065 Y66.5834 (VOLUME-WALL_OUTER)

G1 X215.5998 Y66.4827 (VOLUME-WALL_OUTER)

G1 X214.2302 Y66.1849 (VOLUME-WALL_OUTER)

G1 X212.9151 Y65.6942 (VOLUME-WALL_OUTER)

G1 X211.6842 Y65.0223 (VOLUME-WALL_OUTER)

G1 X210.5604 Y64.1808 (VOLUME-WALL_OUTER)

G1 X209.5692 Y63.1896 (VOLUME-WALL_OUTER)

```
G1 X208.7277 Y62.0657 (VOLUME-WALL_OUTER)
G1 X208.0557 Y60.8348 (VOLUME-WALL_OUTER)
G1 X207.5651 Y59.5198 (VOLUME-WALL_OUTER)
G1 X207.2672 Y58.1502 (VOLUME-WALL_OUTER)
G1 X207.1666 Y56.7434 (VOLUME-WALL_OUTER)
G1 X207.1666 Y18.9166 (VOLUME-WALL_OUTER)
M5 (Turn Pump OFF)
G1 F1079.5276 X209.1666 Y18.9166 (VOLUME-WALL_OUTER-FORWARD TIP
WIPE)
M66 L0.1593 (SET ACCELERATION VALUE)
M5 (TURN PUMP OFF)
G1 X210.3477 Y18.9166 F1379.5276 (TRAVEL-Spiral Lift)
G3 I-1.18 J0.00 X210.3432 Y18.8136 Z-14.6506 (TRAVEL-Spiral Lift)

M61 (TURN FEED SHAKER OFF)
M65 (Turn Tamper OFF)
(PARK)
M68 (PARK)
M11 (Turn OFF Extruder Servoing)
M30(END OF GCODE)

(Printer Settings)
(printAreaDepth 6031230)
(printAreaWidth 2194560)
(printAreaHeight 1938861)
(xOffset 3079115)
(yOffset 1085850)
(printerBaseOffset -373126)
(zTableOffset 0)
```

(zTableMinimum -1549400)
(zTableMaximum 0)
(UseDoffingStation 1)
(DoffingStationW -425450)
(majorDistance 0)
(minorDistance 0)
(showLayerTimes 0)
(gcodeFlavor 8)
(includeStartPoint 0)
(materialLoad 1)
(waitForUser 1)
(demoBoundingBox 0)
(useRelCS 0)
(useSubroutine 0)
(usePurge 1)
(purgeX 6047433)
(purgeY -11430)
(purgeZ -30533)
(purgeWipeDistance 254000)

(Material Settings)
(initialLayerThickness 3810)
(layerThickness 3810)
(layer0extrusionWidth 0)
(extrusionWidth 8509)
(initialSpeedupLayers 0)
(initialLayerSpeed 275)
(printSpeed 275)
(inset0Speed 275)

(insetXSpeed 276)
(infillSpeed 330)
(travelSpeed 1041)
(liftSpeed 25)
(supportSpeed 275)
(initialSupportSpeed 275)
(powderSpeed 275)
(tipWipeSpeed 457)
(prestartSpeed 257)
(skeletonSpeed 0)
(zTableSpeed 3)
(raftSpeed 0)
(spiralLiftSpeed 584)
(retamperSpeed 0)
(upSkinSpeed 302)
(downSkinSpeed 302)
(buildUpSpeed 0)
(slowDownSpeed 0)
(minimumSpeed 0)
(useAccel 1)
(defaultAccel -2147483648)
(perimAccel -536870912)
(insetAccel 1073741824)
(infillAccel -2147483648)
(travelAccel -1610612736)
(useArcFitting 0)
(numPointArcFitting 0)
(centerThreshold 0)
(pointDistanceThreshold 0)

(useSmoothing 1)

(smoothingType 7)

(smoothingTolerance 762)

(nPoint 0)

(perpNumPasses 0)

(opheimMinimum 0)

(opheimMaximum 0)

(langLookAhead 0)

(noSkins 1)

(upSkinCount 0)

(downSkinCount 0)

(skinOverlap 0)

(skinPattern 0)

(insetCount 1)

(insetWidth 8458)

(layer0insetWidth 8458)

(randomStartPoint 0)

(noInfill 1)

(sparseInfillLineDistance 0)

(infillOverlap 0)

(infillPattern 0)

(infillPumpRPM 0)

(fillAngle 90)

(fillRotation 90)

(useSupport 0)

(supportExtruder 0)

(supportType 0)

(supportAngle 0)

(supportLineDistance 0)

(supportXYDistance 0)
(printSupportFirst 0)
(minSupportInfillArea 0)
(minSupportArea 0)
(usePowder 0)
(powderExtruder 0)
(powderRaise 0)
(powderLayers 0)
(powderOnTime 0)
(powderOffTime 0)
(useThumper 0)
(thumperOn 0)
(thumperOff 0)
(pumpOffMinimalDistance 0)
(usePumpTime 0)
(pumpOffMinimalTime 0)
(pumpOnDelay 50)
(initialPumpRPM 300)
(insetPumpRPM 400)
(pumpRPM 400)
(ExtruderServo 1)
(useTipWipe 0)
(reverseTipWipe 0)
(wipeDistance 50800)
(retamper 0)
(usePrestart 0)
(prestartDistance 38100)
(prestartPumpRPM 50)
(useBuildUp 0)

(buildUpDistance 0)

(buildUpPumpRPM 0)

(useSlowDown 0)

(slowDownDistance 0)

(slowDownPumpRPM 0)

(spiralLift 1)

(spiralLiftEverythingButOuterPerimeter 1)

(spiralLiftDistance 30000)

(spiralLiftHeight 1000)

(spiralLiftPoints 20)

(spiralLiftContourThreshold 254000)

(spiralLiftLineThreshold 101600)

(spiralLiftAllInfill 1)

(travelLift 12192)

(useTamper 1)

(tamperVoltage 700)

(mainExtruder 0)

(nozzleDiameter 7620)

(minimalExtrudeLength 12700)

(minimumTravelDistance 76200)

(extruderOffset.Value[1].X 0)

(extruderOffset.Value[1].Y 0)

(extruderOffset.Value[3].X 0)

(extruderOffset.Value[3].Y 0)

(forceMinimumLayerTime 1)

(minimalLayerTime 500000)

(minimumPumpRPM 0)

(parkBeforePurge 0)

(parkAfterPurge 0)

(addRaft 0)
(raftMargin 0)
(raftLineSpacing 0)
(usePercentagePaths 0)
(useSkeletons 0)
(minInfillLineSize 25400)
(minSkeletonSize 0)
(minContourSize 127000)
(useExtendSharpCorners 0)
(sharpCornerExtensionDistance 0)
(sharpCornerAngleThreshold 0)
(cornerClosePointsThreshold 0)
(matrix 0)
(fixHorrible 1)
(multiVolumeOverlap 0)
(spiralizeMode 0)
(wipeDelayTime 1000)
(seamOptimization 1)
(seamSelectionX 0)
(seamSelectionY 0)
(manualDensity 0)
(plasticType 2)
(actualDensity 0)

A.2 List of Tests

Table A.2. List of Test Control Values

	Kp	Ki	Kvfb	Kvff	Kaff
#1 Originals	0.01	0	0.6	0.285	1
#2	0.01	0	0.4	0.285	0.01
#3	0.01	0	0.6	0.285	0.01
#4	0.01	0	0.4	0.05	0.01
#5	0.01	0	0.5	0.05	0.01
#6	0.01	0	0.5	0.285	0.01
#7	0.01	0	1	0.05	0.01
#8	0.01	0	1	0.01	0.01
#9	0.01	0	1	0.01	0.25
#10	0.01	0.01	1	0.01	0.01
#11	0.01	0.01	1	0.01	0.25
#12	0.01	0.01	2	0.01	0.25
#13	0.01	0.01	3	0.01	0.25
#14	0.01	0.01	0.4	0.01	0.25
#15	0.01	0.01	0.6	0.01	0.25
#16	0.01	0.01	0.4	0.3	0.25
#17	0.01	0.01	0.4	0.5	0.25
#18	0.01	0.01	0.4	0.1	0.25
#19	0.01	0.01	0.4	0.001	0.25
#20	0.01	0.01	0.4	0.01	0.01
#21	0.01	0.01	0.4	0.01	1
#22	0.01	0.01	0.4	0.01	0.001
#23	0.01	0.01	0.4	0.01	0.5
#24	0.01	0	0.4	0.01	0.5
#25	0.01	0.001	0.4	0.01	0.5
#26	0.01	0.005	0.4	0.01	0.5
#27	0.01	0.0001	0.4	0.01	0.5
#28	0.001	0.001	0.4	0.01	0.5
#29	0.005	0.001	0.4	0.01	0.5
#30	0.007	0.001	0.4	0.01	0.5
#31	0.003	0.001	0.4	0.01	0.5
#32	0.01	0.001	0.4	0.01	0.5
#33	0.01	0.001	0.4	0.01	0.5
#34	0.01	0.001	0.4	0.01	0.5
#35	0.01	0.001	0.4	0.01	0.5

A.3 Measurement Locations

Table A.3. List of all Measurement Points

Number	Location
Point 1	2 inches from start point
Point 2	middle of straightaway 1
Point 3	2 inches from start of radius 1
Point 4	start of radius 1
Point 5	middle of radius 1
Point 6	end of radius 1
Point 7	2 inches from end of radius 1
Point 8	1/3 way straightaway 2
Point 9	2/3 way straightaway 2
Point 10	2 inches from start of radius 2
Point 11	start of radius 2
Point 12	1/3 way of radius 2
Point 13	2/3 way of radius 2
Point 14	end of radius 2
Point 15	2 inches from end of radius 2
Point 16	middle of straightaway 3
Point 17	2 inches from start of radius 3
Point 18	start of radius 3
Point 19	1/4 way radius 3
Point 20	1/2 way radius 3
Point 21	3/4 way radius 3
Point 22	end radius 3

A.4 Bead Width Measurement Raw Data

		Bead Width (inches)					
		1	**2**	**3**	**4**	**5**	**6**
Straightaway 1	**Point 1**	0.322	0.386	0.375	0.364	0.373	0.365
	Point 2	0.463	0.457	0.485	0.437	0.425	0.446
	Point 3	0.545	0.504	0.532	0.51	0.479	0.539
Radius 1	**Point 4**	0.651	0.544	0.633	0.555	0.574	0.63
	Point 5	0.724	0.574	0.665	0.58	0.637	0.645
	Point 6	0.559	0.541	0.564	0.497	0.475	0.605
Straightaway 2	**Point 7**	0.497	0.485	0.488	0.471	0.476	0.525
	Point 8	0.552	0.537	0.531	0.532	0.613	0.534
	Point 9	0.446	0.515	0.481	0.508	0.505	0.469
	Point 10	0.482	0.472	0.442	0.458	0.468	0.436
Radius 2	**Point 11**	0.496	0.51	0.531	0.496	0.497	0.493
	Point 12	0.468	0.485	0.468	0.493	0.497	0.501
	Point 13	0.443	0.451	0.434	0.436	0.46	0.458
	Point 14	0.406	0.401	0.389	0.416	0.402	0.398
Straightaway 3	**Point 15**	0.357	0.426	0.409	0.39	0.404	0.394
	Point 16	0.466	0.389	0.407	0.409	0.387	0.403
	Point 17	0.412	0.388	0.385	0.425	0.402	0.409
Radius 3	**Point 18**	0.401	0.421	0.405	0.416	0.419	0.432
	Point 19	0.421	0.466	0.438	0.461	0.442	0.385
	Point 20	0.43	0.416	0.412	0.436	0.443	0.4
	Point 21	0.395	0.404	0.411	0.431	0.408	0.389
	Point 22	0.411	0.37	0.397	0.415	0.4	0.379

Bead Width (inches)

7	8	9	10	11	12	13	14	15
0.383	0.357	0.309	0.345	0.415	0.31	0.309	0.365	0.302
0.475	0.444	0.425	0.432	0.459	0.425	0.451	0.403	0.324
0.519	0.523	0.491	0.498	0.546	0.455	0.483	0.474	0.465
0.665	0.674	0.648	0.498	0.536	0.658	0.825	0.491	0.499
0.656	0.668	0.593	0.549	0.629	0.476	0.719	0.557	0.584
0.485	0.518	0.463	0.492	0.579	0.437	0.451	0.521	0.534
0.436	0.473	0.452	0.497	0.493	0.398	0.373	0.461	0.495
0.516	0.577	0.526	0.54	0.527	0.485	0.525	0.491	0.473
0.497	0.479	0.479	0.475	0.469	0.475	0.483	0.452	0.5
0.462	0.476	0.469	0.446	0.431	0.423	0.447	0.417	0.473
0.479	0.497	0.502	0.479	0.508	0.471	0.512	0.453	0.522
0.446	0.469	0.434	0.482	0.443	0.427	0.475	0.477	0.455
0.468	0.453	0.442	0.426	0.478	0.416	0.437	0.419	0.435
0.376	0.434	0.394	0.406	0.396	0.379	0.389	0.354	0.406
0.405	0.386	0.379	0.441	0.438	0.346	0.337	0.397	0.387
0.397	0.401	0.415	0.408	0.398	0.375	0.395	0.385	0.387
0.403	0.427	0.405	0.402	0.383	0.371	0.393	0.388	0.419
0.396	0.418	0.396	0.404	0.386	0.396	0.411	0.347	0.387
0.423	0.421	0.424	0.41	0.425	0.409	0.389	0.401	0.406
0.401	0.405	0.395	0.402	0.389	0.373	0.39	0.384	0.441
0.406	0.387	0.392	0.401	0.371	0.351	0.366	0.372	0.376
0.379	0.404	0.388	0.387	0.372	0.372	0.373	0.382	0.352

Bead Width (inches)

16	17	18	19	20	21	22	23	24
0.376	0.37	0.38	0.392	0.391	0.363	0.407	0.368	0.345
0.293	0.457	0.465	0.452	0.433	0.437	0.442	0.438	0.417
0.445	0.529	0.497	0.488	0.485	0.489	0.508	0.465	0.458
0.473	0.547	0.531	0.521	0.513	0.546	0.555	0.501	0.553
0.591	0.619	0.611	0.589	0.573	0.59	0.586	0.578	0.539
0.489	0.529	0.456	0.527	0.502	0.529	0.562	0.478	0.433
0.467	0.513	0.511	0.474	0.523	0.504	0.509	0.47	0.421
0.525	0.539	0.524	0.524	0.514	0.532	0.519	0.482	0.502
0.486	0.48	0.469	0.481	0.466	0.496	0.489	0.458	0.479
0.458	0.429	0.461	0.437	0.447	0.441	0.437	0.466	0.428
0.491	0.444	0.49	0.458	0.474	0.456	0.49	0.44	0.469
0.531	0.457	0.493	0.441	0.462	0.476	0.477	0.477	0.41
0.441	0.399	0.456	0.431	0.431	0.424	0.461	0.421	0.392
0.388	0.384	0.421	0.385	0.415	0.388	0.42	0.39	0.381
0.44	0.402	0.406	0.399	0.389	0.414	0.426	0.409	0.36
0.412	0.378	0.425	0.39	0.386	0.417	0.411	0.378	0.369
0.395	0.363	0.388	0.382	0.39	0.391	0.396	0.378	0.356
0.381	0.374	0.381	0.385	0.382	0.401	0.375	0.38	0.374
0.421	0.419	0.451	0.397	0.376	0.387	0.395	0.396	0.371
0.409	0.406	0.411	0.402	0.384	0.398	0.435	0.397	0.387
0.376	0.391	0.384	0.37	0.373	0.355	0.394	0.343	0.348
0.351	0.362	0.372	0.354	0.387	0.404	0.382	0.356	0.334

Bead Width (inches)

25	26	27	28	29	30	31	32	33
0.345	0.359	0.347	0	0.33	0.363	0.314	0.376	0.348
0.423	0.444	0.426	0.544	0.433	0.46	0.465	0.492	0.402
0.451	0.493	0.481	0.481	0.507	0.512	0.527	0.541	0.466
0.537	0.559	0.554	0.852	0.613	0.681	0.722	0.562	0.563
0.499	0.626	0.601	0.82	0.596	0.565	0.632	0.573	0.491
0.443	0.571	0.437	0.361	0.421	0.506	0.37	0.499	0.452
0.425	0.54	0.421	0.204	0.459	0.482	0.411	0.507	0.467
0.491	0.542	0.501	0.586	0.564	0.528	0.567	0.536	0.544
0.435	0.476	0.471	0.506	0.506	0.466	0.511	0.471	0.519
0.433	0.455	0.462	0.473	0.465	0.472	0.458	0.467	0.482
0.448	0.464	0.455	0.513	0.531	0.485	0.509	0.489	0.518
0.437	0.469	0.451	0.475	0.448	0.465	0.445	0.452	0.486
0.408	0.473	0.431	0.449	0.421	0.433	0.428	0.431	0.438
0.366	0.417	0.429	0.384	0.399	0.41	0.38	0.412	0.447
0.361	0.422	0.382	0.292	0.369	0.377	0.342	0.386	0.412
0.359	0.413	0.415	0.409	0.404	0.414	0.413	0.412	0.416
0.387	0.422	0.445	0.435	0.393	0.389	0.406	0.401	0.406
0.385	0.405	0.432	0.425	0.377	0.402	0.411	0.397	0.421
0.416	0.402	0.425	0.42	0.404	0.392	0.415	0.402	0.402
0.402	0.429	0.409	0.424	0.415	0.382	0.416	0.422	0.443
0.359	0.391	0.364	0.38	0.384	0.354	0.377	0.374	0.374
0.368	0.421	0.407	0.41	0.378	0.375	0.351	0.395	0.392

Bead Width (inches)	
34	35
0.363	0.345
0.45	0.375
0.533	0.467
0.569	0.545
0.547	0.514
0.508	0.473
0.497	0.493
0.544	0.575
0.52	0.489
0.46	0.448
0.489	0.47
0.466	0.451
0.423	0.437
0.384	0.386
0.378	0.384
0.404	0.411
0.393	0.423
0.398	0.401
0.407	0.366
0.411	0.385
0.392	0.379
0.375	0.414

VITA

Alex Roschli was born in Oak Ridge, Tennessee in June of 1993. He graduated from Roane County High School in 2011 before entering the University of Tennessee Knoxville, in Electrical Engineering. Following his freshman year of college, he began research work at Oak Ridge National Laboratory and continued part-time research alongside his studies. Working at ORNL's Manufacturing Demonstration Facility (MDF), he learned about 3D Printing, and in 2013, he began research on the development of large scale 3D Printing. He received a Bachelor's of Science in Electrical Engineering in the spring of 2015. In August of 2015, he began graduate school at the University of Tennessee as part of the five-year BS/MS program and is expected to graduate in spring of 2016.