

October 2013

# Effects of Build Parameters on Additive Materials

Anthony Joseph DeCicco  
*Worcester Polytechnic Institute*

Jessica L. Faust  
*Worcester Polytechnic Institute*

Follow this and additional works at: <https://digitalcommons.wpi.edu/mqp-all>

---

## Repository Citation

DeCicco, A. J., & Faust, J. L. (2013). *Effects of Build Parameters on Additive Materials*. Retrieved from <https://digitalcommons.wpi.edu/mqp-all/3544>

This Unrestricted is brought to you for free and open access by the Major Qualifying Projects at Digital WPI. It has been accepted for inclusion in Major Qualifying Projects (All Years) by an authorized administrator of Digital WPI. For more information, please contact [digitalwpi@wpi.edu](mailto:digitalwpi@wpi.edu).

EFFECTS OF BUILD PARAMETERS ON ADDITIVE MATERIALS

A Major Qualifying Project Report

Submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirements for the

Degree of Bachelor of Science

in Mechanical Engineering

by:

Anthony DeCicco

Jessica Faust

Date: October 30<sup>th</sup>, 2013

Approved By:

---

Professor Nikolaos A. Gatsolis, Advisor  
Mechanical Engineering Department, WPI

---

Derek S. Straub, Affiliate Advisor  
Group 77, MIT Lincoln Laboratory

## Abstract

This project investigates the effects of build parameters on the properties of a thermoplastic used in fused deposition modeling (FDM) technology at MIT Lincoln Laboratory. Dogbone and cubed shaped FDM samples were produced with varying raster angle, build orientation, bead width, and number of plies. Tensile strength experiments are presented and show that unlike typical polymers FDM parts fail due to brittle fracture, while parts built with larger bead-width are more ductile. Structural simulations are presented with the samples considered as orthotropic composite laminates. Thermal expansion experiments show that FDM parts expand 15% less than bulk material. Heat transfer simulations are presented for samples with various raster angle orientations and temperatures between  $-55^{\circ}\text{C}$  to  $85^{\circ}\text{C}$ .

## Acknowledgements

We would like to thank the following people for their support during this project:

Project Advisor	Prof. Nikolaos A. Gatsonis
Lincoln Laboratory Supervisor	Derek Straub
Lincoln Laboratory Supervisor	James Ingraham
WPI Computer Applications Scientist	Prof. Adriana Hera
Lincoln Laboratory Specialist	Robert Longton
Lincoln Laboratory Polymer Technician	Roger Shields
Lincoln Laboratory Polymer Technician	Peter Anderson
Lincoln Laboratory Environmental Technician	Jon Howell
Lincoln Laboratory Electronics Technician	Maureen Torpey

Their assistance during this project helped realize the goals of our Major Qualifying Project (MQP) and its success.

This work is sponsored by the Air Force under Air Force Contract #FA8721-05-C-0002. Opinions, interpretations, conclusions and recommendations are those of the author and are not necessarily endorsed by the United States Government.

## Authorship

Section	Author
<i>Introduction</i>	JF
Overview of Fused Deposition Modeling	JF
Effects of Build Parameters on the Final Part	JF
Building End-Use Parts with FDM at Lincoln Laboratory	AD
Literature Review on ABS used in FDM	AD
Outstanding Issues in FDM	AD
<i>Objectives and Approach</i>	AD, JF
<i>Tensile Testing and Modeling of FDM Manufactured Parts</i>	
Design of Tensile Testing Articles	AD
Stress-Analysis in SolidWorks	AD
Tensile Experimental Set Up	AD, JF
Tensile Testing Results	AD
Theoretical Modeling of FDM Manufactured Parts	AD
<i>Thermal Expansion Testing of FDM Manufactured Parts</i>	
Design of Thermal Expansion Test Articles	AD
Strain Gauge Decision for Thermal Expansion	JF
Thermal Expansion Test	JF, AD
Thermal Expansion Test Results	JF
<i>Summary, Conclusions, and Recommendations for Future Work</i>	
Tensile Testing and Modeling	AD
Thermal Testing and Modeling	JF
Benefits to MITLL and Recommendations for Future Work	AD, JF



## Contents

Contents .....	5
Table of Figures .....	8
1 Introduction .....	13
1.1 Overview of Fused Deposition Modeling .....	14
1.1.1 Effects of Build Parameters on the Final Part .....	17
1.1.2 Building End-Use Parts with FDM at Lincoln Laboratory .....	20
1.2 Literature Review on ABS used in FDM .....	21
1.2.1 Outstanding Issues in FDM .....	23
1.3 Objectives and Approach .....	23
2 Tensile Testing and Modeling of FDM Manufactured Parts .....	28
2.1 Design of Tensile Testing Articles .....	28
2.1.1 Geometric Design Guidelines and Limitations for FDM Parts .....	28
2.1.2 Stratasys Insight Software in Fortus FDM Machines .....	29
2.1.3 Stratasys Control Center .....	31
2.1.4 Deciding on the FDM parameters to test .....	31
2.2 Stress-Analysis in SolidWorks .....	34
2.3 Tensile Experimental Set Up .....	36
2.3.1 The Wheatstone Bridge .....	38
2.3.2 Tensile Test .....	39
2.4 Tensile Testing Results .....	42
2.4.1 Mohr's Circle for Principle Strains and Principle Stresses. ....	42
2.4.2 Y-axis Oriented Dogbones .....	44
2.4.3 Z-Axis Oriented Dogbones .....	51

2.4.4	Number of Plies for 45/-45 and Quasi Isotropic.....	58
2.4.5	Bead Width Alterations.....	67
2.4.6	Evaluation of the Ultimate Tensile Strength.....	71
2.4.7	Summary of Tensile Material Properties for Tested Dogbones .....	72
2.5	Theoretical Modeling of FDM Manufactured Parts.....	74
2.5.1	Orientation and its Effect on Stacking .....	74
2.5.2	Stress-Strain Mathematics.....	81
2.5.3	ANSYS Modeling of Tensile Tests.....	83
3	Thermal Expansion Testing of FDM Manufactured Parts .....	90
3.1	Design of Thermal Expansion Test Articles .....	90
3.2	Strain Gauge Decision for Thermal Expansion .....	90
3.3	Thermal Expansion Test .....	92
3.4	Thermal Expansion Test Results.....	94
3.4.1	Experimental Results for Thermal Expansion Coefficient .....	94
3.4.2	COMSOL Modeling of Thermal Expansion Tests .....	98
4	Summary, Conclusions and Recommendations for Future Work.....	106
4.1	Tensile Testing and Modeling .....	106
4.2	Thermal Testing and Modeling .....	107
4.3	Benefits to MITLL and Recommendations for Future Work .....	107
5	References.....	109
6	Appendix.....	112
6.1	Appendix A- Design and Manufacturing .....	112
6.2	Appendix B- Comparing Tensile Results.....	124
6.3	Appendix C- Thermal Test Set-up and Schematics .....	131



6.4	Appendix B- Matlab functions.....	132
6.4.1	Stress-Strain Plot Function.....	133
6.4.2	Mohr's Circle Plot Function.....	133
6.4.3	Plotting Data for 45/-45 Z-axis.....	134
6.4.4	Plotting Data for 45/-45 Y-Axis .....	136
6.4.5	Plotting Data for 0/90 Y-axis.....	139
6.4.6	Plotting Data for 45/-45 n = 1 .....	142
6.4.7	Plotting data for 45/-45 n = 6 .....	144
6.4.8	Plotting Data for 45/-45 Large Bead .....	146
6.4.9	Plotting Data for Quasi Isotropic Y-axis .....	147
6.4.10	Plotting Data for Quasi Isotropic n = 1.....	150
6.4.11	Plotting Data for Quasi Isotropic n = 6.....	153
6.4.12	Plotting Data for Quasi Isotropic Large Bead.....	155
6.4.13	Plotting Data for Quasi Isotropic Z-axis .....	158
6.4.14	Plotting data for 0/90 Z-axis .....	160
6.4.15	Determining the Stress and Strain Relationships in a Quasi-Isotropic Laminate .	161
6.4.16	Determining the stress and strain relationships in a 0/90 Ply Laminate.....	172
6.4.17	Determining the stress and strain relationships in a 45/-45 Laminate .....	178
6.4.18	Sub-functions for Stress and Strain Analysis .....	185
6.4.19	Plotting Functions used for Stress and Strain Analysis.....	185
6.4.20	To determine the Coefficient of Thermal Expansion for Orthotropic Laminates, it uses Mathematics presented in di Scalea .....	189
6.4.21	Plotting Functions used for Heat Expansion.....	192

## Table of Figures

Figure 1-1 Main components of an FDM machine. Image provided by (FDM, n.d.).....	15
Figure 1-2 Heating block and tip assembly on FDM machine. Image provided by (Kouhi et al, 2008) .....	16
Figure 1-3 Diagram of different parameters of printing with FDM. Image provided by (Kalita et al, 2003) .....	18
Figure 1-4 Various raster angle orientations .....	19
Figure 1-5 Differences in density options on FDM system.....	19
Figure 1-6 Example of different build orientations .....	20
Figure 1-7-Tensile Stress-Stain Example (ASTM 3039, 2008).....	25
Figure 1-8-Material Necking and Failure (Mechanical Properties, n.d.) .....	26
Figure 1-9-Thermal Expansion Graphical Relation (ASTM E831, 2012).....	27
Figure 2-1 Custom Groups in Insight .....	30
Figure 2-2 Layering Technique in Insight .....	30
Figure 2-3- Tensile Test Orientation .....	32
Figure 2-4 Regular Dogbone (Left) Modified Dogbone (Right).....	34
Figure 2-5 Constraint and Stress Set-up Regular Dogbone (Left) Modified Dogbone (Right).....	35
Figure 2-6 Results of SolidWorks Express for Modified Dogbone (Left) and Regular Dogbone (Right).....	36
Figure 2-7 Diagram showing location of strain gauges on tensile and thermal samples.....	37
Figure 2-8 One-Quarter Wheatstone Bridge (Strain Gauge Measurement, n.d.) .....	38
Figure 2-9 Three Wire Variation for Quarter-bridge (Strain Gauge Measurement, n.d.) .....	39
Figure 2-10 NI 9945 Quarter Bridge Completion (NI 9944, NI 9945, n.d.) .....	40
Figure 2-11 Tensile Test Set-up: 1. Quarter Bridge Completion, 2. Test Section, 3. Moving Instron Clamp, 4. Stationary Instron Clamp, 5. Shims .....	41
Figure 2-12 Rosette Element Coordinate (Strain Gage Rosettes, 2010) .....	42
Figure 2-13 Mohr's Circle for Strain (Strain Gage Rosettes, 2010).....	43
Figure 2-14 Stress-Strain Curve for 45/-45 Raster Y-Axis Orientation.....	45

Figure 2-15 45/-45 Y-Axis Oriented Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)..... 45

Figure 2-16 45/-45 Raster Y-Axis Orientation Mohr's Circle at Material Failure..... 46

Figure 2-17 Stress-Strain Curve for 0/90 Raster Y-Axis Orientation..... 47

Figure 2-18 0/90 Y-Axis Oriented Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)..... 47

Figure 2-19 0/90 Raster Y-Axis Orientation Mohr's Circle at Material Failure ..... 48

Figure 2-20 Stress-Strain Curve for Quasi Isotropic Raster Y-Axis Orientation ..... 49

Figure 2-21 Quasi Isotropic Y-Axis Oriented Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right) ..... 50

Figure 2-22 Quasi Isotropic Raster Y-Axis Orientation Mohr's Circle at Material Failure ..... 50

Figure 2-23 Stress-Strain Curve for 45/-45 Raster Z-Axis Orientation..... 52

Figure 2-24 45/-45 Z-Axis Oriented Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)..... 53

Figure 2-25 45/-45 Raster Z-Axis Orientation Mohr's Circle at Material Failure..... 53

Figure 2-26 Stress-Strain Curve for 0/90 Raster Z-Axis Orientation..... 54

Figure 2-27 0/90 Z-Axis Oriented Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)..... 55

Figure 2-28 Stress-Strain Curve for Quasi Isotropic Raster Z-Axis Orientation ..... 56

Figure 2-29 Quasi Isotropic Z-Axis Oriented Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right) ..... 57

Figure 2-30 Quasi Isotropic Raster Z-Axis Orientation Mohr's Circle at Material Failure ..... 57

Figure 2-31 Stress-Strain Curve for 45/-45 Raster  $n = 1$  ..... 59

Figure 2-32 45/-45  $n = 1$  Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)..... 60

Figure 2-33 45/-45  $n = 1$  Mohr's Circle at Material Failure ..... 60

Figure 2-34 Stress-Strain Curve for 45/-45 Raster  $n = 6$  ..... 61

Figure 2-35 45/-45  $n = 6$  Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)..... 62

Figure 2-36 Stress-Strain Curve for Quasi Isotropic Raster n = 1..... 63

Figure 2-37 Quasi Isotropic n = 1 Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)..... 63

Figure 2-38 Quasi Isotropic n = 1 Mohr's Circle at Material Failure ..... 64

Figure 2-39 Stress-Strain Curve for Quasi Isotropic Raster n = 6..... 65

Figure 2-40 Quasi Isotropic n = 6 Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)..... 66

Figure 2-41 Quasi Isotropic n = 6 Mohr's Circle at Material Failure ..... 66

Figure 2-42 Stress-Strain Curve for 45/-45 raster with large bead ..... 68

Figure 2-43 45/-45 Large Bead Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)..... 68

Figure 2-44 Stress-Strain Curve for Quasi Isotropic raster with Large Bead ..... 70

Figure 2-45 Quasi Isotropic Large Bead Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right) ..... 70

Figure 2-46 Quasi Isotropic Large Bead Mohr's Circle at Material Failure ..... 71

Figure 2-47 Unidirectional, No flattening ..... 75

Figure 2-48 Coordinate set-up for 45/-45 raster ..... 75

Figure 2-49 Midplane cross-section view 45/-45 ..... 76

Figure 2-50 Stacking at half cycle for 45/-45 ..... 76

Figure 2-51 Stacked, no flattening..... 76

Figure 2-52- Unidirectional and Flattening..... 77

Figure 2-53 Stacked and Flattening ..... 77

Figure 2-54 Unidirectional (red, left) & Stacked (blue, right)..... 78

Figure 2-55 Stacked, area reduction without flattening ..... 78

Figure 2-56 Z'-Y plane Bottom Layer (Black) Top Layer (Red) ..... 79

Figure 2-57 Orthogonal, no flattening ..... 80

Figure 2-58 Orthogonal and flattening ..... 80

Figure 2-59 Fixture and Force Set-up for ANSYS ..... 86

Figure 2-60 ANSYS Orthotropic Analysis for 45/-45 Dogbone using Tensile Data ..... 87

Figure 2-61 ANSYS Orthotropic Composite Laminate Analysis for 45/-45 Dogbone using Tensile Data ..... 88

Figure 2-62 ANSYS Orthotropic Composite Laminate Analysis with True Density for 45/-45 Dogbone using Tensile Data ..... 89

Figure 3-1 Strain gauges on FDM active block and reference block ..... 91

Figure 3-2: Image of thermal test set-up ..... 93

Figure 3-3: CTE of [45/-45] block in X direction ..... 95

Figure 3-4: CTE of [45/-45] block in the Z direction ..... 95

Figure 3-5: CTE of [0/90] block in the XY direction ..... 96

Figure 3-6: CTE of [0/90] block in the Z direction ..... 96

Figure 3-7: CTE of quasi-isotropic block in the X direction (Gauge 1) ..... 97

Figure 3-8: CTE of quasi-isotropic block in the X direction (Gauge 3) ..... 97

Figure 3-9: CTE of quasi-isotropic block in the Z direction ..... 98

Figure 3-10 - Air circulation diagram in thermal oven. Picture provided by Thermal Product Solutions ..... 99

Figure 3-11 - Simplified model geometry for Comsol analysis ..... 100

Figure 3-12 - Geometry of block modeled in Comsol Multiphysics ..... 101

Figure 3-13 - Diagram of heat transfer analysis ..... 102

Figure 3-14 - Modeling results for conductive heat transfer through samples ..... 102

Figure 3-15 – Isothermal contours for heat transfer through sample ..... 103

Figure 3-16 - Total deformation in the Y direction for an isotropic sample ..... 104

Figure 3-17 - The total deformation model in X direction for isotropic sample ..... 104

Figure 6-1- Printing Arrangement ..... 112

Figure 6-2-Thermal Geometry for 0/90 and 45/-45 ..... 115

Figure 6-3- Thermal Geometry for Quasi Isotropic ..... 116

Figure 6-4- Geometry for 45/45 Dogbone ..... 117

Figure 6-5- Geometry for Thin Specimen of 45/-45 Dogbone ..... 118

Figure 6-6- Geometry for Thick Specimen of 45/-45 Dogbone ..... 119

Figure 6-7- Geometry for Quasi Isotropic Dogbone Specimen ..... 120

Figure 6-8- Geometry for Thin Quasi Isotropic Dogbone Specimen .....	121
Figure 6-9-Geometry for Thick Quasi Isotropic Dogbone Specimen .....	122
Figure 6-10 Modified Dogbone .....	123
Figure 6-11 Comparing three rasters for Y-Axis Dogbone .....	124
Figure 6-12 Comparing 45/-45 Y-axis with Large Bead (Disclaimer: Strain in Plastic region for n=6 is a result of slip) .....	125
Figure 6-13 Comparing Quasi Isotropic Y-axis with Large Bead .....	126
Figure 6-14 Comparing three rasters for Z-axis Dogbones.....	127
Figure 6-15 Comparing Ply Thickness for 45/-45.....	128
Figure 6-16 Comparing Ply Thickness for Quasi Isotropic .....	129
Figure 6-17 0 90 Y-Axis Dogbone at UTS displaying Max Normal Strain .....	130
Figure 6-18 Thermal Block with Labeled Wires and Gauges .....	131
Figure 6-19 Half Bridge for Gauge 1 Schematic (Modeled using NI) .....	131
Figure 6-20 Half Bridge for Gauge 2 Schematic (Modeled using NI) .....	132
Figure 6-21 Half Bridge for Gauge 3 Schematic (Modeled using NI) .....	132

## 1 Introduction

Fused Deposition Modeling (FDM), an extrusion based process that builds a three dimensional part in layers of plastic, is a relatively new process to the world of additive manufacturing. It is most commonly used for quickly building prototypes to test the fit and function of an end use part before manufacturing it in more conventional methods. As the technology improves, there is a growing interest in using FDM for building end-use parts. However, little is known of how the final products material properties will behave when compared to the bulk material properties. In order to design end-use parts with FDM, understanding how the build parameters affect the final product must be known.

FDM technology uses liquefied thermoplastics extruded through a heated nozzle on to a platform to build a part layer by layer. A typical FDM system is shown in Figure 1-1. It first deposits the outline of the part, and then fills in the center before moving up to the next layer. FDM technology is virtually self-sufficient and can greatly reduce the time it takes to build a part for manufacturers. Due to the layered design, it is also very flexible in building multiple parts that interact together. For example, an FDM printer can build an adjustable wrench as one complete part instead of two separate pieces requiring post assembly when using conventional machining. Due to these reasons, FDM technology has become very popular in the manufacturing industry.

The FDM technology is also very flexible. There are many different build parameters that the operator can choose to change while building the part. For example, the operator can choose to fill the center of the part completely or to leave spaces between each extruded bead to save time during the build process and lower the final weight. However, if there are spaces between the extruded beads, they will not bond together as strongly and result in a final part that lacks in strength. All of the build parameters available on the FDM will change the final products' material properties in such a way as this. When building prototypes using FDM, the final material properties are not as crucial. These parts are typically used for modeling purposes and are temporary pieces that are discarded after their use. Knowing the final material properties of

an FDM printed part becomes important when building end-use parts, a process that has just begun to be investigated by manufacturers.

Lincoln Laboratory is interested in FDM technology to build unmanned aerial vehicles (UAV's). The majority of the parts on these UAV's are printed using a FDM printer, and Lincoln Laboratory hopes to use these parts as the final product. It is unknown how these parts will react when exposed to environmental conditions or to stress in the part due to the small knowledge of the material properties of FDM printed parts. The objective of this project is to investigate the specific effects of build parameters on the final product in order to provide Lincoln Laboratory with a better idea of the material properties of the parts used for the UAV's.

## **1.1 Overview of Fused Deposition Modeling**

This section will provide a brief overview of the FDM process including the main terminology used for the build parameters in additive manufacturing. A brief literature overview is also presented focusing on analytical and experimental studies on the effects of build parameters on material properties.

### *Principles of FDM*

Fused deposition modeling (FDM) is a rapid prototyping process that utilizes the extrusion method to build a product in layers. It liquefies thermoplastics fed from a heating block into a small tip to build each layer of the part, as well as structures to support the more complicated geometry. The main components of a FDM system are the build material spools, the build platform, and the extrusion head including the heated liquefiers and extrusion nozzles. These components can be seen in Figure 1-1.



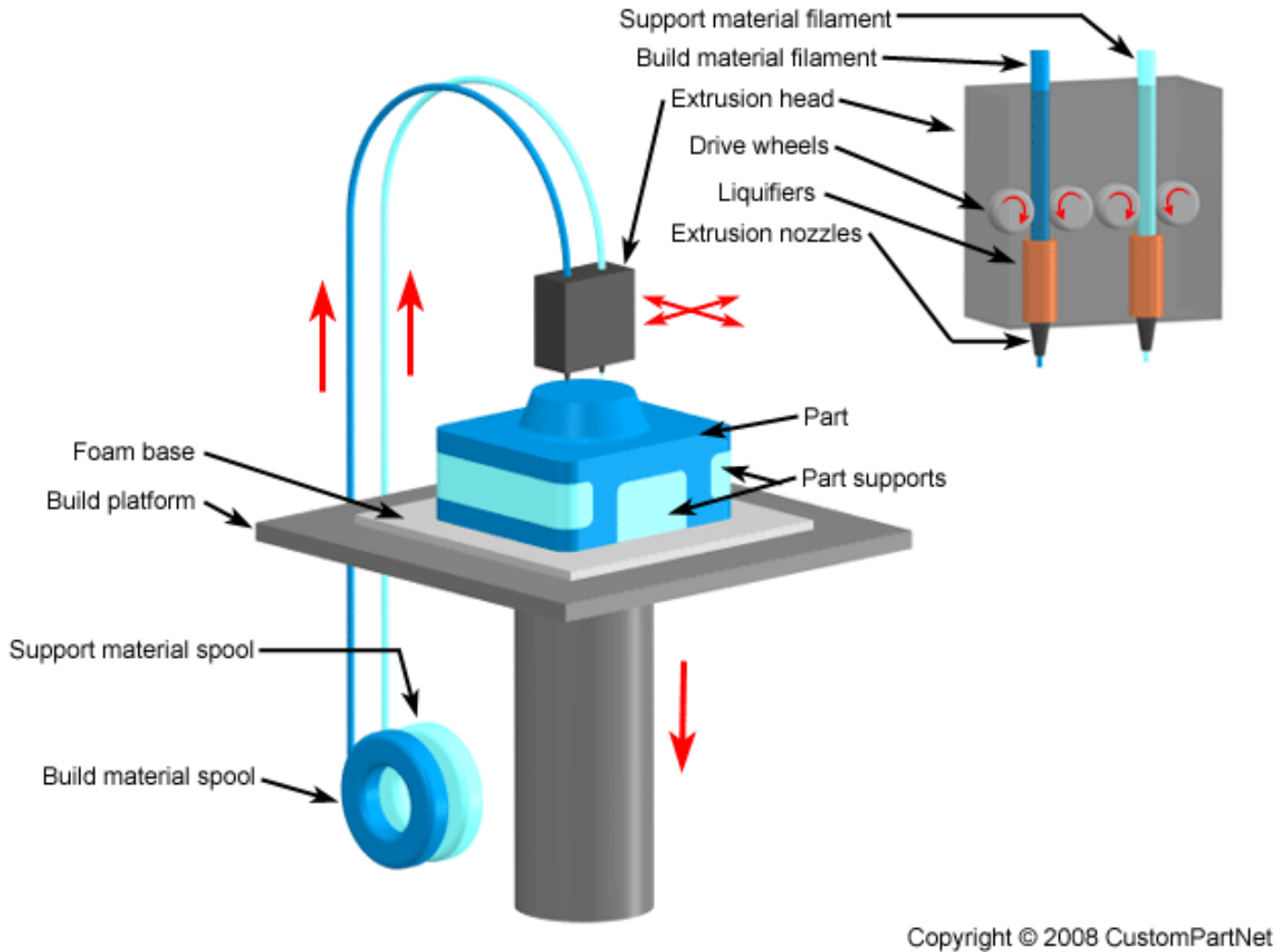


Figure 1-1 Main components of an FDM machine. Image provided by (FDM, n.d.)

The FDM uses various thermoplastics such as ABS, ABS-M30, PC-ABS, and ULTEM 9085 [1]. ABS-M30 (Acrylonitrile/butadiene/styrene) is a common plastic used in FDM due to its well-rounded range of material properties and relatively low cost. The plastic is fed into the heating block by a spool at a specific feed rate and the plastic is liquefied, as shown in Figure 1-2. The plastic is kept at a low enough temperature to keep it liquid without over heating it and causing the plastic to degrade. The plastic then is pushed through the extrusion tip from the pressure of the feed rate from the spool and deposited on the part.

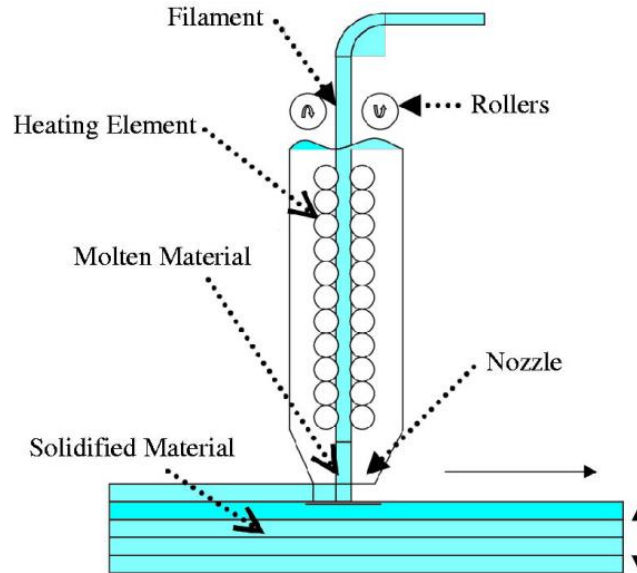


Figure 1-2 Heating block and tip assembly on FDM machine. Image provided by (Kouhi et al, 2008)

The tip moves across the part in the x and y direction at a defined speed based on the feed rate to create the geometry. The tip size will change the feed rate of the flowing plastic as well as the rate the tip moves across the part. Larger tips will require a faster feed rate and therefore a faster build time, however they do not provide as much detail and mechanical stability as the smaller tips produce. The tip size also defines the minimum feature that is allowed on a part.

The part is first defined by depositing plastic on the outer diameter creating an outline of the part and then the inside is filled in. The outline is created at a slower speed in order to have more control over the plastic and improve the surface accuracy on the final part. The inside of the part is filled in using a faster speed and the outline serves as a wall to contain the material flow when rapid movements occur at the edges. Once the layer is complete, the part is moved down in the vertical direction to allow the extrusion head to make the next layer.

The extruded plastic must be kept at a temperature high enough that allows it to bond with previously deposited material, but not so hot that it overheats the deposited layers causing a misshaped part. If it is too cool, the plastic may adhere to the other layers but create boundaries that cause issues with the mechanical properties in the final part.

As the part is built, it may require supports for any overhang of more than 45°. If a support is needed, the plotting head switches to a secondary plotting head with a different material to build the support. The support material can either be made of weaker plastic that is broken away or with water soluble material that can be dissolved in a bath after the part is made. Unfortunately, adding support structures can be a time consuming process and increase build time.

### 1.1.1 Effects of Build Parameters on the Final Part

The FDM process begins in the form of a CAD file which is imported into software as an STL file and sliced into thin layers. The software then determines the fill path for the inside of the part on each layer as well as any supports that are needed for the part. Careful consideration should be taken to design the part in such a way that minimal supports are needed in order to keep build time at a minimum. At this point, the user has various options for how to build the part by changing the build parameters of the FDM machine. These build parameters include *the layer thickness, the raster angle and width, density, and the air gap*. The following explanations will go into detail of each of these parameters and how they affect the final part.

***Ply thickness and bead width:*** Ply (layer) thickness and bead (raster) width is based on the chosen tip size and diameter of the extruded plastic. Each tip size has a range of bead widths available and to increase the speed of the build, a larger tip size should be chosen and the plastic extruded on the lower end of the bead width (*Tip Selection*). The tip size can depend on the material being used for the print as well as the user deciding between speed and accuracy. Ply thickness and bead width are shown in Figure 1-3.

***Air gap:*** The air gap is the space between each extruded bead of the plastic. The default setting on the FDM machine is to set it at zero, meaning that the beads will just touch. This can also be set to a positive value that will leave space between each bead and form a less dense product that takes less time to build. Alternatively, the air gap can be set to a negative value causing the beads to overlap forming a dense part with slow build time. If the beads are overlapping, the plastic will form a close bond with the previous bead and therefore have less

gaps and stronger mechanical properties, however the dense structure and long build time is something to consider for each application. The air gap is shown in Figure 1-3.

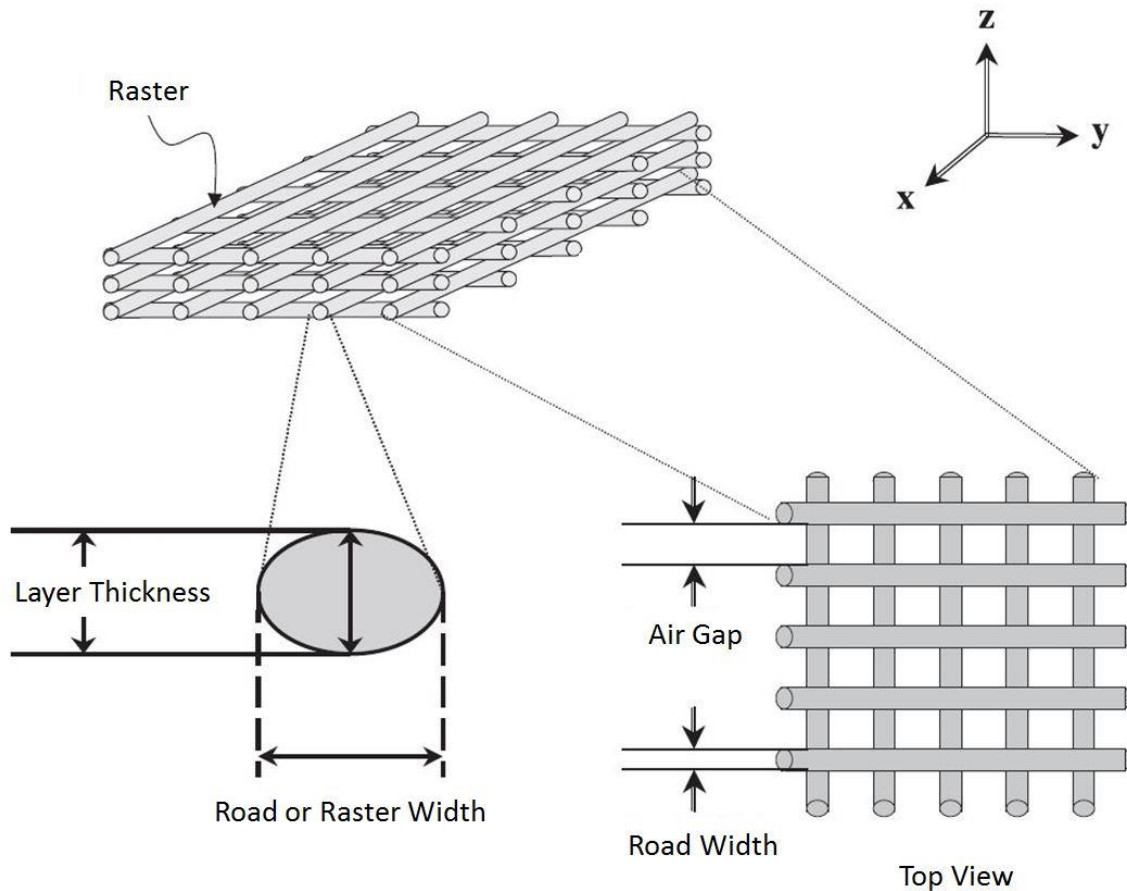


Figure 1-3 Diagram of different parameters of printing with FDM. Image provided by (Kalita et al, 2003)

**Raster angle:** The raster angle is the direction in which the beads are printed with respect to the orientation of the part. The standard raster angle orientation for the printer to use is to alternate between  $45^\circ$  and  $-45^\circ$  on each ply. However, the user has control over the raster angle before printing and can choose to use a different orientation set such as the options shown in Figure 1-4. Different raster angle orientations will have an effect on the mechanical properties of the part, including helping it to become more isotropic. The quasi-isotropic raster angle orientation is expected to perform the best.

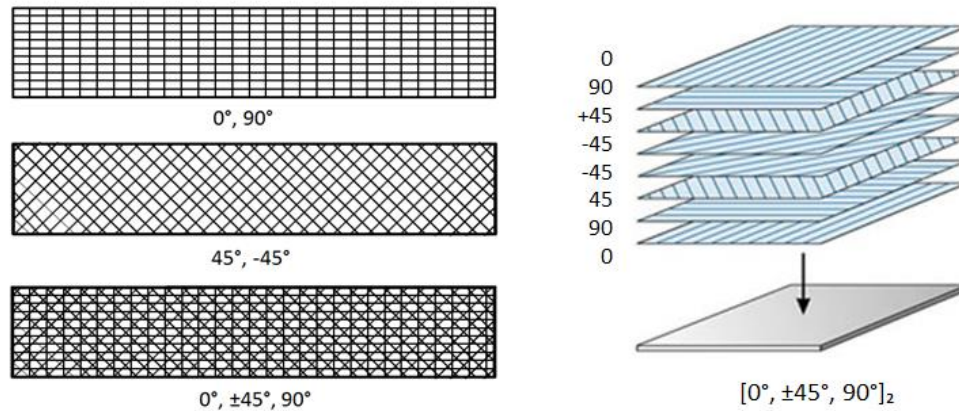


Figure 1-4 Various raster angle orientations

**Density:** The FDM has three different options for densities; sparse, sparse double-dense, and dense. These options are shown in Figure 1-5. Density is a parameter to consider based on the application of the part. Sparse filled parts print much faster than dense parts, however they lack the strength. The density of the part can also be manually changed by changing the parameters of each ply to alternate between the density options.

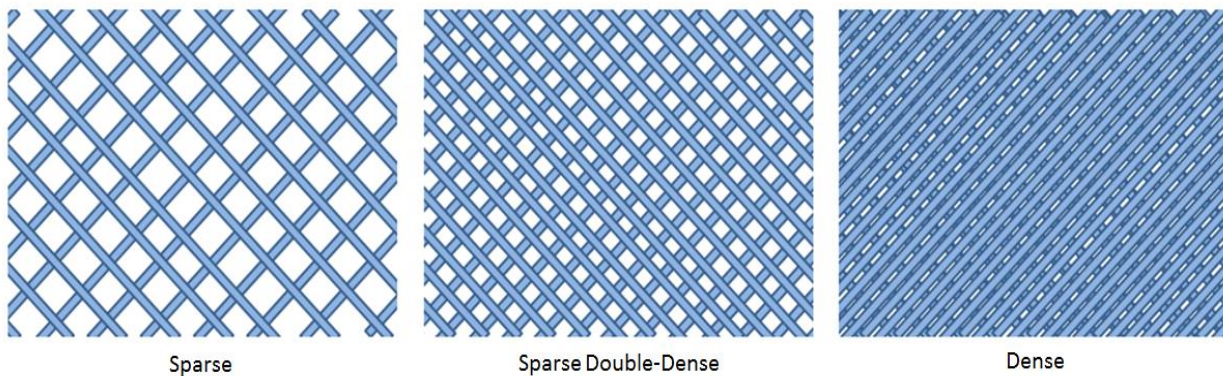


Figure 1-5 Differences in density options on FDM system

**Build orientation:** Build orientation is an important parameter to consider for build time and material properties when printing. The direction your part is orientated will change the way the machine prints the part as well as the support material. A simple example of this is to look at how the FDM system would print a standard table. The table in Figure 1-6 (A) would require a large amount of material to support the top and the table in (B) would require support material under the legs of the table. However, if the table is orientated upside down, as in (C), the FDM

machine is able to print the entire table without switching to the support material once. Using this type of planning can save a considerable amount of build time.

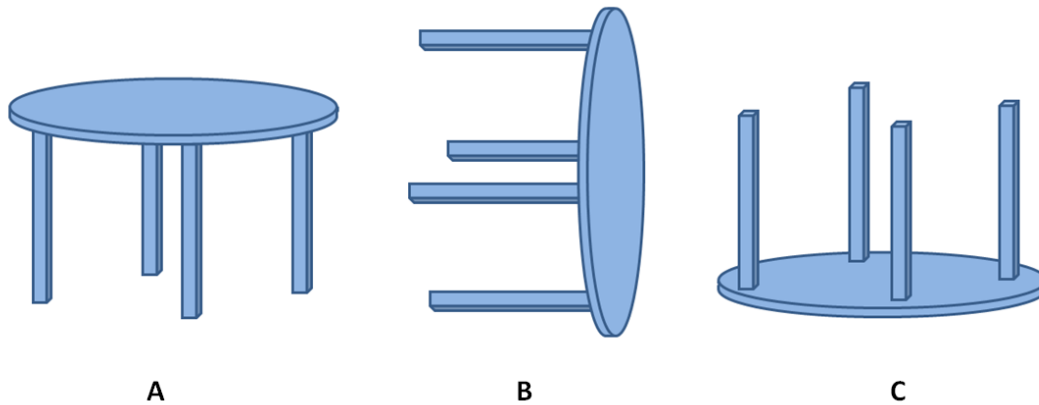


Figure 1-6 Example of different build orientations

The build time also comes into play when thinking about how the FDM system prints the parts. FDM prints parts in the vertical direction layer by layer. This causes the parts to be vulnerable to tensile forces in the vertical direction due to de-lamination. The build orientation is a parameter to consider when printing end-use parts that experience tensile forces in order to avoid this vulnerability.

### 1.1.2 Building End-Use Parts with FDM at Lincoln Laboratory

The Rapid Prototyping Group at Lincoln Laboratory has actively been using FDM in order to make end-use products. It has allowed the group greater capabilities in producing one-off products for testing and field use. Their interest is in determining any boundaries that this technology has so informed decisions can be made into using FDM in the future. Example products that engineers at Lincoln Laboratory have created using an FDM include experimental UAVs, adjustable wrenches, miniature bridges for intergroup competitions, and test beds for hardware.

Mechanical properties are typically reported in what is known as “bulk material properties” where specimens are specially prepared to reduce any variables other than the material for destructive and non-destructive tests. This provides the baseline for material selection for end-use products. For example: if engineers determine that the loading on a member is  $50\text{N/m}^2$  but

the material originally chosen starts to permanently deform at  $30\text{N/m}^2$ , then they will seek out another material that will not have a risk of failing in its application. Only recently has the Laboratory been conducting experiments to determine the mechanical properties of FDM processed thermoplastics.

## 1.2 Literature Review on ABS used in FDM

Acrylonitrile butadiene styrene (ABS) is a thermoplastic used in the 3D printer Stratasys Fortus 400; its variant, ABS-M30 made by Stratasys, is the material that our group will use within this project. Thermoplastics in general are flexible due to the nature of polymer chains being only weakly cross-linked and thus can be untangled by applying a force. In the thermal expansion testing, ABS will be maintained at temperatures below its glass transition temperature (ABS-M30, n.d.) (the temperature at which thermoplastics transition from a crystalline glassy-state to that of a rubbery state). It is expected that ABS should retain its general shape as a result of this environmental condition. In terms of the tensile testing, ABS should exhibit linear elastic deformation, non-linear elastic deformation, and a relatively long period of plastic deformation where necking will be evident. (Askeland et al., 2011)

Analysis of FDM processed ABS is atypical of other manufacturing processes associated with polymers. Generally, processes such as extrusion, ejection molding, and compression molding nullify most air gaps between mers (the fundamental repeating unit for the characteristic arrangement of a polymer). The chain networks of these mers also tend to align in the direction of an applied tensile force. The nature of FDM processing is different from these other processes and thus creates a lamina structure typical to composites. However, composites are at least two component structures with fibers coupled with a background matrix material typically designed to enhance certain material properties. Multi-polymer FDM is outside the scope of this project and we may not solely rely on the canonical analysis for laminar composites. Literature has investigated this issue of analysis and Li et al. (2002) developed a theoretical estimation of material properties from an applied stress. They addressed the issue of voids in the FDM parts and how it can alter the material properties. Image data was taken to observe the mesostructure due to preset gaps and determine void density within the material.

With this data comparison between theoretical and experimental work was possible for tension tests. The fundamental mathematics from this study will be used in our analysis to look at such material properties as Young's Modulus and Poisson's Ratio.

Mechanical properties as dependent on a variety of factors in FDM processed ABS products is an active research area. Raster angle is a very important consideration and in Fatimatuzahraa et al. (2011) they use this factor to compare tensile strength, deflection, flexural strain, and impact strength. The 45/-45 sample outperformed the 0/90 in three of the four tests. In the tensile force test the 0/90 performed better as it was aligned with the force. However, according to the presented data there was only a 1.2% decrease in Tensile strength from the 0/90 test sample to the 45/-45 one. The deflection test demonstrated that the 45/-45 sample had a higher elasticity than the 0/90 which the authors attribute to the fact that the roads are, "not parallel to the load applied at the center." In a flexure test, the 45/-45 was able to withstand great stress and exhibited flexural strains greater than the 0/90 where the authors propose that a main contributing factor is the stacking of the 45/-45 versus the 0/90. The air gap between roads is greater within the 0/90 thus contributing to lower mechanical properties as a result of stress concentrations.

Insight into the difficulties of testing FDM parts has also been presented in literature. Ahn et al. (2002) encountered problems in premature failure for their test samples in tension and compression. Modifications to ASTM standards were needed as the regular "dog-bone" structure encountered stress concentrations at the radii. For reference, the FDM test articles were compared to injection molded copies. The authors repeated their destructive tensile and compressive tests to determine error. From their research, Ahn et al. (2002) established building rules for engineers who are designing for tensile and/or compressive loading. As the air gap is decreased between the roads, the authors show that there is a significant increase in strength and stiffness. They also warned against transitions such as radiused corners citing the premature failing of the "dog-bone" sample and advise that engineers take special care in programming the printing process when working with such geometries.



### 1.2.1 Outstanding Issues in FDM

From the literature search conducted there are outstanding issues in knowledge as to the full scope of material properties as dependent upon a wide set of variables. One notable missing raster orientation is the Cross-Plied Quasi Isotropic which has the greatest potential for exhibiting the reported bulk material properties of ABS-M30. Tests in blending different densities and tool-paths to strengthen certain areas of a 3D printed part are also missing in literature which may be due to the intellectual property value it has for companies that specialize in 3D printing. Literature widely cites that end-use FDM products are possible and are being used. In relation to the Rapid Prototyping Group at Lincoln Laboratory, knowing the thermal stability of ABS-M30 in different raster orientations will boost confidence as to the integrity of their Unmanned Aerial Vehicles being used in the field for a variety of climates and altitudes. This study will seek to develop a catalog of several parameters and their effect in the strength of FDM processed ABS.

### 1.3 Objectives and Approach

The Unmanned Aerial Vehicle (UAV) project in Group 77 at Lincoln Laboratory is seeking to improve the strength of their FDM printed UAV while reducing the weight of the structure. Different components of the UAV require certain strengths: the internal wing spar structure requires a resistance to shear; the fuselage should maximize the possible payload weight; the entire aircraft should be thermally resistive to avoid excessive expansion or contraction in varied climates. Therefore, the tests that we are most concerned with in this project are tensile and thermal expansion. In order to maximize useful data provided to Lincoln Laboratory within the timeframe given, the following objectives were determined:

- Investigate tensile material properties for FDM parts with different build parameters
  - Use Instron 8801 to test dogbones as defined by ASTM D638 with rosette strain gauge to characterize strain state
- Compare thermal expansion coefficients between FDM parts and bulk properties
  - Use thermal chamber with thermocouple and strain gauge instrumentation to record data from thermal cycle.

- Model thermal expansion, heat transfer, and strain at ultimate tensile strength load
  - Create 2D and 3D model in Comsol for thermal modeling and use ANSYS Workbench to model dogbones in tension.
- Provide recommendations in build parameters for Unmanned Aerial Vehicle (UAV) structure
  - Compile data using Matlab and draw conclusions from observed trends in build parameter alteration.

The following section explains the breakdown for the tensile and thermal tests and the motivation behind why these two tests were used to compare build parameters to the material properties.

### *Tensile Tests*

Tensile tests provide useful data on the material properties of the ABS test specimens. Paramount to information on failure characteristics is the Ultimate Tensile Strength,  $\sigma_{UTS}$ . This value (denoted as 'Ultimate' in Figure 1-7) represents the value of stress at which the material exhibits the onset of necking (Figure 1-8) which quickly leads to failure of the specimen. This information provides a stress boundary at which design loads must be below. The Ultimate Tensile Strain,  $\epsilon_{UTS}$ , provides engineers with data on the amount of elongation that a material can maintain before failure. The higher the value, the more ductile or 'stretchable' the material is before breaking. Represented by the slope triangle in Figure 1-7, the Tensile Chord Modulus of Elasticity,  $E$ , looks at the linear portion or Hookean region of the stress-strain curve to determine the stress to strain ratio. The higher the ratio, the greater the required stress is to induce a strain. With this value, engineers have a predicative capability to determine the amount of strain from a given tensile stress in the elastic region or vice-versa. The transition strain,  $\epsilon_T$ , shown in Figure 1-7 as the transition point, is the portion of the stress-strain curve where the material may change slope meaning that  $E$  is no longer representative of the material response to an induced stress. Lastly in the tensile test, Poisson's Ratio,  $\nu$ , represents

the negative ratio of lateral strain to longitudinal strain measured from orthogonal strain gauges<sup>1</sup>. The application of this ratio informs engineers as to the cross-sectional area decrease in tension or cross-sectional area increase in compression that the material will exhibit. The experiments in this study will closely follow ASTM D638.

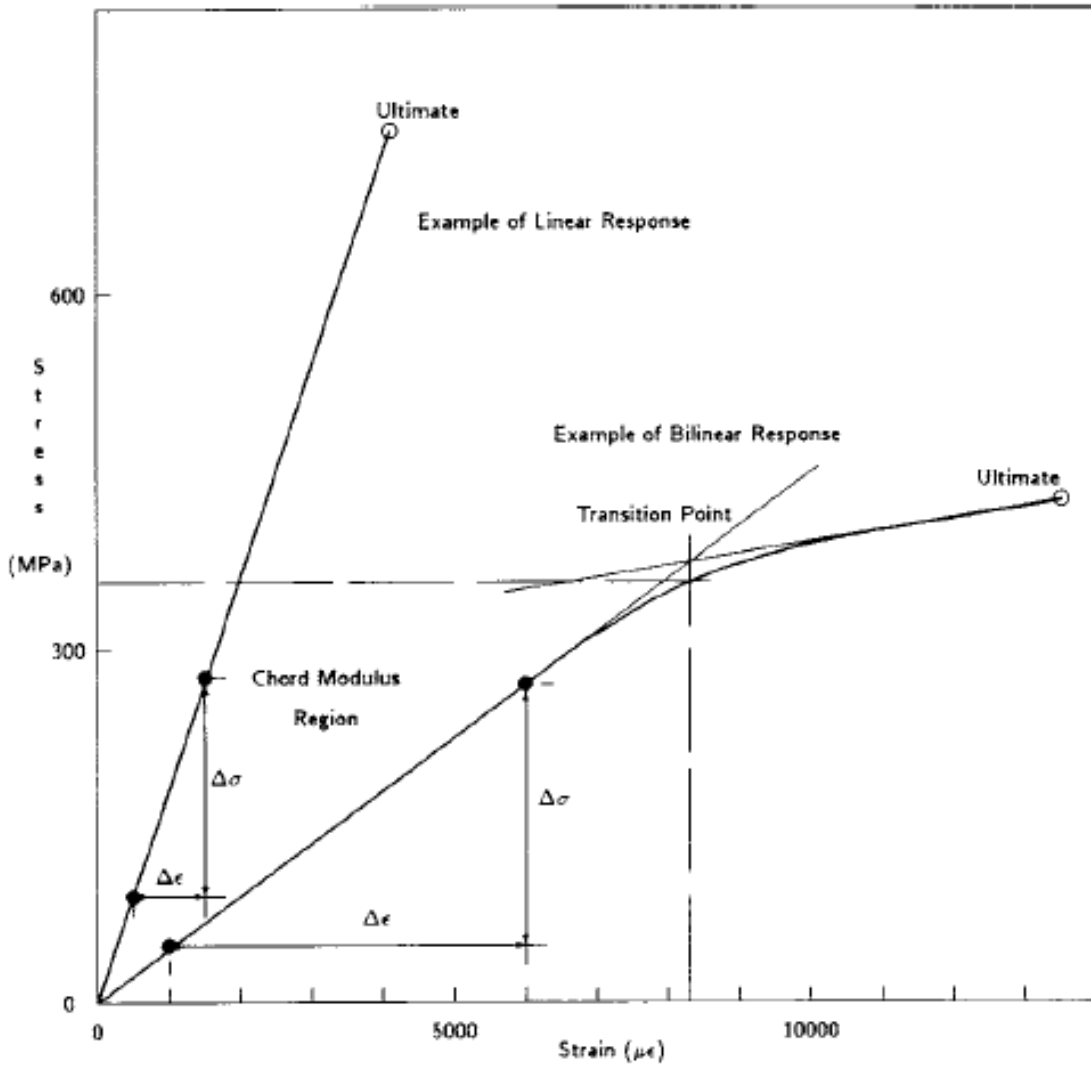


Figure 1-7-Tensile Stress-Stain Example (ASTM 3039, 2008)

<sup>1</sup> This set-up will be further explained in 2.3.2.

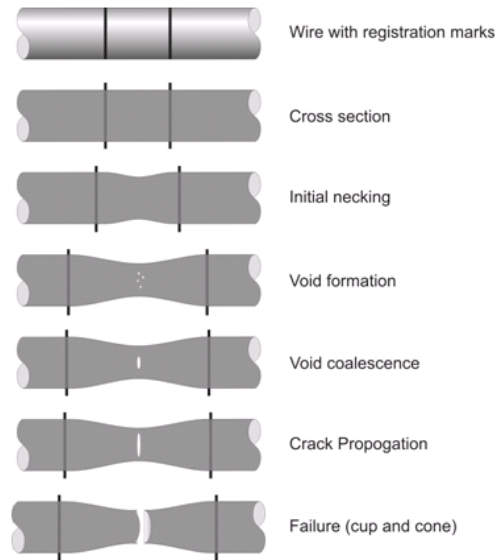


Figure 1-8-Material Necking and Failure (Mechanical Properties, n.d.)

### *Thermal Expansion Test*

The mean coefficient of linear thermal expansion,  $\alpha$ , is typically denoted in units of  $\mu\text{m}/\text{m}^{\circ}\text{C}$  or  $\mu\text{m}/\text{m}^{\circ}\text{K}$ . In Figure 1-9, the example material expands along an exponential curve with an increase in temperature. The slope of the secant line between two points on the curve is represented by  $\Delta L/\Delta T$  [ $\mu\text{m}/^{\circ}\text{C}$  or  $\mu\text{m}/\text{K}$ ];  $\alpha$  then represents, “the change in length, relative to the specimen length at ambient temperature.” (ASTM E831, 2012). Thus by multiplying the secant slope by  $L^{-1}$  [m], we are given the coefficient of linear thermal expansion. The expansion properties of a material are important for engineers to understand because they can cause unwanted stress and/or strain in a structure or cause sliding mechanisms to jam from unequal expansion. An application where a UAV may fly from the desert to 60,000 ft altitude represents a large temperature difference and if the strain is too great for a material, mission failure could be a likely scenario. There is also an interest in the thermal stability of FDM parts as it could prove to be a quick method for producing structures for optical devices where precision is paramount.

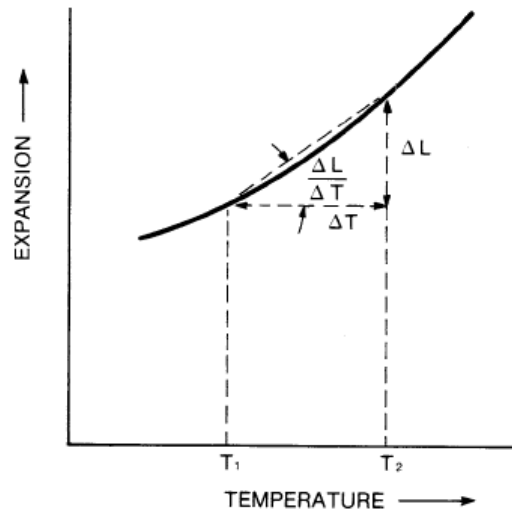


Figure 1-9-Thermal Expansion Graphical Relation (ASTM E831, 2012)

In chapter 2 the tensile testing and modeling is presented. In chapter 3 the thermal expansion testing and modeling is presented. In chapter 4 the findings from both experiments are summarized and conclusions are drawn from the data.

## 2 Tensile Testing and Modeling of FDM Manufactured Parts

This chapter presents the standard engineering practices for designing the tensile testing parts that will be produced through additive manufacturing. It summarizes the post-processing software used by Stratasys in planning and executing a Fused Deposition Modeling job. The chapter presents different tensile test specimen designs and provides reason for the choice of the final design. The tensile test set-up and experimental results are presented and analyzed. The chapter concludes with modeling in ANSYS Workbench to validate experimental results.

### 2.1 Design of Tensile Testing Articles

#### 2.1.1 Geometric Design Guidelines and Limitations for FDM Parts

The nature of the FDM process requires that certain considerations be made into the design of printed parts. Because the printer works by creating the product in discretized slices, it is prudent that engineers design parts whose printed z-dimension is a multiple of the desired slice height. This ensures that the software will correctly slice the product and provide equal layers while not trying to compensate for an imaginary “half-layer.” In our dogbone specimens, this became a very important design standard as they needed to be printed in different raster angles and in different orientations. As a result, all lateral dimensions were set as multiples of the slice height 0.1778mm. Notably there is a difference in dimensions between the 45/-45 raster angle and the 0/90/±45 raster angle plies. This is due to the fact that the repetition patterns are different between the two. In order to create a symmetric 45/-45 specimen, an extra layer must be added to create a pattern of n/-n/....n. In the 0/90/±45 specimen, the pattern of a block of eight layers is repeated and naturally is considered symmetric whether there are an odd or even number of repetitions. In ASTM 638, our group was fortunate that the tolerance was large enough to allow for an entire ply to create specimens as close to specifications as possible. The use of four decimal places was necessary to facilitate the slicing in Insight. The dimensions for the test articles can be found in Appendix A- Design and Manufacturing. The dimensions for the thermal test articles of the 0/90 and 45/-45 ply were the same because of the similar pattern repetition but different for the Quasi Isotropic specimen because of the greater amount of layers that are required. For the dogbone

geometries, all tensile tests except for modifying the thickness used the geometries from Figure 6-4 and Figure 6-7 for the 45/-45 ply and Quasi Isotropic ply respectively.

### 2.1.2 Stratasys Insight Software in Fortus FDM Machines

Insight is the programming tool used by Stratasys for printing on their Fortus FDM machines. It allows for the user to manipulate the toolpath of the printer head for each layer. Insight takes STL files and slices them into equal height layers as dictated by the user. From this point, users can either use the default function to draw in the tooling paths or custom paths can be created. Our group opted for the later method as variable manipulation plays a central role in our project. However, Insight does have limitations on automating the process for advanced composite design. In Figure 2-1 the “Other raster fill controls” option allows for raster angle manipulation, key to being able to creating composite structures with rotated angles between layers. However when it comes to creating the quasi-isotropic layering, there is not a constant delta between angles meaning that several groups have to be created and manually set for each group of two layers. This is a highly time-consuming process and improvements should be made to the program to allow for more complicated layering geometries. Under the “Main Parameters” section in Figure 2-1 the only modification made was through the Raster Width option. A raster width (also known as Bead Width) of 0.3048mm, 0.6790mm, and 0.4790mm was used for the small bead width, large bead width, and all other specimens respectively.

Another serious consideration in using 3D printing for end-use parts is how the geometries will be affected by the layering technique. Figure 2-2 shows a severe result of poor planning and understanding of printing geometries; by orienting the part in this manner, the radiused corner becomes terraced and it is expected that it will be significantly weaker than an arrangement with smooth radiused corners. There is currently not a smoothing function in Insight to deal with this type of issue so engineers must try to minimize geometries that will not resolve into smooth contours during printing. We are exploring this case in order to better understand the stresses experienced by a material printed in this orientation.

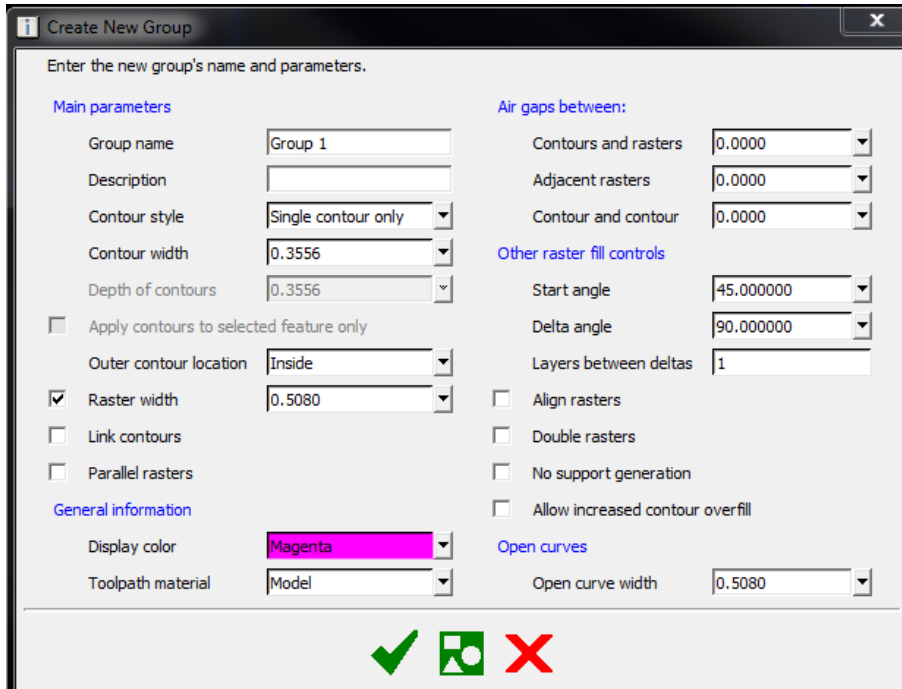


Figure 2-1 Custom Groups in Insight

Beneficial from a visual standpoint is the different display colors that are employed for each new custom group. This can give the designer a clear indication if their pattern has been repeated correctly throughout the part as also shown by Figure 2-2.

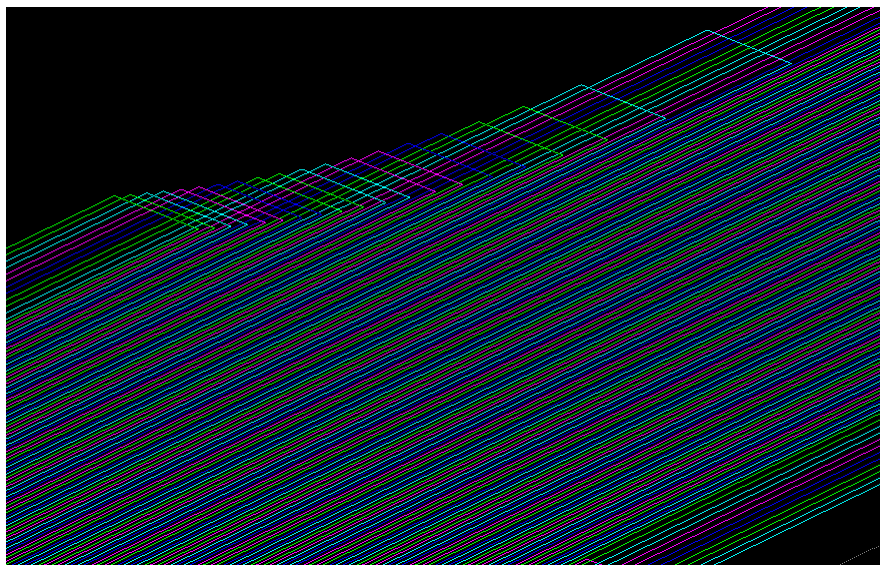


Figure 2-2 Layering Technique in Insight



### 2.1.3 Stratasys Control Center

Control Center is used as the printing sheet planner for parts already programmed through Insight. It predicts the time needed for printing as well as the material needed from support and modeling. An entire sheet was used for the printing job to print extra parts should some fail prematurely. The arrangement of this sheet can be seen in Appendix 6.1.

### 2.1.4 Deciding on the FDM parameters to test

The limited literature available on FDM processed ABS did have its benefit in reducing the number of potential test specimens. Early estimates placed the test articles per temperature test at an overwhelming 96 different configurations. From Ahn et al. (2002) it was determined that an increase in density is best for better performance in tensile and compressive testing. This directly relates to the problem of air gaps in FDM and by reducing the air gap density by increasing the density of the fill, there are lesser opportunities for stress concentrators to cause catastrophic failure at lower stresses. Fatimatuzahraa et al. (2011) also showed the dependency of raster angle for tensile tests, but our group determined that the difference was not great enough to neglect it from our tests when other variables will be tested in parallel. With this initial reduction, we retained 32 test articles as only the dense option would be used in FDM. In addition, due to utilization constraints this number of test articles was high. This figure was also high given the fact that many groups share the same polymer laboratory. The project's original intention of testing the relationships between and optimizing parameters was too resource intensive leading to a study that compartmentalizes this variable testing.

**Raster Angle:** Table 2-1 presents the catalog and compares the strength of raster angles in tension tests. Both the 45/-45 and 0/90 raster were not tested in the X-Axis (see Figure 2-2) as it was expected that they would maintain good strength in the X and Y directions. The Z-direction was of great interest to all test articles because delamination, the main destruction mechanism, elucidates the bonding strength between layers for different raster angles. Whereas X-Y-Z is the Width, Length, and Height respectively, the Y-stress will be testing the strength of the large cross-section orientation. In order to test in the Z-Axis (relative to the original large cross-section specimen) another dogbone will need to be created to reflect the

delamination mechanism for fracture. This will be accomplished by rebuilding the specimen where the layers originally built in the Z-direction are now built in the Y-direction, hence the small cross-section as explained earlier. Finally, in order to find what the stress would be in the X-direction as seen in Figure 2-2, the build orientation needs to once again be altered. By taking the face observed in the Y-Z plane and building it along the X-axis, the dogbone is now set to observe the stress along the X-Axis. The Quasi-Isotropic specimen will be tested in all three-axis as there is great interest within our group and the Lincoln Labs group as to this raster angle providing near-bulk material properties without directional dependency.

Test Article	0/90	45/-45	0/±45/90	X-Axis	Y-Axis	Z-Axis
1						
2						
3						
4						
5						
6						
7						

Table 2-1- Testing Tension by Raster

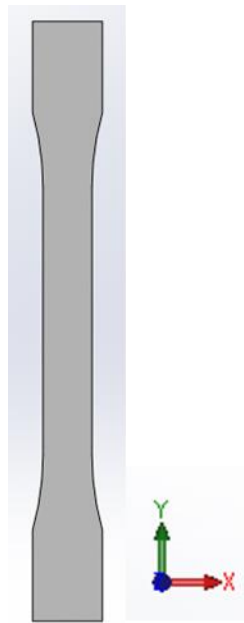


Figure 2-3- Tensile Test Orientation

**Bead Width:** Bead width is another important parameter to test between the standard 45/-45 and Quasi Isotropic raster angles. The two extremes for bead width were tested with the T12 tip available on the Fortus 400 to determine if there is a marked difference in tensile properties within the test specimens. It was expected that the smaller bead width would reduce air gaps within the test articles thus providing a tensile strength similar to the bulk material properties. It was also expected that the two raster angles chosen for this set will stack better than the 0/90; with better stacking between layers, there will be a further reduction in air gaps thereby strengthening the test articles. Table 2-2 summarizes the experiments that will be conducted in this experiment set.

Test Article	45/-45	0/±45/90	Bead 0.6790mm	Bead 0.3048mm
1				
2				
3				
4				

Table 2-2- Testing Tension by Bead Width

**Number of Plies:** The number of plies for the test articles is of wide importance to the MITLL group. With their UAV project, the interest in printing single ply structures to save weight is in an experimental phase. The experiments in Table 2-3 are designed to show the difference in strength from a single ply structure to a multiple ply one. The goal of this test is to find an empirical relationship between number of plies and strength with other parameters held constant. By providing the group with data as to the expected structural integrity of this factor, they will be able to make informed decisions in a trade-off analysis.

Test Article	45/-45	0/±45/90	Plies n = 1	Plies n = 6
1				
2				
3				
4				

Table 2-3- Testing Tension on Number of Plies

## 2.2 Stress-Analysis in SolidWorks

As stated in our review in Sec. 1.2, Ahn et al. (2002) had cautioned against radiused corners with FDM manufactured test articles. It was decided to conduct an analysis in SolidWorks for two different dogbone specimen designs. SolidWorks uses the bulk material properties and a solid mesh to conduct a simulation of stresses and forces against an object with defined fixed faces or nodes. Though the analysis is not perfect for the orthotropic nature of FDM parts we seek only to understand how stress is distributed through different dogbone geometries. This model strictly only satisfies the conditions for a Linear Elastic Isotropic body. In Figure 2-4 the standard dogbone (as designed using geometry from ASTM 638) and the modified dogbone are presented. The modified dogbone consists of filleted corners using a constant radius to maintain a continuous geometry. The idea behind this design was to eliminate corners so as to avoid a stress-concentrator scenario and a possible premature fracture scenario. Its dimensions can be found in Figure 6-10.

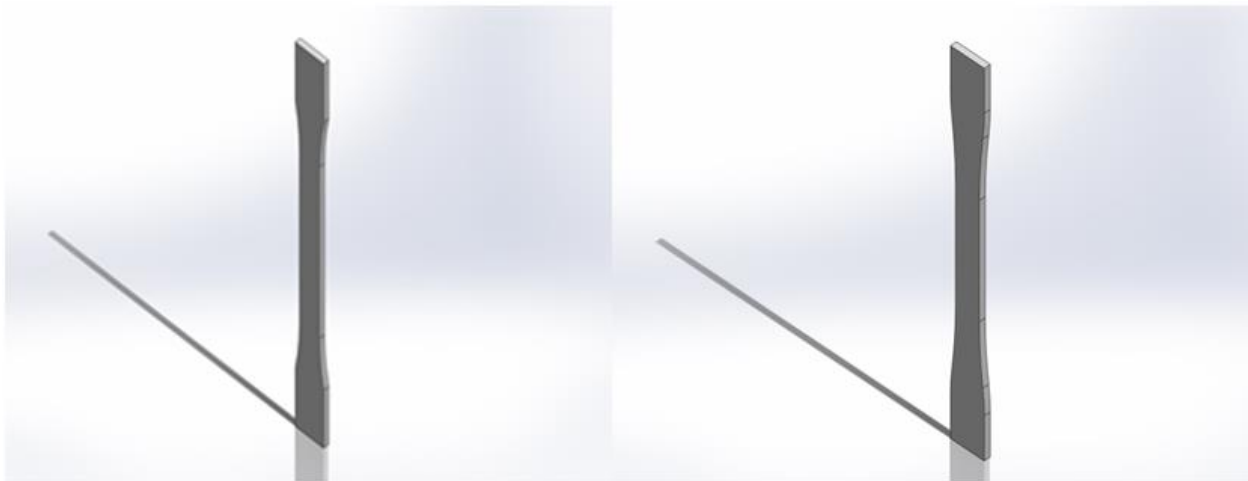


Figure 2-4 Regular Dogbone (Left) Modified Dogbone (Right)

Both specimens were loaded and fixed in the exact same manner as shown in Figure 2-5. The purpose was to show the effects of an axial stress on the test specimens so the specimens were fixed on the bottom face (shown in green arrows) and a  $100 \text{ N/m}^2$  stress was applied to the top face (shown in red). The material was set as ABS to reflect nearest to the material properties of ABS-M30 from Stratasys. It was expected that the standard dogbone would develop a large gradient in stress at the corners. This was the reasoning behind trying to eliminate corners in

the modified dogbone as we wanted to determine the effect of corners in axial stress conditions.

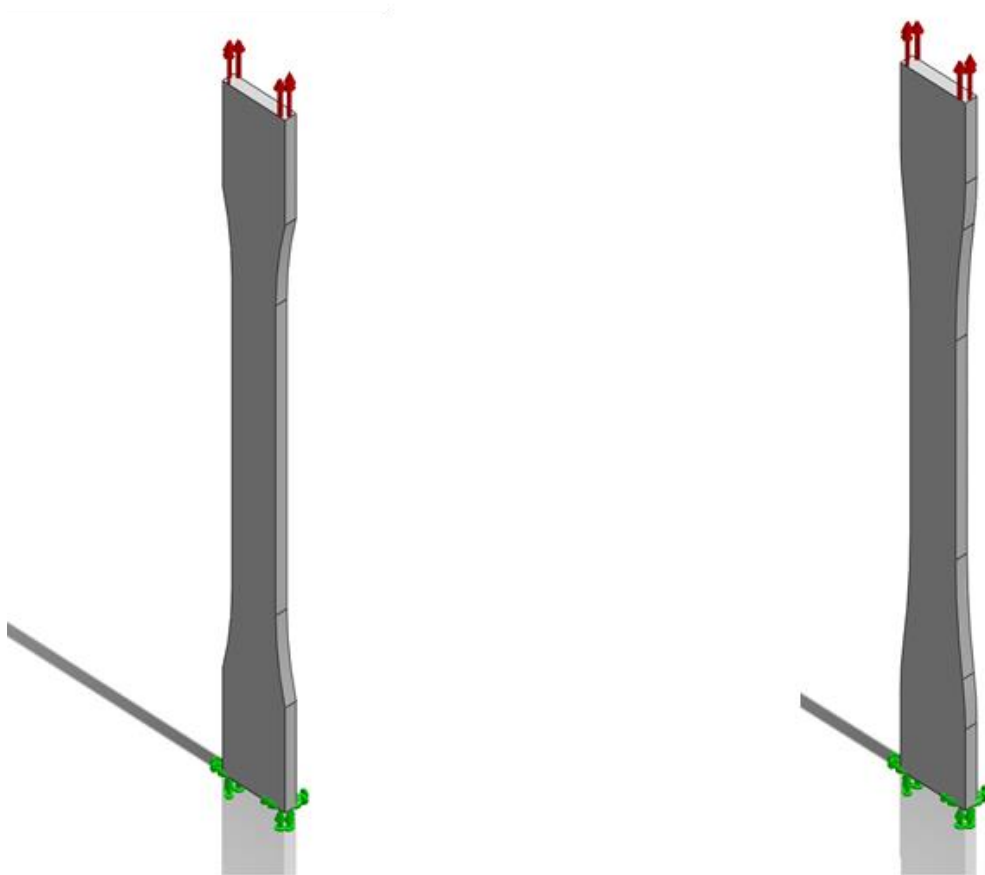


Figure 2-5 Constraint and Stress Set-up Regular Dogbone (Left) Modified Dogbone (Right)

The results show in Figure 2-6 that the corner provides some strengthening mechanism to the dogbone. The regular dogbone provides more strength before the radiused corner in this tensile test meaning that we would not expect to see a premature fracture at the corner. Overall, however, the modified dogbone experiences less von Mises stress throughout as evident from the scale. Our study is not concerned with making an overall stronger part but to obtain accurate material properties without the chance of premature failure. Literature has cautioned against radiused corners for FDM parts so this further solidifies the design of the tensile dogbone. Thus, it was decided to use the regular dogbone design (as described in ASTM 638) for the tensile testing.

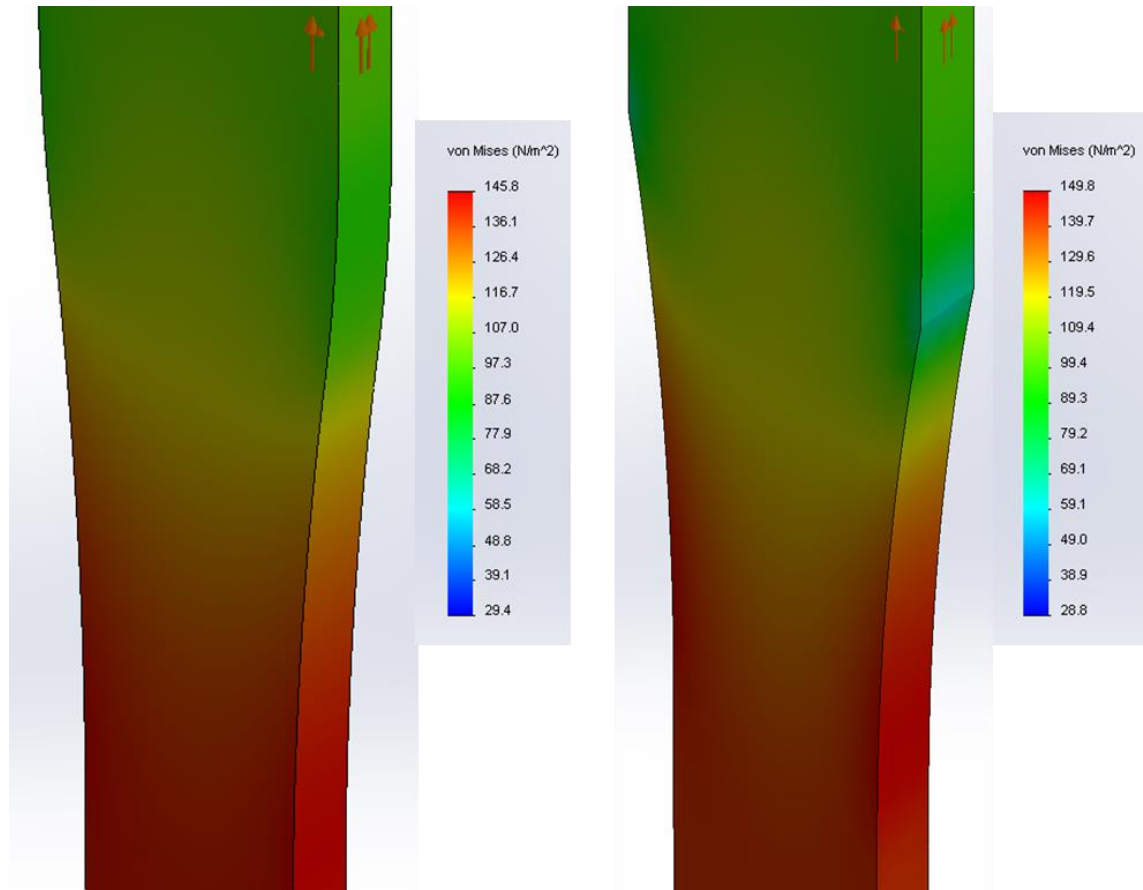


Figure 2-6 Results of SolidWorks Express for Modified Dogbone (Left) and Regular Dogbone (Right)

## 2.3 Tensile Experimental Set Up

This section presents the method of selection and use of strain gauges for the experiments carried out.

### *Strain Gauge Selection for Tensile Testing*

Strain gauges were used to measure strain along different axis of the samples. The tensile test uses strain to compare the FDM printed parts to composite theory. Strain gauges are commonly found in applications on metals, typically with steel and aluminum, which have very different material properties than plastics. Choosing the correct strain gauges for measurements on plastics requires careful consideration with respect to multiple different factors including the expected applied strain, operating temperature, and duration of the test (Micro-Measurements, 2010).

A rectangular rosette gauge was used in the tensile shown in Figure 2-7. The geometry of the gauge is important based on the types of data we want to acquire. In the tensile test samples, a rectangular rosette is used to record measurements parallel and perpendicular to the direction of the force as well as at a 45° angle. Using this type of gauge allows us to collect enough data to relate the FDM printed parts to mechanical properties of composites and analytical theory.

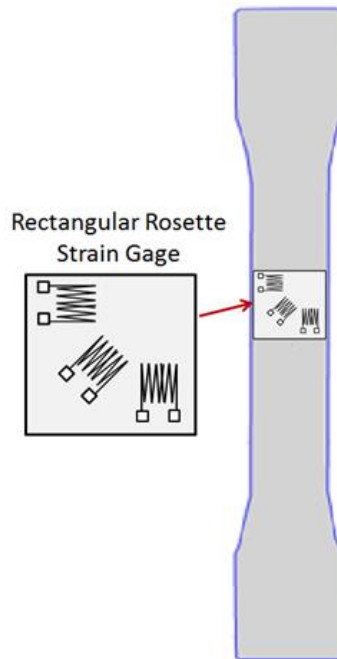


Figure 2-7 Diagram showing location of strain gauges on tensile and thermal samples

The thermal conductivity of the plastic is important to consider. The most common application of strain gauges is for testing metals and alloys. Metals have a thermal conductivity much greater than that of plastics. Strain gauges are typically calibrated to match the thermal conductivity of the material they are testing. Because the thermal conductivity of plastics is so much lower, a gauge calibrated for a material with a low thermal conductivity should be used and the excitation voltage on the gauge should be kept very low in order to avoid overheating the gauge and causing poor data collection. This also is affected by the resistance of the gauge. A gauge with a higher resistance should be used on plastics to help avoid overheating of the gauge. In our case, we choose a 350  $\Omega$  strain gauge with a maximum excitation voltage of 9 V, originally rated for steel.

The SGD-3/350-RYT81 rectangular rosette gauge was chosen for tensile testing from OMEGA Engineering. Each gauge is  $350\Omega$  and rated for steel ( $V_{EX} \leq 9V$ ) with a maximum applied strain of  $30,000 \mu\epsilon$ . Lead wires are soldered to the gauges and installed on the sample using SG401 ethyl-based cyanoacrylate adhesive supplied by OMEGA.

### 2.3.1 The Wheatstone Bridge

The Wheatstone bridge is the standard set-up for utilizing strain gauges. In simple terms it is a voltage divider that encompasses variable resistors (the strain gauge). In Figure 2-8 a Quarter-bridge set-up is presented where one resistor has been replaced by a strain gauge. When designing the Wheatstone bridge it is essential that the resistance on either side of the voltage divide is proportionally the same as it provides the necessary calibration. In a balanced state, the voltmeter,  $V_O$ , will read a zero voltage, but as the resistance is changed on the right side by an induced strain, a non-zero reading will occur.

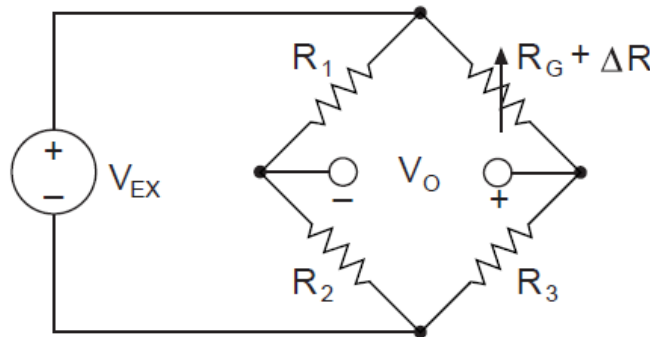


Figure 2-8 One-Quarter Wheatstone Bridge (Strain Gauge Measurement, n.d.)

By using Equation 2-1, the induced strain on the specimen can be obtained through:

$$\frac{V_O}{V_{EX}} = \frac{-G * \epsilon}{4 \left( 1 + G * \left( \frac{\epsilon}{2} \right) \right)}$$

Equation 2-1 Calculating the Strain from Voltage Change (Strain Gauge Measurement, n.d.)

where,  $V_O$  is the measured voltage,  $V_{EX}$  is the excitation voltage,  $G$  is the Gauge Factor, and  $\epsilon$  is the strain. It is also recommended that in using a quarter-bridge design that special considerations be made to account for the resistance of lead wires in the bridge and the impact



of temperature on resistance. The three-wire variation presented in Figure 2-9 is a large advantage to lead wire resistance in the measurement arm  $R_{L2}$  as it approaches near zero, thus giving a more accurate reading between the two arms.

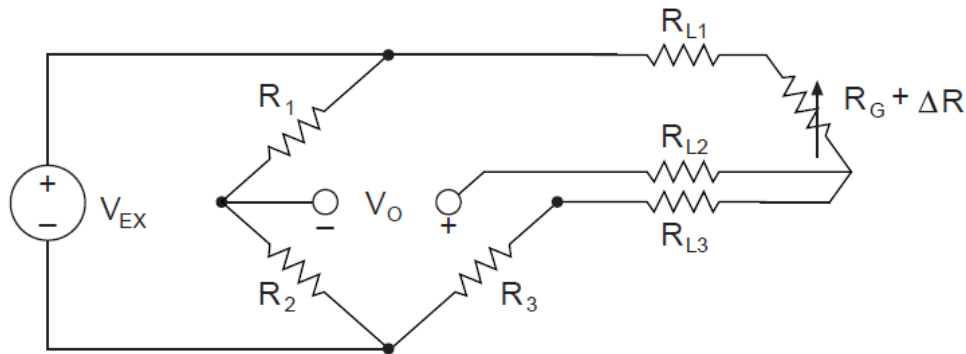


Figure 2-9 Three Wire Variation for Quarter-bridge (Strain Gauge Measurement, n.d.)

In Wheatstone bridges, the  $R_1$  &  $R_2$  resistors are typically set as high precision resistors to, “provide a stable reference voltage of  $V_{EX}/2$  to the negative input lead of the measurement channel” (Strain Gauge Measurement, n.d.). The left arm remains stable thus allowing for a more precise reading from the voltage difference. For our purposes, we will be using quarter-bridge designs for the Tensile Test portion and half-bridge designs for the Heat Expansion Test. Reasoning behind each bridge design will be presented in the subsequent sections.

### 2.3.2 Tensile Test

Data from this experiment will provide the ultimate tensile strength, ultimate tensile strain, tensile chord modulus of elasticity, Poisson’s ratio, and the transition strain. A rosette strain gauge was applied to the test articles as shown in Figure 2-7 where each gauge within the rosette was wired with a NI 9945 Quarter Bridge Completion circuit (Figure 2-10) to complete the arm as  $R_3$  (see Figure 2-9) as a one-quarter bridge. This was then connected with the NI 9237 signal conditioner to complete the Wheatstone Bridge and data was fed into a laptop. Port 0 on the NI9945 provides the excitation voltage to the strain gauge from the NI 9237. The requirement for a three-wire quarter bridge as demonstrated in Figure 2-9 was met with a jumper across ports 2 and 1 as it was deemed that the strain gauge pad would be too fragile to adequately support two 30 Gauge lead wires. Referencing Figure 2-9, it is hoped that the

difference in resistance between  $R_{L2}$  and  $R_{L3}$  will be much smaller than the change in resistance that the strain gauge experiences during the tensile test. Upon conference with Mr. Longton, who is experienced in applying strain gauges in Quarter-Bridge set-ups, it was validated that if the strain gauge set-up with this alteration would calibrate, then it would be adequate to provide accurate strain data during testing.



Figure 2-10 NI 9945 Quarter Bridge Completion (NI 9944, NI 9945, n.d.)

Testing this set-up on a bench, it was discovered that the strain gauges were very sensitive to any bend in the dogbone test articles. In order to mitigate this problem during testing, the articles were loaded into the Instron with a minimal tensile force to ensure that there was no residual bend in the part and then calibrated. The program used for calibration and data acquisition was Signal Express from National Instruments. It allows for a quick set-up of our strain gauges, calibration, data recording and exporting, and troubleshooting without the need for extensive block programming that would be necessary in a regular LabView VI.

The set-up for the Tensile Test is shown in Figure 2-11. In order to reduce the time for breakdown and set-up between tests, the Quarter Bridge Completion Circuits were affixed to the structure of the Instron using electrical tape (denoted by 1). Wires could be removed from the NI 9945 and replaced while working at the Instron. This also helped in stabilizing the lead wires from the rosette strain gauge (shown in 2) from any jitter induced from airflow in the room. The Instron has two clamps that will provide a uniaxial tension load. (4) remains

stationary during the test while (3) was set to move with a displacement of 5 mm/min. In the preliminary test with a test dogbone it was discovered that the clamps of the Instron provided a large crushing force seriously damaging the tabs of the dogbone. In order to rectify this problem, shims (5) were used to alleviate the crushing force. By measuring the thickness of the test articles with a caliber, shims were matched up to be 0.003in less than the thickness; this provided a firm grip on each dogbone and eliminated the problem of significant crushing from the Instron.

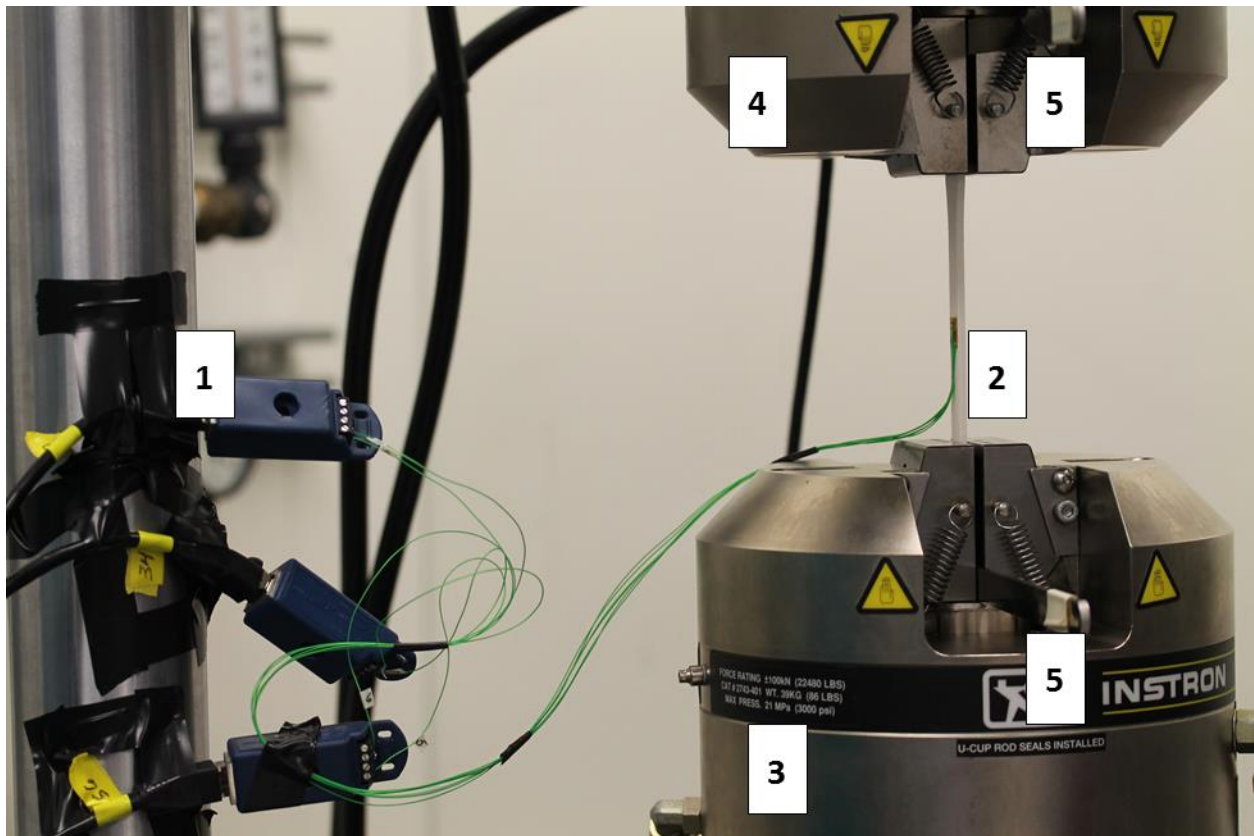


Figure 2-11 Tensile Test Set-up: 1. Quarter Bridge Completion, 2. Test Section, 3. Moving Instron Clamp, 4. Stationary Instron Clamp, 5. Shims

During the test, the Signal Express data recording for the full rosette strain gage was initiated at a recording of 10 samples at 1.62 kHz. The Instron was then initiated and recorded at 100Hz; in order to match the two data sets, sync correlation was used to linearly interpolate the Instron recording frequency to that of the strain gauge.

## 2.4 Tensile Testing Results

Due to time limitations of this project, only one sample of each build parameter set was tested. Some dogbones did not yield full sets of data due to premature failure of the gauges through leads breaking off the pads during testing or tears in the backing foil. Some samples also slipped in the grips when subjected to higher loads thus failing to destruct during the tensile test. These articles could not be used again as they had most likely experienced stress relaxation and would not exhibit the true material properties for the specimen.

### 2.4.1 Mohr's Circle for Principle Strains and Principle Stresses.

Mohr's circle is an extremely powerful tool to understand stress and strain tensors in a two-dimensional state. The rosettes used for the tensile tests, when aligned properly provide the necessary data to draw Mohr's Circle and calculate principle stresses or strains and their direction at any point during testing. Vishay Inc. provides a useful explanation of the strain transformation equations and how to create Mohr's Circle with gauges aligned at  $0^\circ$ ,  $45^\circ$ , and  $90^\circ$  to the induced stress. Figure 2-12 shows the arrangement of a planar rosette with its coordinate system placing gauge 3 along the y-axis and gauge 1 along the x-axis.

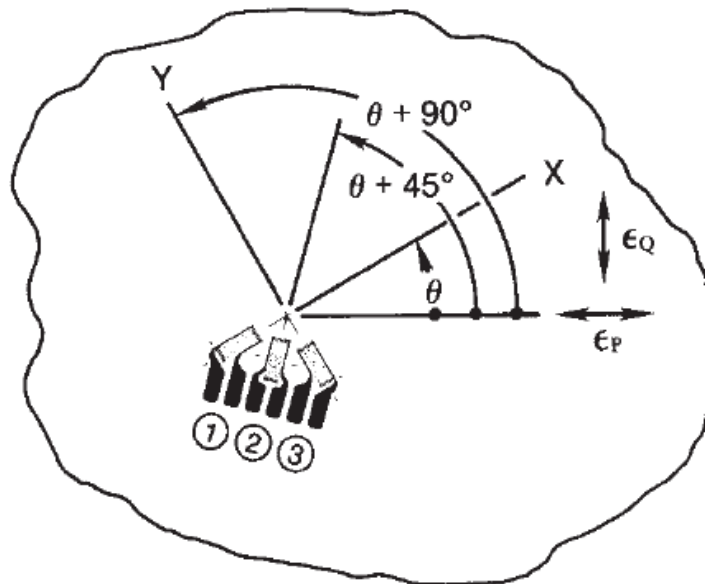


Figure 2-12 Rosette Element Coordinate (Strain Gage Rosettes, 2010)

When using a gauge such as this, Vishay Inc. has provided the mathematics to find the principle strains and the rotated angle as shown in Equation 2-2 and Equation 2-3.

$$\varepsilon_{P,Q} = \frac{(\varepsilon_1 + \varepsilon_3)}{2} \pm \frac{1}{\sqrt{2}} * \sqrt{(\varepsilon_1 - \varepsilon_2)^2 + (\varepsilon_2 - \varepsilon_3)^2}$$

Equation 2-2- Principle Strains (Strain Gage Rosettes, 2010)

$$\theta = \left(\frac{1}{2}\right) \tan^{-1} \left( \frac{\varepsilon_1 - 2\varepsilon_2 + \varepsilon_3}{\varepsilon_1 - \varepsilon_3} \right)$$

Equation 2-3- Rotated Angle (Mohr's Circle) (Strain Gage Rosettes, 2010)

where,  $\varepsilon_n$  is the strain along the n strain gauge. Figure 2-13 shows how we can apply these calculations into creating Mohr's Circle for strain. We are allowed to apply it in this fashion as the spacing between the "fibers" is much smaller than the length of the rosette strain gauge. Therefore, we are obtaining a bulk strain as compared to a microscopic strain between two "fibers." In Mohr's circle every theta rotation from the principle axis is double that of the orientation. Mohr's circle will also provide the principle shear strains helping to better characterize the articles in testing.

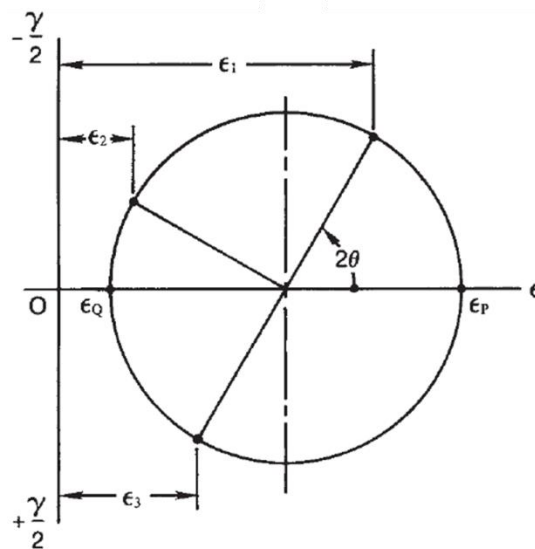


Figure 2-13 Mohr's Circle for Strain (Strain Gage Rosettes, 2010)

## 2.4.2 Y-axis Oriented Dogbones

One of the complete set of results originally intended to be compared is the Dogbones printed in the Y-axis orientation. This design follows standard convention with additive manufacturing by reducing the amount of support material and time required to build. Therefore, this provides a good baseline for material strengths of the different raster angles, something both our project team and the Lincoln Laboratory group was interested in. In this section each raster orientation is presented separately, followed by a comparison.

### *45/-45 Raster Printed in the Y-Axis Orientation*

The 45/-45 Dogbone printed in the Y-axis Orientation represents a standard print without any process parameter changes in Insight. Figure 2-14 shows stress-strain data in the direction of the tensile loading as recorded by the Instron DAQ and the strain gauge positioned in-line with the load. From this graph we can determine that the Young's Modulus for this article is 2400MPa which is the reported value from the Stratasys Spec-Sheet for ABS-M30. The Ultimate Tensile Strength (UTS) is 23.86 MPa, a 34% decrease from the reported Stratasys value. Graphically, it appears that the Transition Point lies somewhere near 0.008 Strain but this plastic region is short lived denoting that the test article exhibited a moderately ductile fracture. This observation is corroborated from Figure 2-15 as it is evident that fracture pattern is jagged with little necking evident. This implies that the additive manufactured parts no longer behave like polymers where high ductility results in a large plastic region of deformation.

The fracture of this test article was fortunately in the middle of the fracture zone underneath the rosette strain gauge implying that the principal strains and stresses were measured by the strain gauge. Presented in Figure 2-16, Mohr's Circle gives the strain state just before fracture with a Maximum Shear Strain of 0.006 and Maximum Normal Strain at 0.011. This data will be used to check the ANSYS models for validation of the Principal Strains.

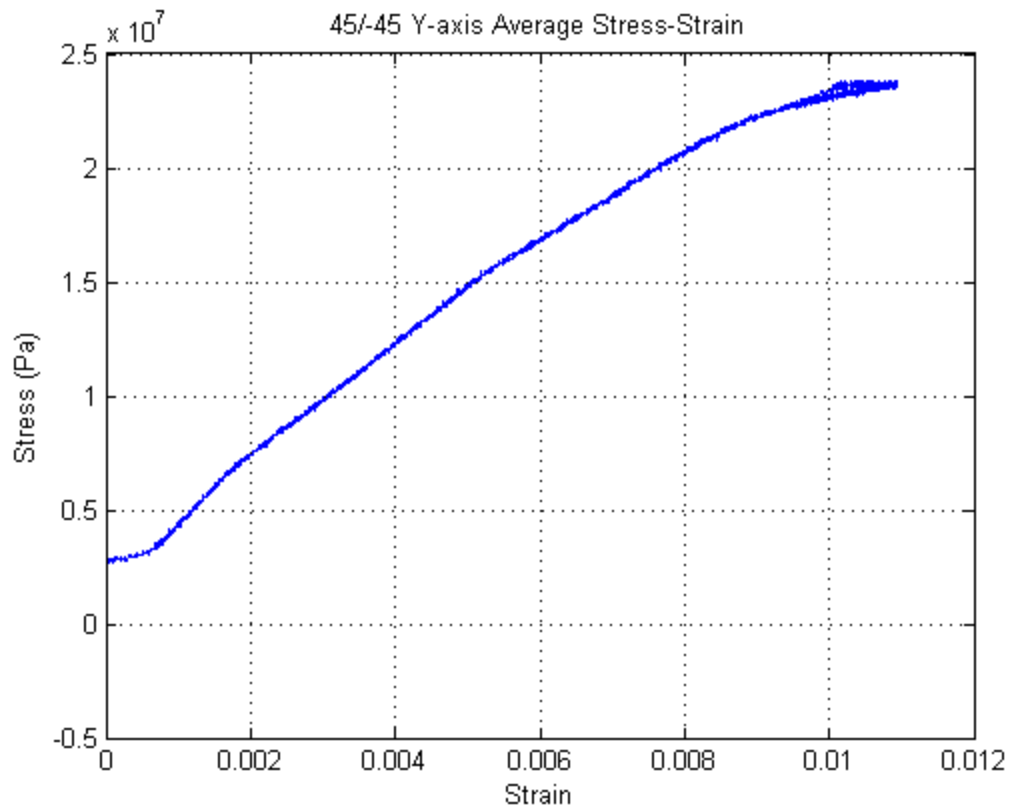


Figure 2-14 Stress-Strain Curve for 45/-45 Raster Y-Axis Orientation

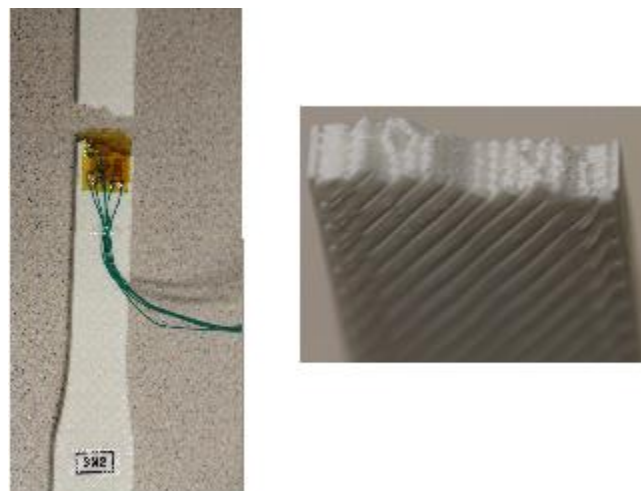


Figure 2-15 45/-45 Y-Axis Oriented Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)

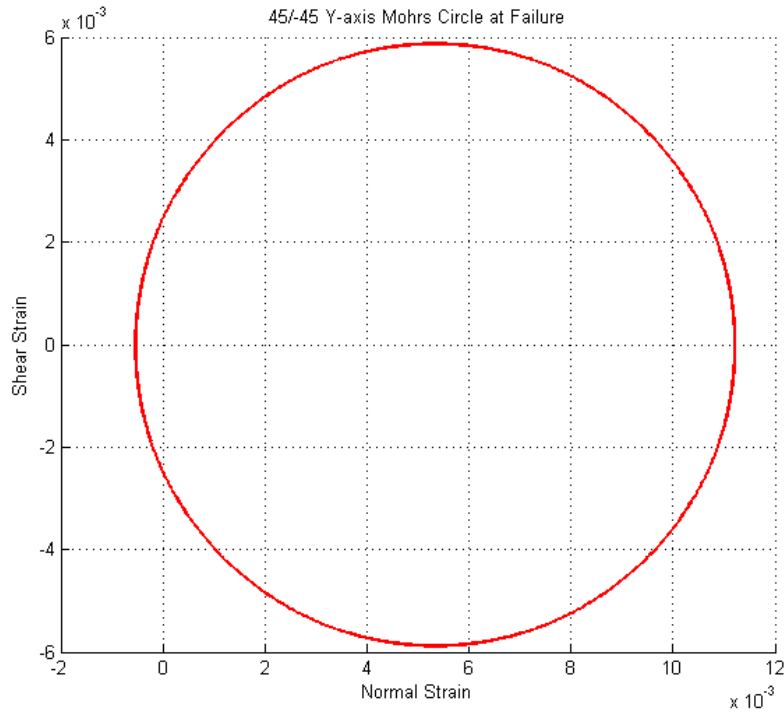


Figure 2-16 45/-45 Raster Y-Axis Orientation Mohr's Circle at Material Failure

### *0/90 Raster Printed in the Y-Axis Orientation*

The 0/90 raster is another common orientation as half of the layers are in line with the tensile load meaning that the stress and strain are in-line with some of the beads. In Figure 2-17 the stress-strain data is presented in the loading direction. Calculating the Young's Modulus from the graphical data gives a value of 2300MPa, which is a 4% decrease from the Stratasys reported value. The Ultimate Tensile Strength for this raster orientation is 20.83MPa which is a 42% decrease from the Stratasys reported value. This data appears to indicate that there is limited to no plastic deformation indicating a brittle fracture. In Figure 2-18, the fracture appears to be an almost clean cut which is logical as the failure of the raster oriented at  $0^\circ$  to the load will sever in-bead and the raster oriented at  $90^\circ$  will sever between the beads leaving no preference to an angular fracture in a purely tensile load.



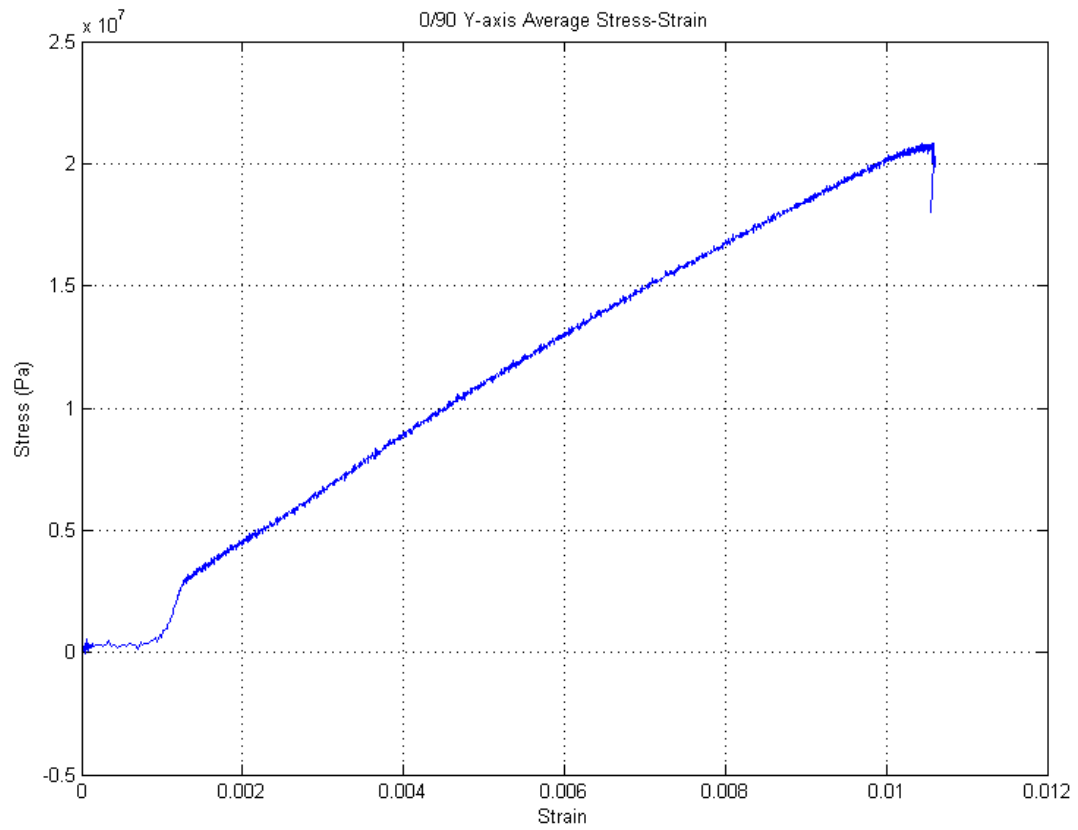


Figure 2-17 Stress-Strain Curve for 0/90 Raster Y-Axis Orientation



Figure 2-18 0/90 Y-axis oriented Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)

Figure 2-19 presents Mohr's Circle before failure of the test specimen. Caution must be taken when using this data as it represents the strain state where the rosette gauge was located and not where the fracture actually occurred (at the termination of the radiused section). From the initial ANSYS model (Figure 6-17), it appeared that the variation in strain from these two sections is minimal; therefore, we can use this data to determine validation in the ANSYS program for a Maximum Shear Strain of 0.007 and Maximum Normal Strain of 0.015 keeping in mind some error will exist.

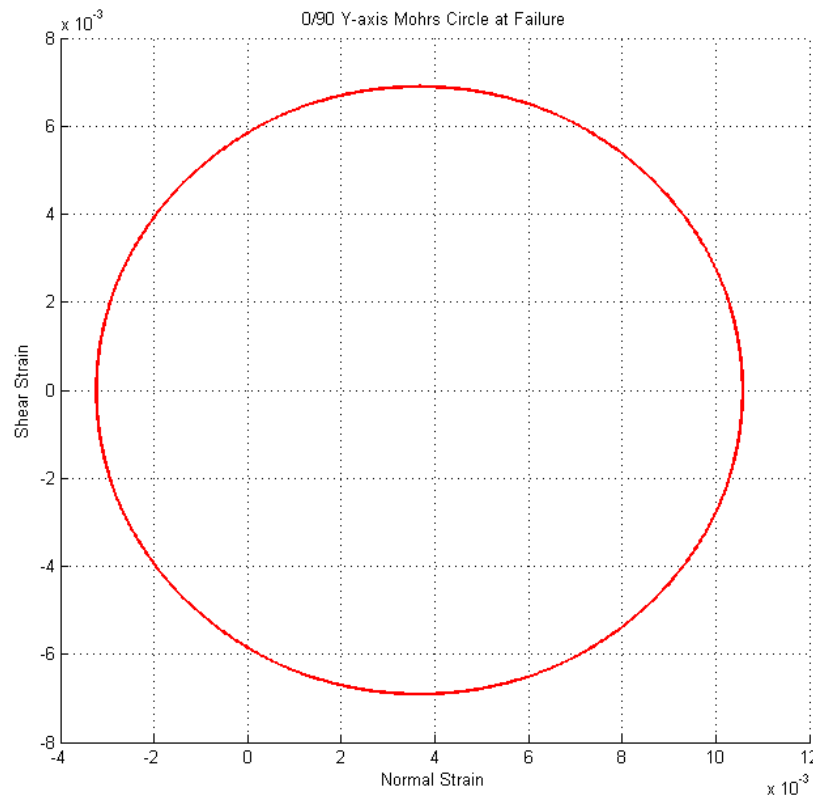


Figure 2-19 0/90 Raster Y-Axis Orientation Mohr's Circle at Material Failure

### *Quasi Isotropic Raster Printed in the Y-Axis Orientation*

The Quasi Isotropic raster orientation was an experiment in attempting to balance the raster angles in order to create a stronger part that behaved more like the bulk properties of ABS-M30. Figure 2-20 presents the stress-strain data for the Quasi Isotropic test article in the loading direction. The Young's Modulus calculated from the graph is 2000MPa which is a 16.7%

decrease from the Stratasys reported value and the Ultimate Tensile Strength is 20.89MPa, a 42% decrease. The graph appears to have a defined elastic and plastic region where the transition point appears to occur at a strain of 0.009. The plastic region is pronounced which means that this material exhibited a ductile fracture. From Figure 2-21, we observe that there is a rough fracture pattern typical to that of a moderately ductile fracture. Once again, there is no evidence of necking for the test specimen which is contrary to what polymers would typically exhibit after a state of plastic deformation.

Mohr's circle, presented in Figure 2-22, shows a strain state above the fracture zone. The data will be used with the caution in ANSYS for model validation. Reported from the graph, the Maximum Shear Strain is 0.01 and the maximum Normal Strain is approximately 0.0155.

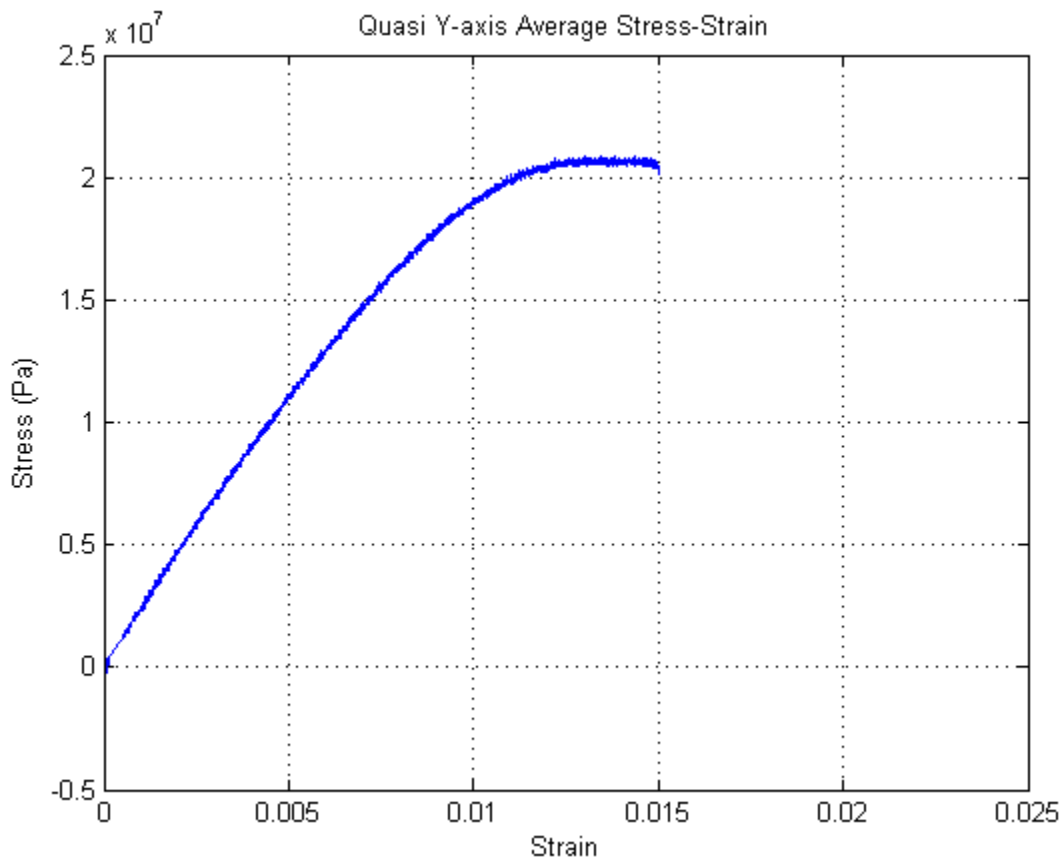


Figure 2-20 Stress-Strain Curve for Quasi Isotropic Raster Y-Axis Orientation

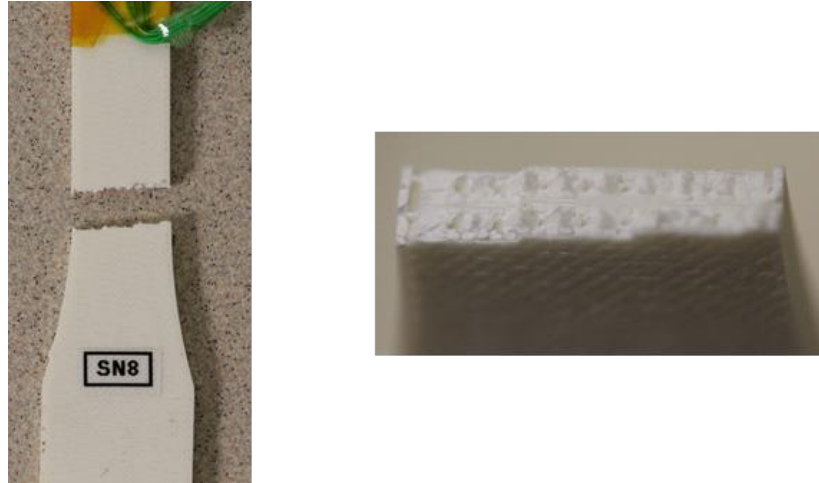


Figure 2-21 Quasi Isotropic Y-Axis Oriented Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)

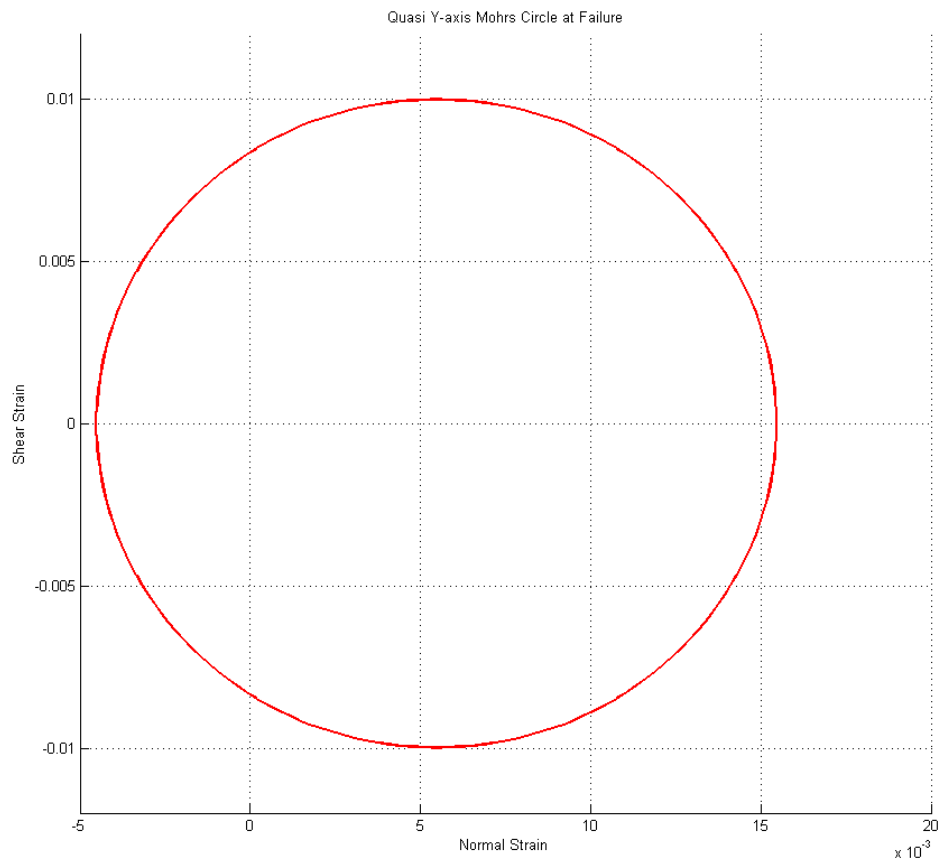


Figure 2-22 Quasi Isotropic Raster Y-Axis Orientation Mohr's Circle at Material Failure

### *Comparing the Y-Axis Oriented Dogbones*

The three different dogbones exhibited some similar and differing qualities from the tensile destructive testing. The 0/90 was considerably more brittle than either the 45/-45 or Quasi Isotropic which is most likely due to the nature of the internal load distribution. The 45/-45 and Quasi Isotropic have an advantage in being able to experience more balanced vector components of stress within the rasters which leads to the plastic region being more pronounced in the both of them. The Quasi Isotropic experiences the largest strain state before fracture owing to the higher ductility this printing orientation has over the other two, but it also has a lower UTS than the 45/-45 meaning the Quasi Isotropic is slightly weaker. All of the test articles did not exhibit any noticeable necking during testing and as a result all of them fundamentally are not behaving like polymers.

#### **2.4.3 Z-Axis Oriented Dogbones**

The Z-axis oriented Dogbones provide a unique opportunity to observe the failure between layers for FDM-printed parts. The bonding strength between layers will be less than that of the layers themselves. In this section we will present the results from the three different raster orientations and compare them with one another. This data also provides a Young's Modulus in the Z-direction for each of the orientations.

#### *45/-45 Raster Printed in the Z-Axis Orientation*

The 45/-45 Raster Orientation Dogbone printed in the Z-Direction represents a brittle printing arrangement for a product that will be subjected to tensile loading. In Figure 2-23, the stress-strain curve is highly linear with no obvious transition point. Thus, the dogbone is in the elastic region until its brittle fracture at the Ultimate Tensile Strength of 20.60MPa. This is a 13.7% reduction in the UTS observed in the Y-Axis Oriented Dogbone with a 45/-45 raster. The Young's Modulus in this orientation is 2252MPa, a 6.2% decrease from that of the Y-Axis Orientation. The brittle fracture is also evident in Figure 2-24 with a clean fracture that the toolpaths can be seen from when the articles were printed.

The fracture on this test article was lower than the strain gauge rosette so Principal Strain data must be taken cautiously. Figure 2-25 provides Mohr's Circle for the Principal Strains before fracture in the location of the rosette. The Maximum Shear Strain was approximately 0.0007 and the Maximum Normal Strain was approximately 0.0102. This strain data will not be used to validate an ANSYS model due to time restrictions, but we provide the data for future researchers to develop the modeling techniques.

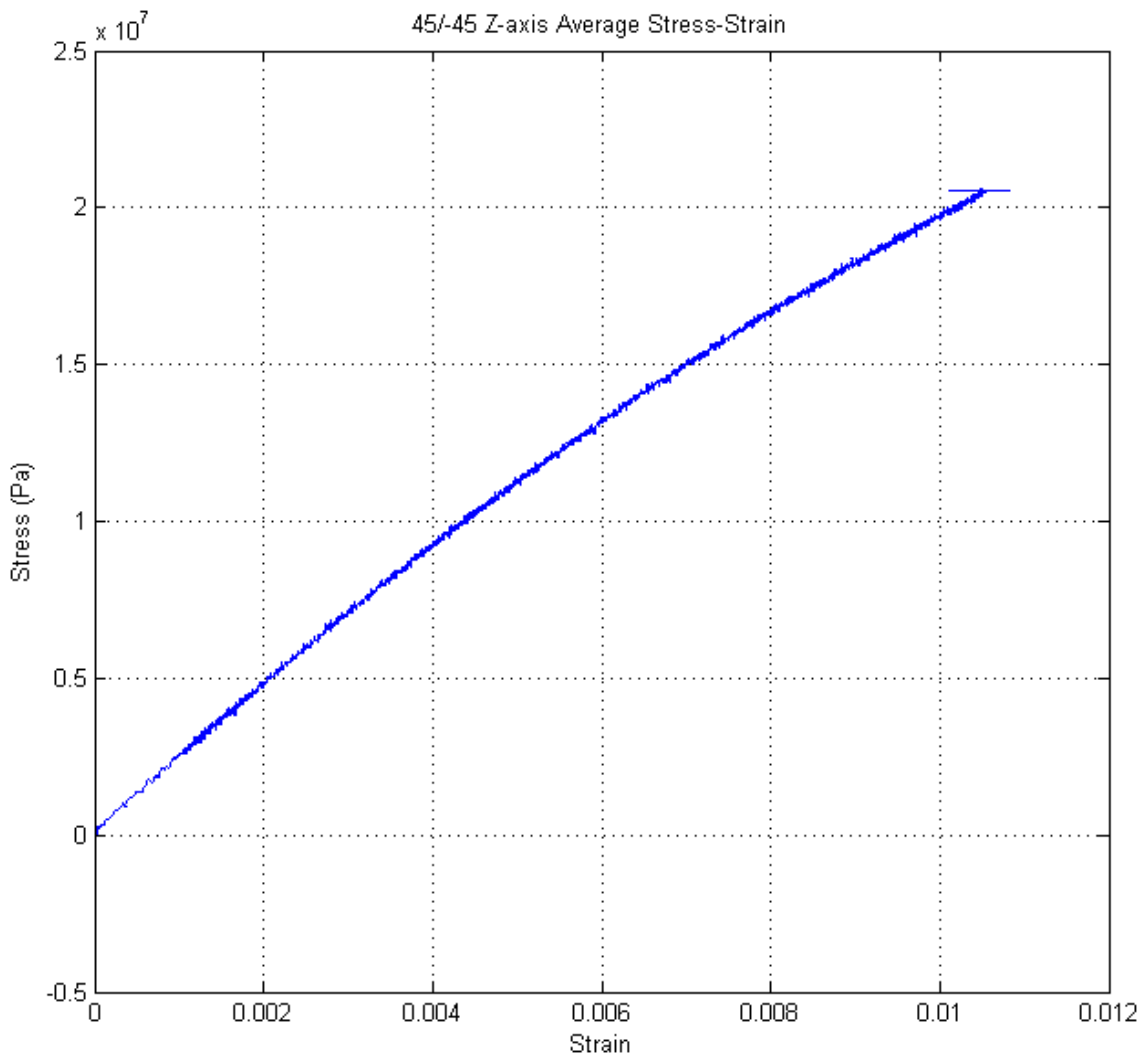


Figure 2-23 Stress-Strain Curve for 45/-45 Raster Z-Axis Orientation



Figure 2-24 45/-45 Z-axis Oriented Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)

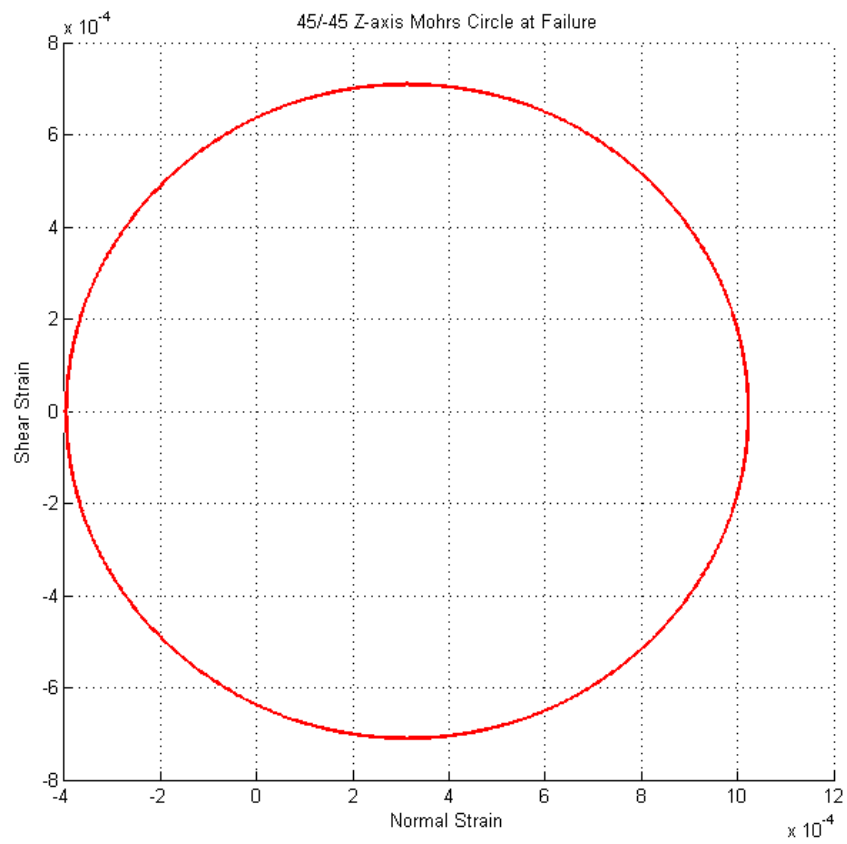


Figure 2-25 45/-45 Raster Z-axis Orientation Mohr's Circle at Material Failure

### *0/90 Raster Printed in the Z-Axis Orientation*

Complete data for this test article was not possible due to premature failure on Gauges 12 and 34. Thus, only data in the direction of the loading was available for this experiment still allowing for the creation of a stress-strain curve as seen in Figure 2-26. The Young's Modulus in this orientation is 2061MPa which is a 10.4% decrease from the Modulus in the Y-Axis Oriented dogbone, and the UTS for this test article is 19.69MPa which is a 5% decrease. The 0/90 test article also exhibits a borderline brittle fracture as the stress-strain graph shows a limited plastic region with a transition around 0.008 strain but the break is fairly clear as seen in Figure 2-27 with the visible toolpath.

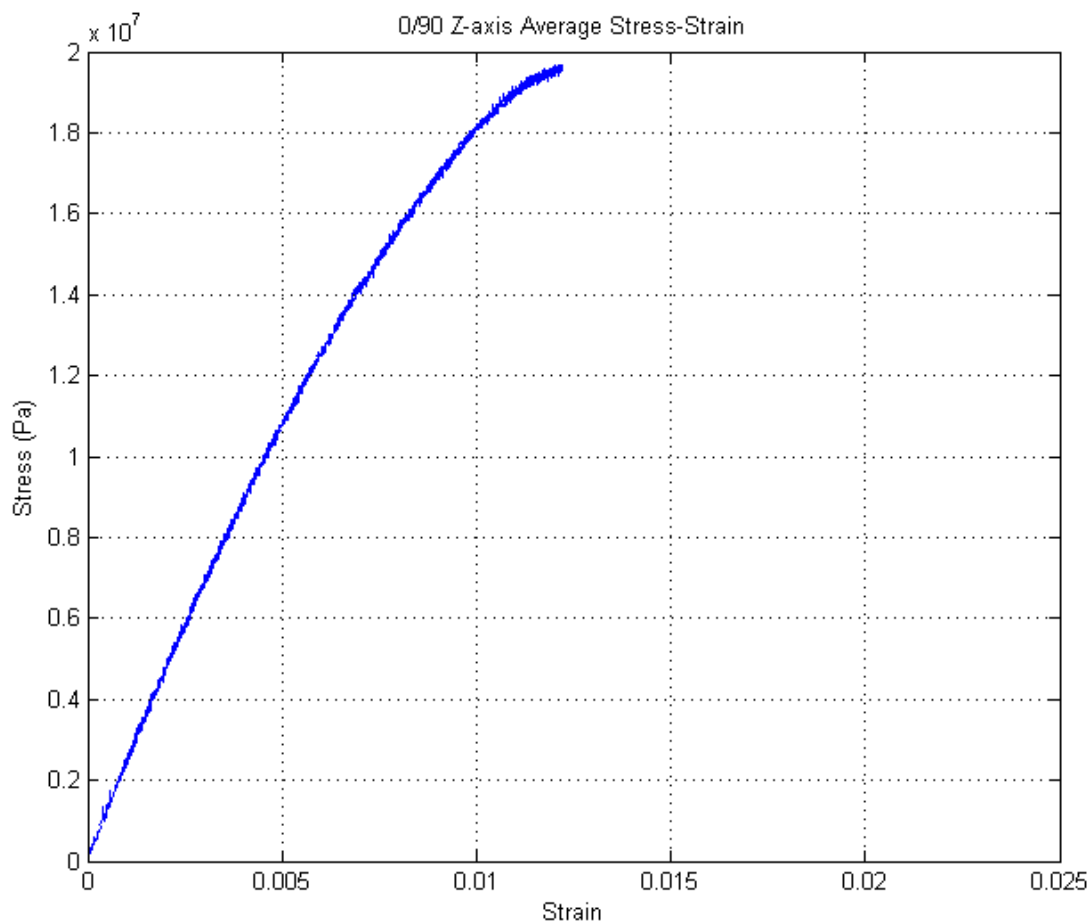


Figure 2-26 Stress-Strain Curve for 0/90 Raster Z-Axis Orientation





Figure 2-27 0/90 Z-Axis Oriented Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)

### *Quasi Isotropic Raster Printed in the Z-Axis Orientation*

The stress-strain curve is included in Figure 2-28 where the Young's Modulus is 1875MPa and its UTS is 20.05MPa, a 6.25% and 4.02% decrease from the Y-Axis oriented parts respectively. The Quasi Isotropic dogbone is no different from its counterparts printed in the Z-Axis orientation, it too exhibits brittle fracture. The stress-strain curve remains in a fairly linear elastic pattern during the duration of the experiment and the fracture picture in Figure 2-29 also demonstrates the brittle nature of this part. Interestingly, this part fractured at the symmetry line between plies where the raster orientation is the same. It would be an interesting study to determine where the weakest bonding between layers is in patterns such as the Quasi Isotropic but it is unfortunately outside the scope of this project.

There was sufficient strain data from the rosette to determine Mohr's circle before material failure (shown in Figure 2-30). The data is reliable as the cleavage of the dogbone occurred very

close to the strain gauge rosette. The Maximum Shear Strain is 0.006 and the Maximum Normal Strain is 0.009. This strain data will not be used to validate an ANSYS model due to time restrictions, but we provide the data for future researchers to develop the modeling techniques.

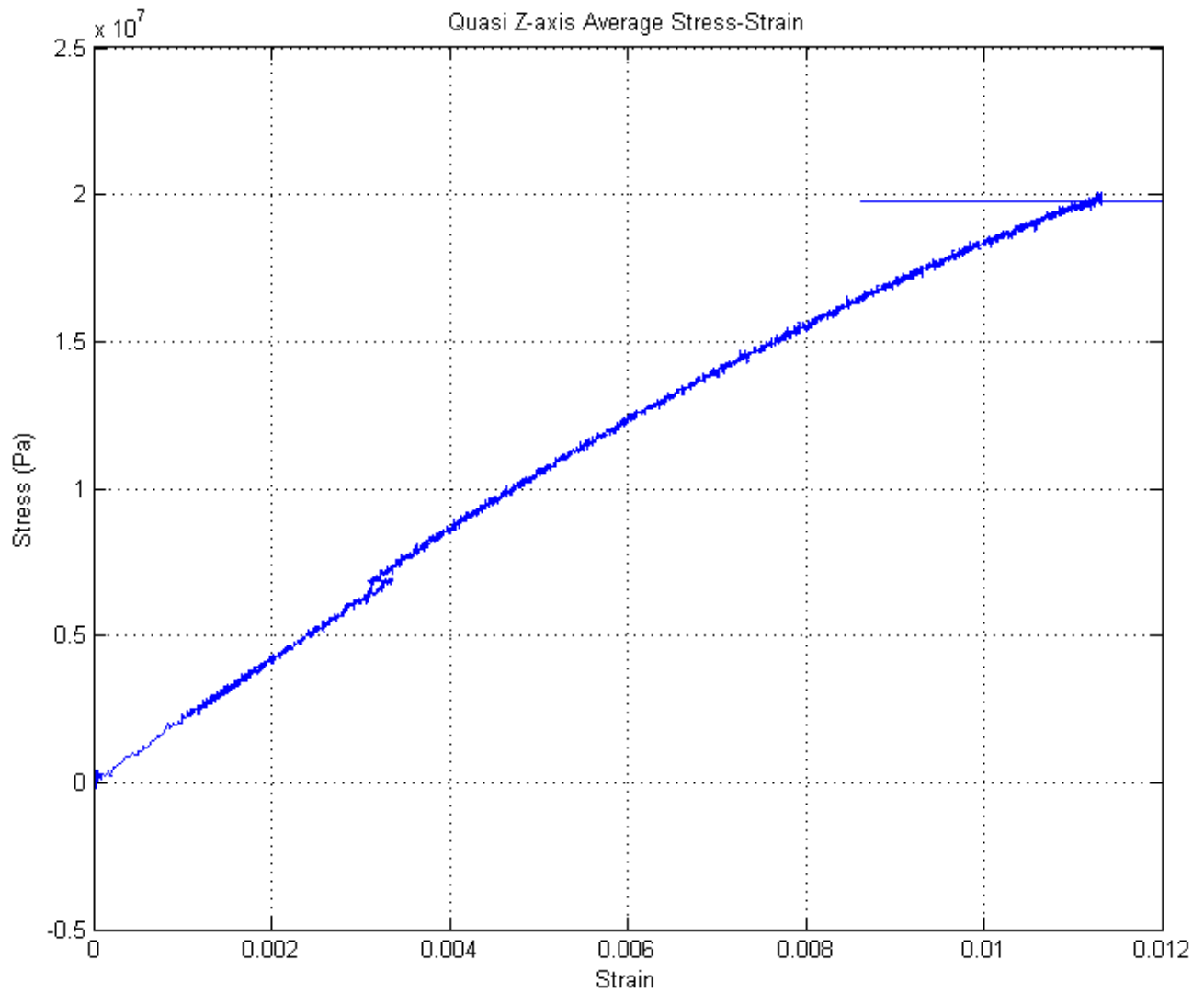


Figure 2-28 Stress-Strain Curve for Quasi Isotropic Raster Z-Axis Orientation

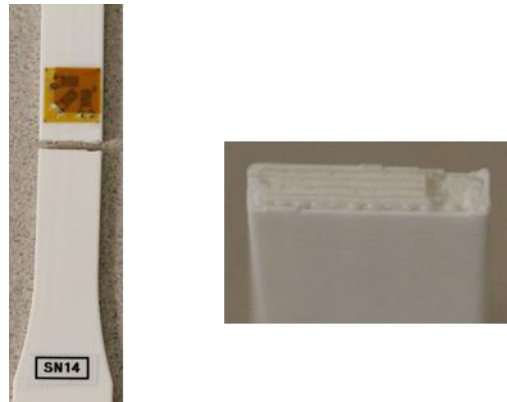


Figure 2-29 Quasi Isotropic Z-Axis Oriented Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)

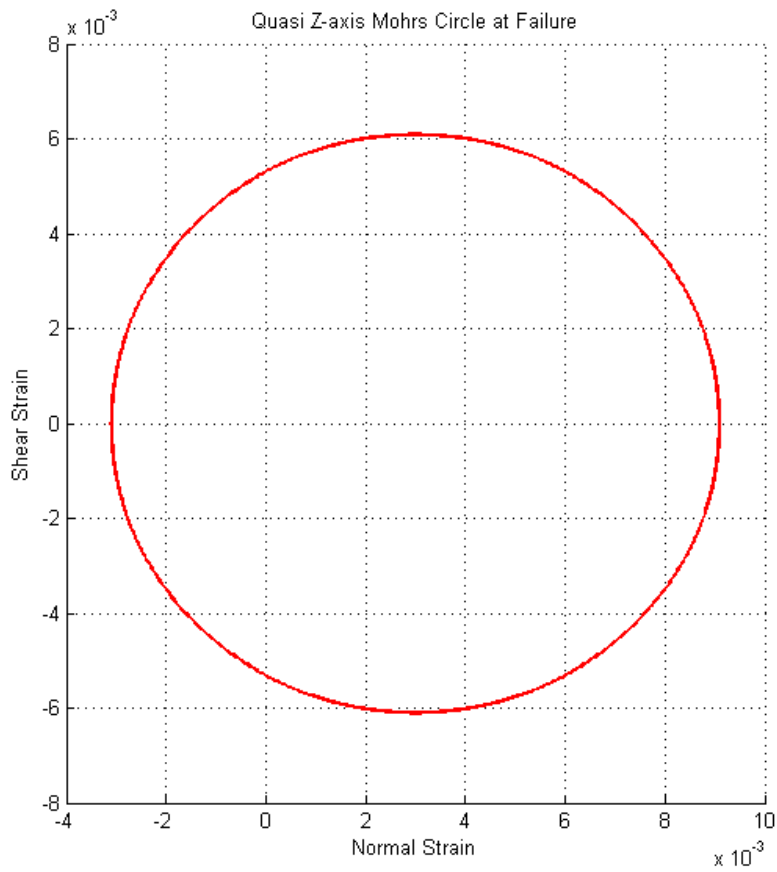


Figure 2-30 Quasi Isotropic Raster Z-Axis Orientation Mohr's Circle at Material Failure

### *Comparing the Z-Axis Oriented Dogbones*

Each of the Z-axis oriented dogbones exhibited a brittle fracture no matter the raster orientation. All of the dogbones were also significantly weaker in terms of their Ultimate Tensile

Strength which is to be expected as the fracture mechanism is the strength of bonding between layers. The 45/-45 had the greatest UTS out of the three but it was also the most brittle where the 0/90 raster was the most ductile of the three. This could be a possible effect of the packing as discussed in 2.5.1 where the effective void is inherently less in a 0/90 arrangement and any flattening will further help in creating a more solid product. Also, the voids act as stress concentrators and in this test we are pulling directly against the voids; this follows into the logic that the 0/90 arrangement, with the lesser total void area, should be able to maintain a higher strain before failure as the severance will take longer to propagate.

#### 2.4.4 Number of Plies for 45/-45 and Quasi Isotropic

In this section we will discuss the investigation into the number of plies and how it affects the material properties along with an assessment into the viability of using single-ply arrangements for end-products. The ability to lessen weight in an end-use product is of interest to the Laboratory and helped initiate this portion of the investigation.

##### *45/-45 Raster n = 1*

The 45/-45 raster at a single play thickness was only three layers thick. Qualitatively, the dogbone was moderately flexible and difficult to remove from the printing board without fracture; in total, four of these were printed but only two were successfully removed from the board. There are two fundamental applications that single ply may be useful within: flexibility and/or weight reduction. The stress-strain curve is shown in Figure 2-31 where the data is not as clean as other test articles tested in tension. This may be due to the roughness of the surface where the strain rosette gauge was applied as this article appeared to more porous than all other dogbones. Without the weight from many other layers to induce flattening, the beads remain in their regular stacked pattern without a deformation along the minor axis. The Young's Modulus is estimated to be 4000MPa which is a 66% increase from the reported material properties and its UTS is 15.4 MPa, a 57% decrease from the Stratasys reported value. The stress-strain diagram also shows a highly brittle fracture which the test article corroborates in Figure 2-32. There are two important observations to take from this test: single plies are extremely brittle and at this scale, the raster direction will dictate the fracture pattern. In Figure

2-33 Maximum Normal Strain is just below 0.004 and its Maximum Shear Strain is 0.0025, a 60% decrease and a 58% decrease from the Y-axis Oriented dogbone respectively. This data will be used with caution as the fracture was below the rosette strain gauge. The reason behind the assertion that the raster direction dictates the fracture pattern is that there is an obvious 45° sever meaning that there was not a strong bond between beads and since there was one more +45° raster it favored failing at that angle.

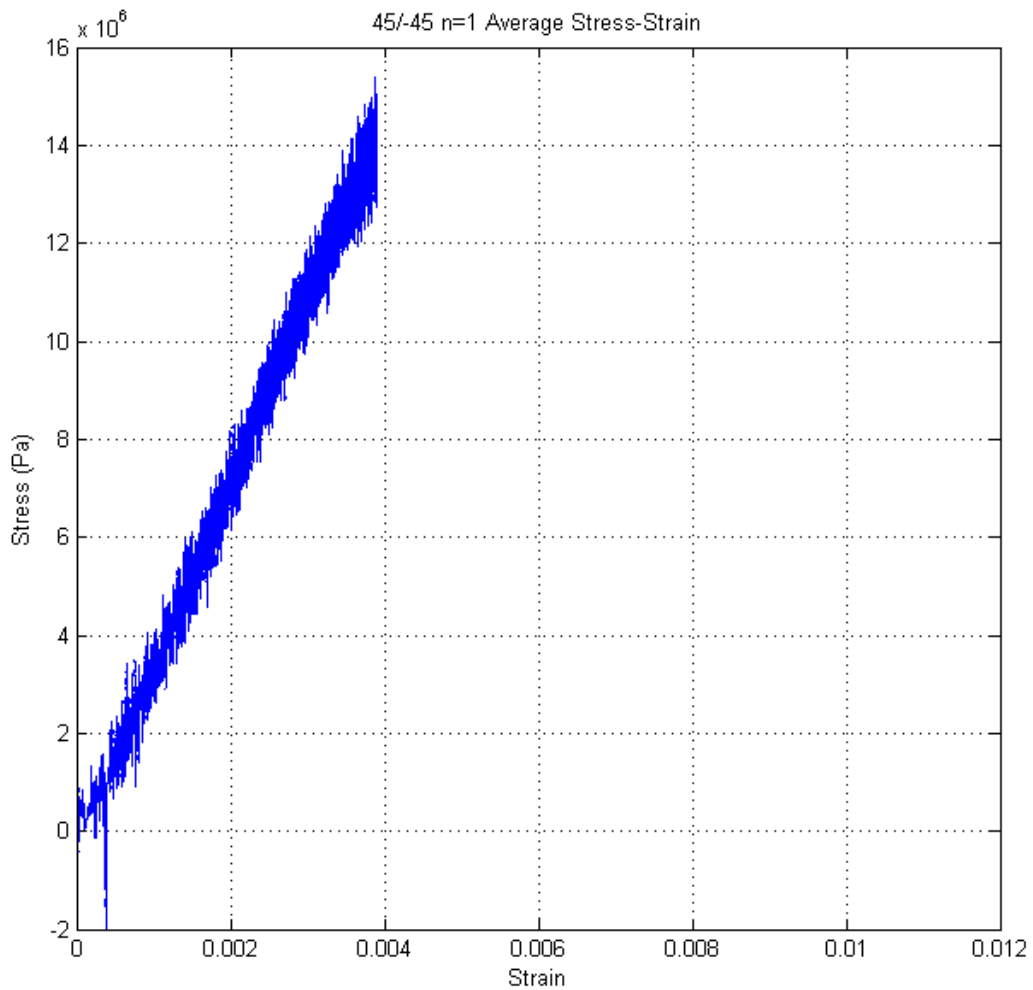


Figure 2-31 Stress-Strain Curve for 45/-45 Raster n = 1

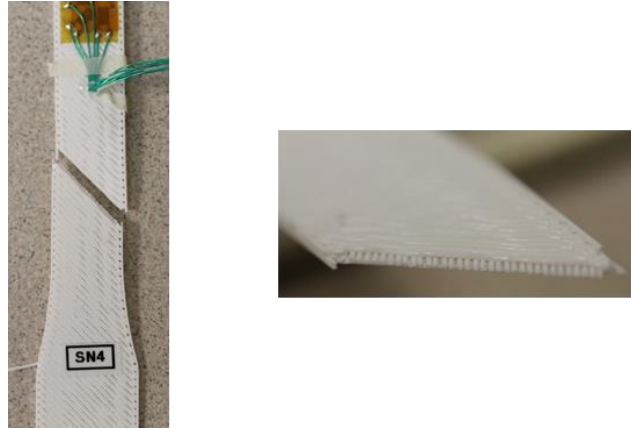


Figure 2-32 45/-45 n = 1 Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)

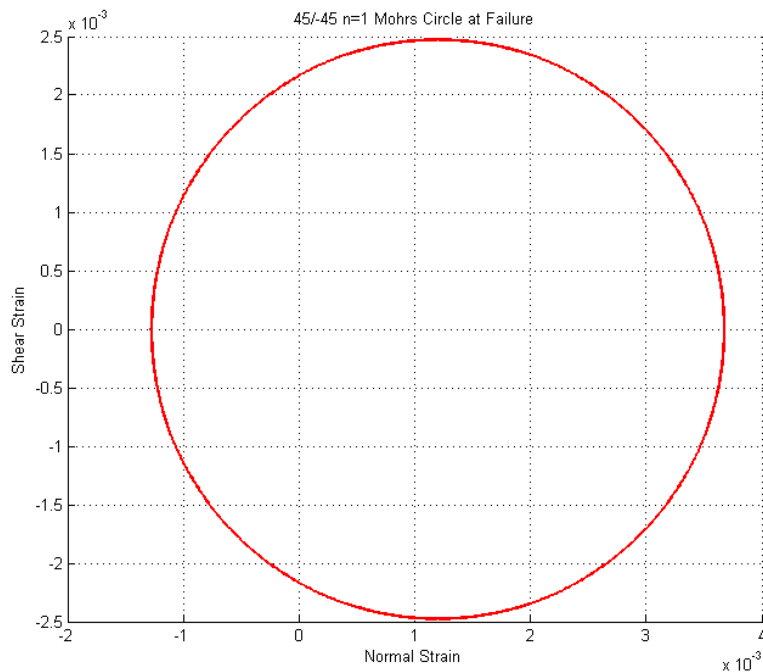


Figure 2-33 45/-45 n = 1 Mohr's Circle at Material Failure

### 45/-45 Raster n = 6

The 45/-45 dogbone at a six-ply thickness shows some interesting characteristics. From the stress-strain curve in Figure 2-34, Young's Modulus is calculated to be 2167MPa and the UTS is 22.95MPa which is a 9.7% and 3.8% decrease respectively from the 45/-45 Y-axis oriented dogbone. This test article appears to have a very large plastic region that is about a factor of 3 larger than the Y-axis oriented dogbone. It is difficult to assert that this printing arrangement is

much more ductile than the Y-axis dogbone as it is only 6 layers less than that printing arrangement. Further testing with both types of dogbones would be necessary to see if this is a common trend. From Figure 2-40, it is evident that the fracture indicates a ductile fracture and not a brittle one. However, for such a maximum strain to exist, we also expect that the mid-section of the dogbone should have exhibited significant necking. The only other explanation for this phenomenon is that the test article slipped in the grips of the Instron during testing. The strain data from the gauge itself was lost due to a premature failure and strain data was taken from the Instron DAQ. It would not have recognized the slippage of the dogbone as its only data input was the displacement of the grips assuming that the article was secure. This may have resulted in the 3-fold increase for the maximum strain and the false reading of a prolonged plastic region. Unfortunately, the data from the orthogonal and 45° gauge also failed with large unexplainable discontinuities in the data eliminating the chance for being able to describe the strain state at failure.

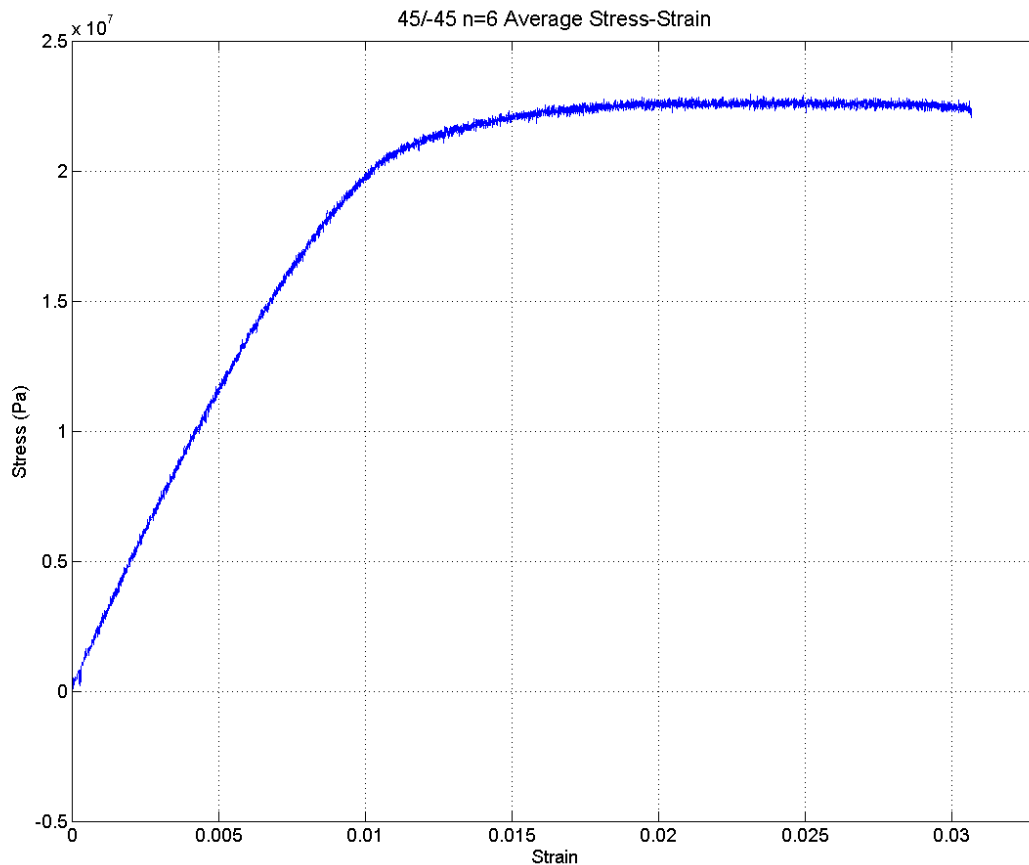


Figure 2-34 Stress-Strain Curve for 45/-45 Raster n = 6

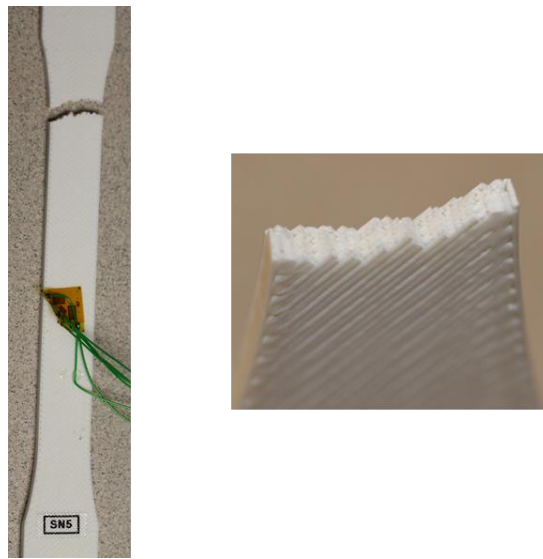


Figure 2-35 45/-45  $n = 6$  Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)

### *Quasi Isotropic Raster $n = 1$*

The single-ply Quasi-Isotropic dogbone behaved comparatively to the Y-axis oriented one most likely due to the fact that it was only one-ply less. The stress-strain curve in Figure 2-36 is almost identical to Figure 2-20. With a Young's Modulus of 2031MPa and UTS of 21.48MPa, there is only a 1.55% and 2.82% increase respectively from the Y-axis dogbone. This article appears to have a smaller plastic region than the Y-axis dogbone which seems to follow in the pattern of reduced plies being more brittle. The fracture pattern shown in Figure 2-37 appears to be only minimally ductile.

The strain data is considered to be fairly accurate for this test article as fracture occurred underneath the rosette strain gauge. Figure 2-33 provides the principal strain data in Mohr's circle with a Maximum Normal Strain of approximately 0.0145 and a Maximum Shear Strain of about 0.01.



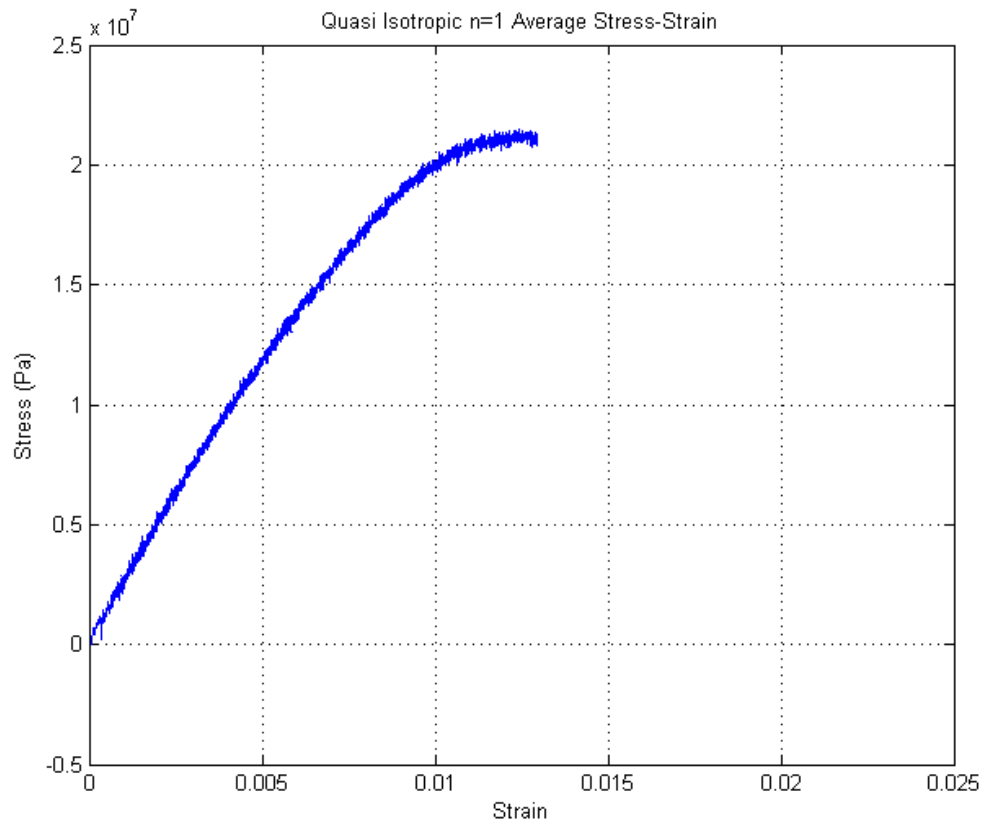


Figure 2-36 Stress-Strain Curve for Quasi Isotropic Raster n = 1

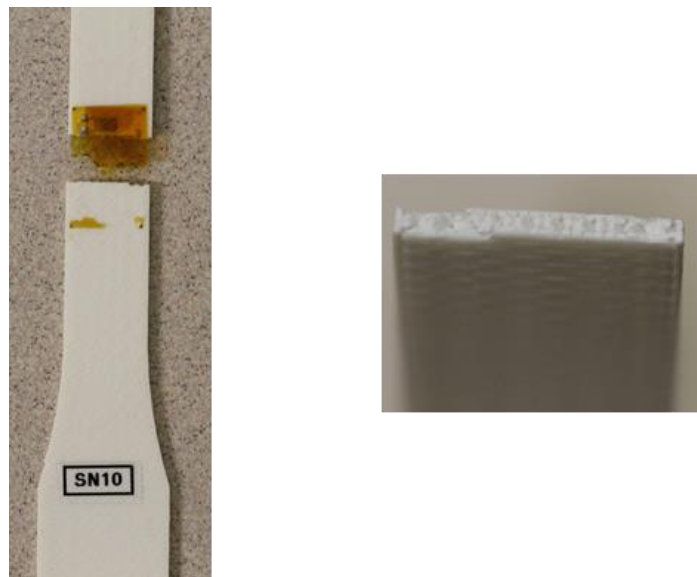


Figure 2-37 Quasi Isotropic n = 1 Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)

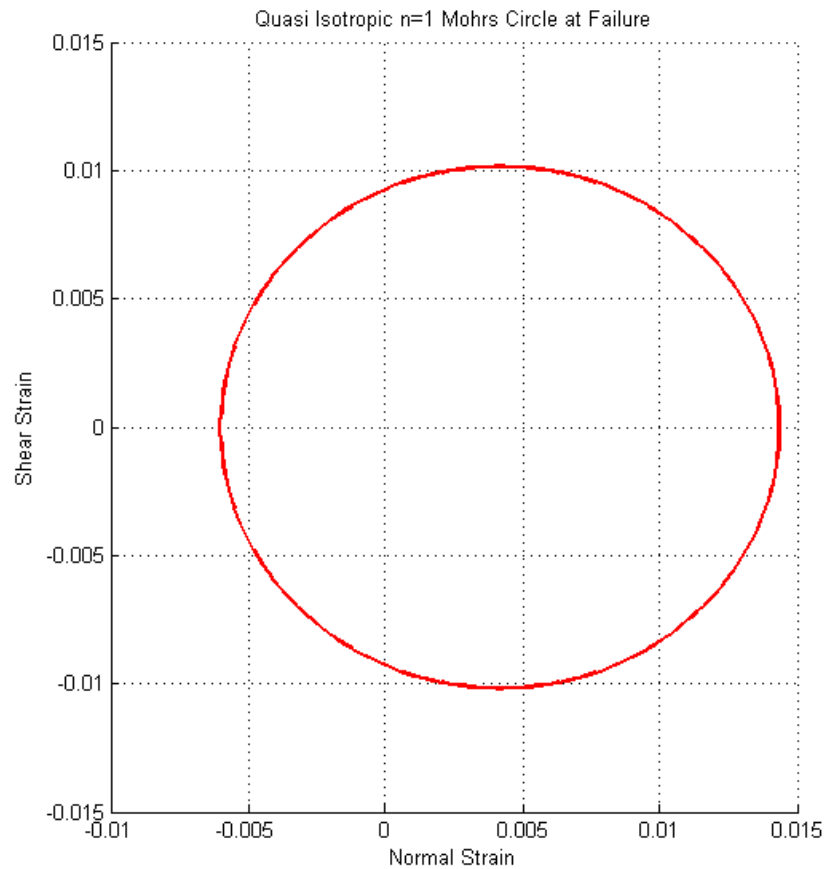


Figure 2-38 Quasi Isotropic n = 1 Mohr's Circle at Material Failure

### *Quasi Isotropic Raster n = 6*

The six-ply Quasi Isotropic Dogbone was the thickest one we tested as a part of this investigation. The stress-strain curve in Figure 2-39 appears to behave like a textbook curve with an elastic, stress-relaxation, and plastic deformation region. The Young's Modulus is 2233MPa and the UTS is 20.68MPa showing an 11.65% increase and a 1% decrease respectively from the Y-axis dogbone. It appears that the strength of the Quasi Isotropic dogbone remains relatively consistent no matter the thickness. The fracture pattern in Figure 2-40 only appears to be moderately ductile with no evident necking. A pattern emerges in the fracture of Quasi Isotropic FDM printed parts in that as compared to a 45/-45 raster, they tend to fracture at a low angle. This implies that the components of stress must be dominant parallel and anti-parallel to the loading. Strain data was available for all components investigated and Mohr's

Circle at failure is presented in Figure 2-41. The Maximum Normal Strain is 0.055 and the Maximum Shear Strain is 0.048. This is a significant increase from the Y-axis oriented Quasi Isotropic with a 450% increase and 210% increase for the Maximum Normal Strain and Shear Strain respectively.

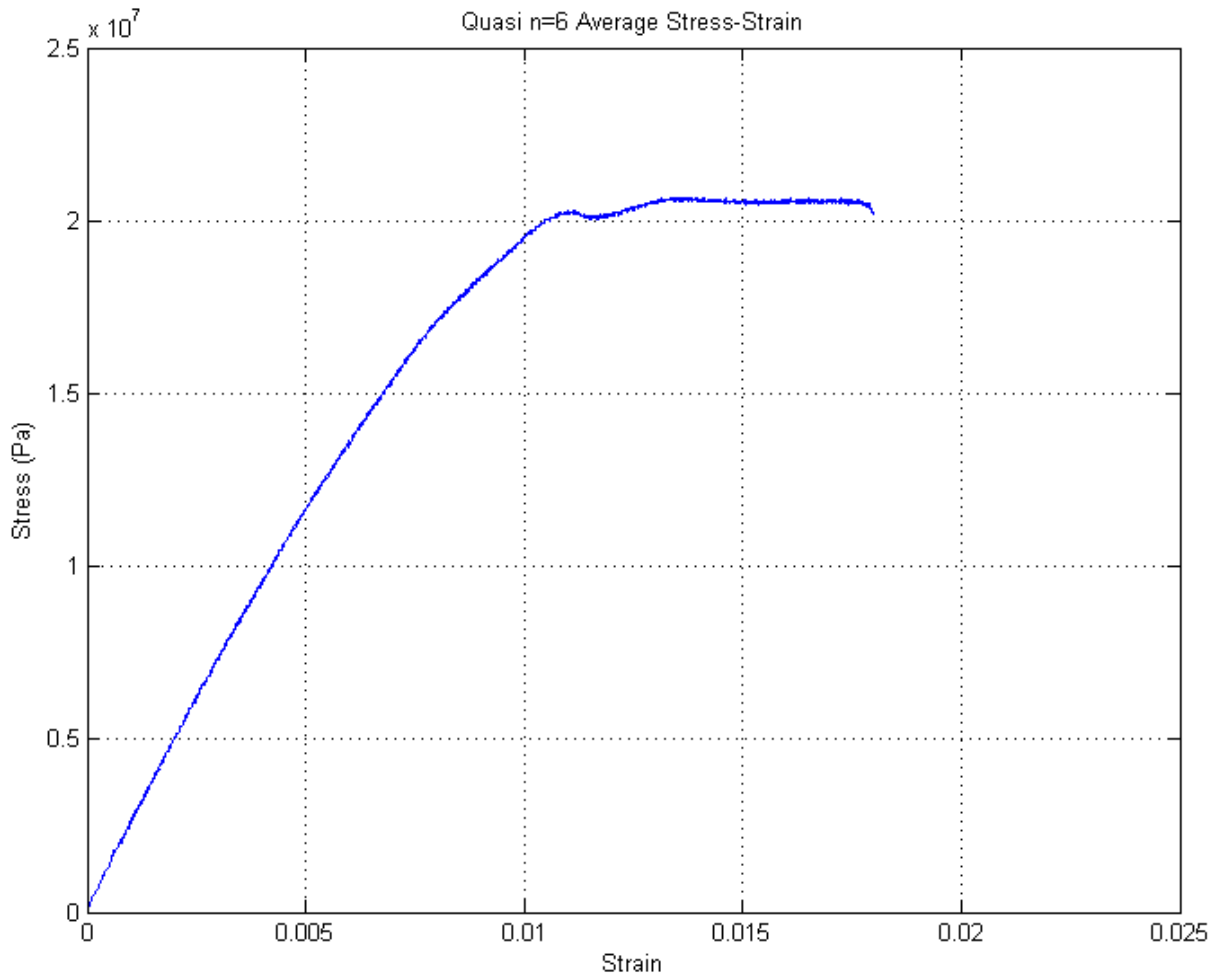


Figure 2-39 Stress-Strain Curve for Quasi Isotropic Raster n = 6

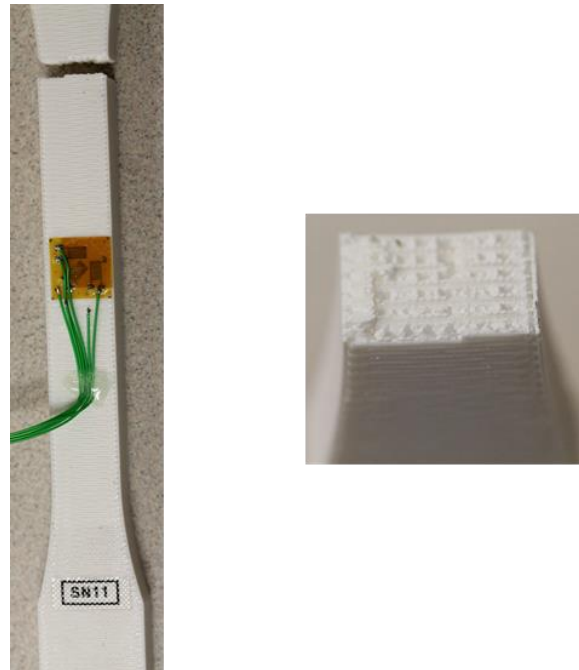


Figure 2-40 Quasi Isotropic n = 6 Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)

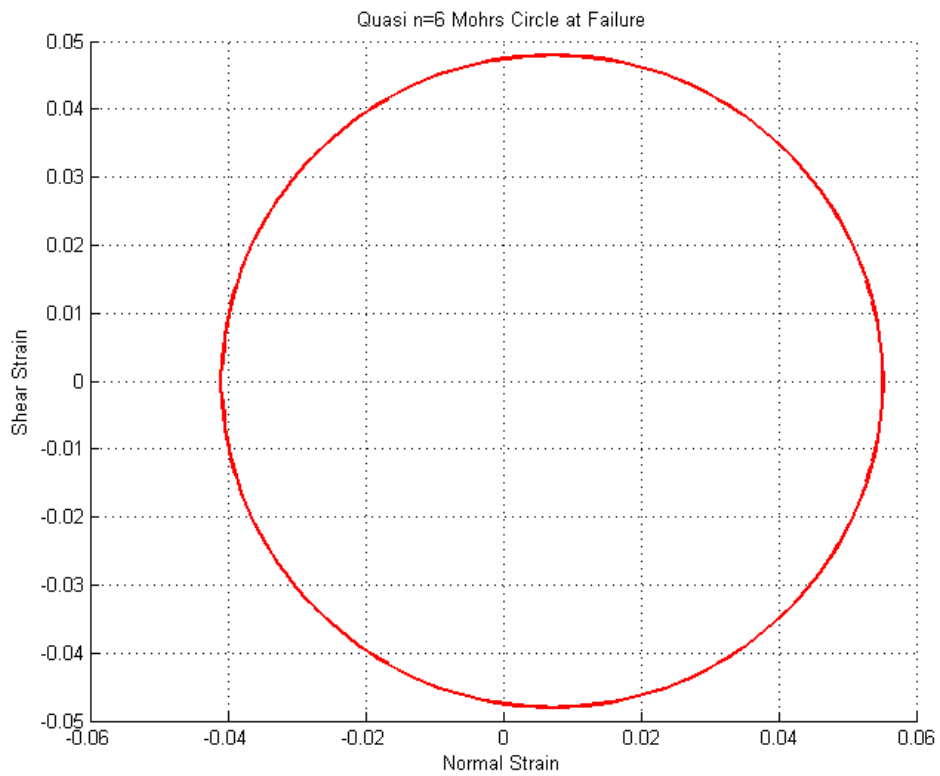


Figure 2-41 Quasi Isotropic n = 6 Mohr's Circle at Material Failure

### 2.4.5 Bead Width Alterations

Bead width has been shown in literature to make minimal to no difference in the tensile strength of FDM parts. In this section, we will present the results on the large bead dogbones. It should be noted that the strain data from the small bead articles was not recoverable for data analysis as the gauges failed prematurely during the experiment due to the fragile nature of the leads.

#### *45/-45 Raster with Large Bead*

The 45/-45 Large Bead dogbone exhibits both an elastic and significant plastic region for deformation during tensile testing. From the stress-strain curve in Figure 2-42, Young's Modulus is calculated to be 2448MPa and the UTS is 23.42MPa which presents a 2% increase and 1.8% decrease from the Y-axis dogbone respectively. This means that there is virtually no change as a result of bead width in the strength of the test article. The large bead dogbone does appear to be more ductile than the Y-axis one as its plastic dogbone with a strain of approximately 0.024 as opposed to 0.011. It is evident from Figure 2-43 that a ductile fracture is occurring, and comparing this fracture with that of the Y-axis dogbone in Figure 2-15, we can clearly see a more violent fracture in the large bead test article. It appears that printing with a large bead does increase the ductility of the part.

Strain data from the gauge aligned to the 45° was prematurely terminated by the severance of leads from the strain gauge pads. Therefore, the full strain state of the test article at failure could not be constructed.

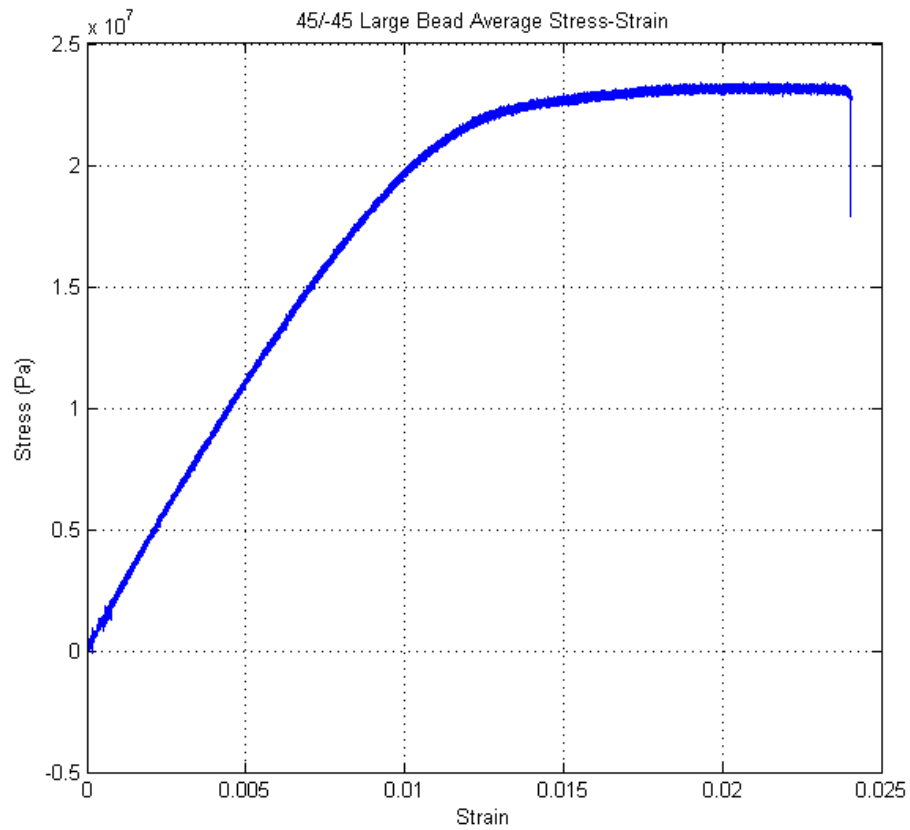


Figure 2-42 Stress-Strain Curve for 45/-45 raster with large bead

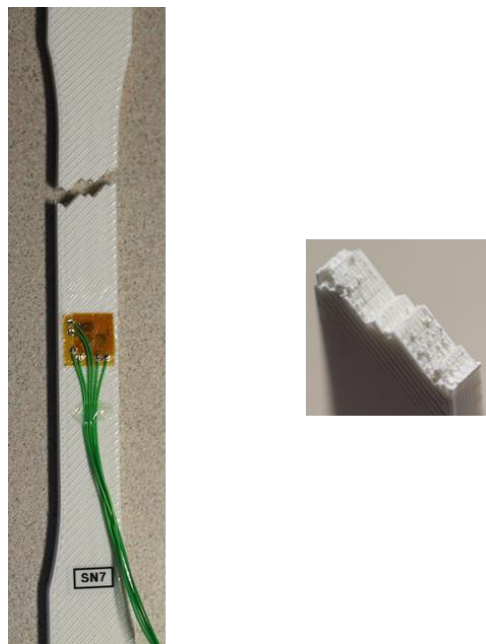


Figure 2-43 45/-45 Large Bead Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)

### *Quasi Isotropic Raster with Large Bead*

The large bead did make changes to the material properties of the Quasi Isotropic raster. In Figure 2-44, the stress-strain curve shows about a 33% increase in the strain before fracture from the Y-axis dogbone. Young's Modulus is calculated to be 2464MPa and the Ultimate Tensile Strength is 20.96MPa a 23% increase and 5% increase respectively from the Y-axis dogbone. Unlike the 45/-45 raster, it appears that the bead width does have an effect on the Quasi Isotropic dogbone material properties. The reason behind this is unclear, but it could possibly be affecting the packing density of the dogbone. Observing these test articles under a microscope might provide an answer to a change in internal geometry from these larger beads against the regular bead size. Like the 45/-45 large bead dogbone, this test article also shows an increase in ductility not only in the stress-strain curve but in the fracture pattern shown in Figure 2-45. This fracture appears to be more ductile than the fracture of the Y-axis dogbone shown in Figure 2-21.

Data was available from all of the gauges in the rosette and strain data in the graphical form of Mohr's Circle is presented in Figure 2-46. The Maximum Shear Strain is 0.055 and the Maximum Normal Strain is approximately 0.064. This is significantly greater than the regular dogbone and appears to conclude that the larger bead dogbones are able to withstand high strain states.

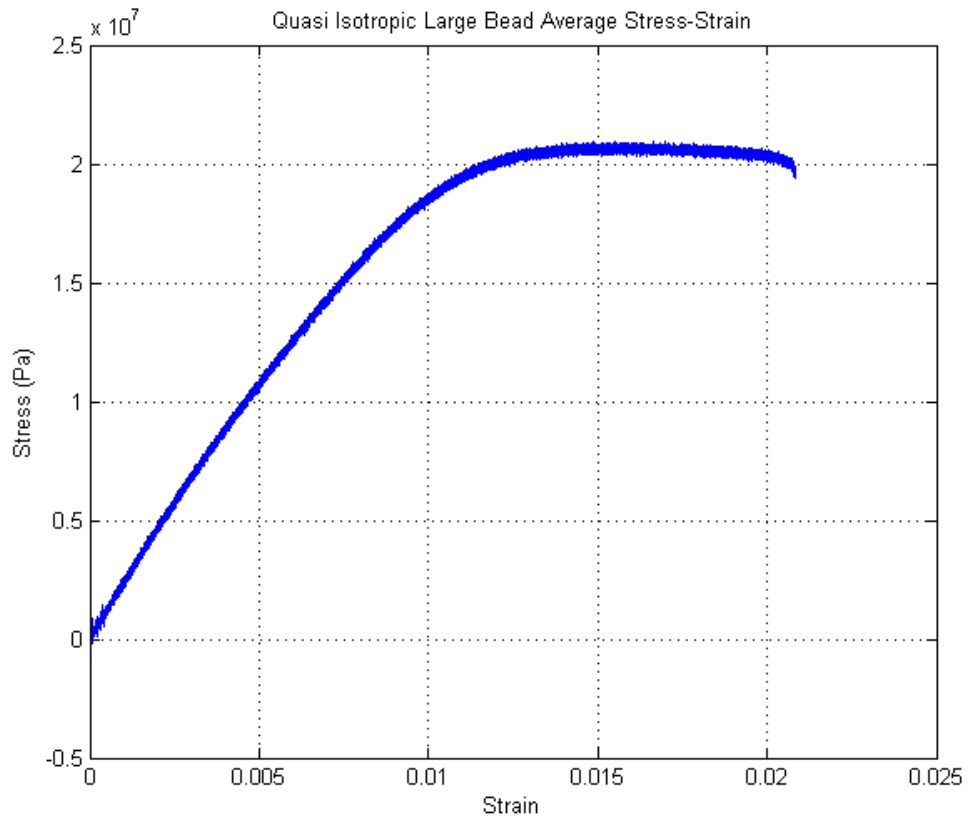


Figure 2-44 Stress-Strain Curve for Quasi Isotropic raster with Large Bead

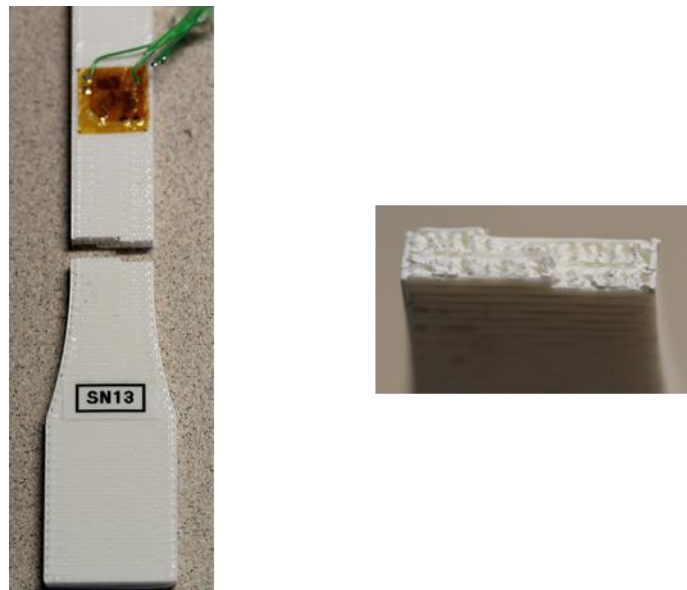


Figure 2-45 Quasi Isotropic Large Bead Test Article after destructive Tensile Testing (Left) Fracture Pattern (Right)



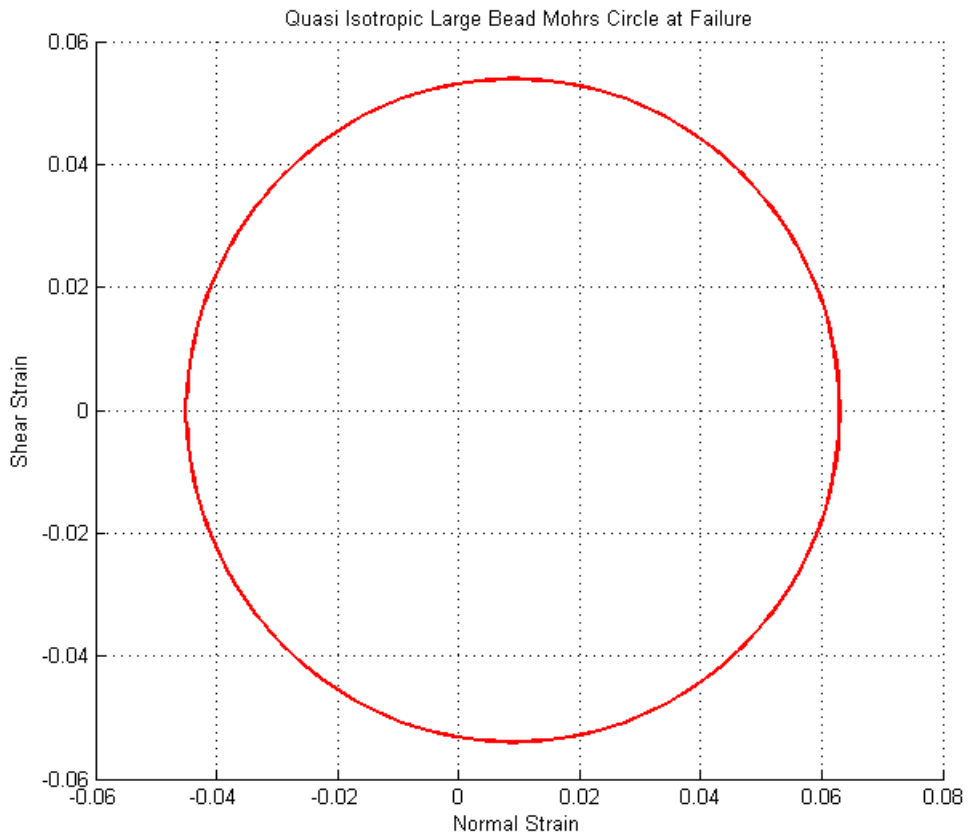


Figure 2-46 Quasi Isotropic Large Bead Mohr's Circle at Material Failure

### 2.4.6 Evaluation of the Ultimate Tensile Strength

Load data was taken using the Instron DAQ and uploaded into an Excel file where the Maximum Load for every Test Article that successfully fractured was calculated. Table 2-4 presents the maximum load data along with the calculated Ultimate Tensile Strength (UTS).

Table 2-4 Ultimate Tensile Strength

Name (Serial)	Maximum Load (lb)	Volume (m <sup>3</sup> )	Height (m)	UTS (MPa)
45/-45 Z-Axis (1)	242.9814	8.66656E-6	0.1651762	20.60
45/-45 Y-Axis (2)	281.4304	8.66656E-6	0.1651762	23.86
0/90 Y-Axis (3)	245.7016	8.66656E-6	0.1651762	20.83
45/-45 n = 1 (4)	28.7396	1.36840E-6	0.1651762	15.40
45/-45 n = 6 (5)	185.2879	5.92975E-6	0.1651762	22.95

45/-45 Lg. Bead (7)	276.1885	8.66656E-6	0.1651762	23.42
Name (Serial)	Maximum Load (lb)	Volume (m <sup>3</sup> )	Height (m)	UTS (MPa)
Quasi Y-Axis (8)	208.9731	7.40367E-6	0.1664208	20.89
Quasi n = 1 (10)	107.4278	3.70183E-6	0.1664208	21.48
Quasi n = 6 (11)	620.5544	22.20840E-6	0.1664208	20.68
Quasi Lg. Bead (13)	209.6109	7.40367E-6	0.1664208	20.96
Quasi Z-axis (14)	200.5876	7.40367E-6	0.1664208	20.05
0/90 Z-axis (15)	232.192	8.66656E-6	0.1651762	19.69

The UTS was calculated by using

$$UTS = \frac{Max\ Load\ (lb) * 4.4482}{\frac{Volume\ (m^3)}{Height\ (m)}}\ (Pa)$$

Equation 2-4 Calculating the Ultimate Tensile Strength

where, the average cross-sectional area represented by the  $\frac{volume}{height}$  term. The value of 4.4482 is used to convert from English to SI units. Our group cautions that this data is only the result of one data set for each different type of Dogbone presented. However, there are some evident trends within the rasters; for each Dogbone that is printed in the Y-axis orientation, the UTS is greater than its counterpart printed in the Z-axis. This most likely has to do with the fact that fracture dynamics are different between the two articles. A dogbone printed in the Z-axis orientation will break between printed layers whereas one in the Y-axis orientation will sever along the beads. This means that the bonding strength between layers is weaker than that of the beads which is expected.

#### 2.4.7 Summary of Tensile Material Properties for Tested Dogbones

Table 2-5, Table 2-6, Table 2-7, and Table 2-8 present the Ultimate Tensile Strength, Young's Modulus, Maximum Normal Strain, and Maximum Shear Strain for the Y-Axis Orientation, Z-Axis Orientation, Ply Thickness, and Bead Width experiments respectively. The main observations

are that the 45/-45 raster has a higher Ultimate Tensile Strength overall but the Quasi Isotropic Raster has a higher ductility.

**Table 2-5 Summary of Tensile Data for Y-Axis Orientation**

Name	Maximum Load (lb)	UTS (MPa)	E (MPa)	Max $\epsilon$	Max $\gamma$
45/-45 Y-Axis	281.4304	23.86	2400	0.01	0.006
Quasi Y-Axis	208.9731	20.89	2000	0.0155	0.01
0/90 Y-Axis	245.7016	20.83	2300	0.015	0.007

**Table 2-6 Summary of Tensile Data for Z-Axis Orientation**

Name	Maximum Load (lb)	UTS (MPa)	E (MPa)	Max $\epsilon$	Max $\gamma$
45/-45 Z-Axis	242.9814	20.60	2252	0.00102	0.0007
Quasi Z-Axis	200.5876	20.05	1875	0.009	0.006
0/90 Z-Axis	232.192	19.69	2061	0.008	N/A

**Table 2-7 Summary of Tensile Data for Ply Thickness**

Name	Maximum Load (lb)	UTS (MPa)	E (MPa)	Max $\epsilon$	Max $\gamma$
45/-45 n = 1	28.7396	15.40	4000	0.004	0.0025
45/-45 n = 6	185.2879	22.95	2167	N/A	N/A
Quasi n = 1	107.4278	21.48	2031	0.0145	0.01
Quasi n= 6	620.5544	20.68	2233	0.055	0.048

**Table 2-8 Summary of Tensile Data for Large Bead**

Name	Maximum Load (lb)	UTS (MPa)	E (MPa)	Max $\epsilon$	Max $\gamma$
45/-45 Lg. Bead	276.1885	23.42	2448	0.024	N/A

Quasi Lg. Bead	209.6109	20.96	2464	0.064	0.055
----------------	----------	-------	------	-------	-------

## 2.5 Theoretical Modeling of FDM Manufactured Parts

This section provides an extension upon the work by Li et al. (2002) to include all types of raster angles used in this study. The geometry of stacking patterns will be shown in idealized and non-idealized cases. Following this, a composite laminate theory will be presented to provide the fundamental mathematics used in an analytical approach to fully define the relation between stress and strain.

### 2.5.1 Orientation and its Effect on Stacking

It is well known that FDM does not produce a laminate composite in the traditional sense; the 3D printer leaves the fibers of ABS-M30 without a background matrix to create a smooth layer of stackable plies. In the simplest form, an FDM printed part has a cross-section of an ellipse perfectly stacked upon one another as shown in Figure 2-47. This represents the baseline where at a cut in the material the toolpaths are unidirectional with one another and have minor axis  $b$  and major axis  $a$ . The width at which the ellipses touch one another side by side is represented by  $2y$ . Li et al. (2002) introduced two variables:  $\rho_1$ , the area void density and  $\rho_2$ , the linear void ratio. The area void density is intended to measure the percent of area gap that the FDM part has between toolpaths while the linear void ratio is intended to describe flatness between two adjacent roads. Figure 2-47 circumscribes the ellipses in a rectangle of dimensions  $2b$  by  $2a$  to facilitate an ease of understanding between the various stacking sequences. From geometry, the area void density and the linear void ratio are given by:

$$\rho_1 = 1 - \frac{ab\pi}{2a * 2b}$$

Equation 2-5 Area Void Density for Unidirectional

$$\rho_2 = \frac{b - y}{b}$$

Equation 2-6 Linear Void Ratio for Unidirectional

For the purposes of this project, we will not be considering the effect of overlapping from roads adjacent to one another as all Insight toolpaths were set as a zero spacing between roads.

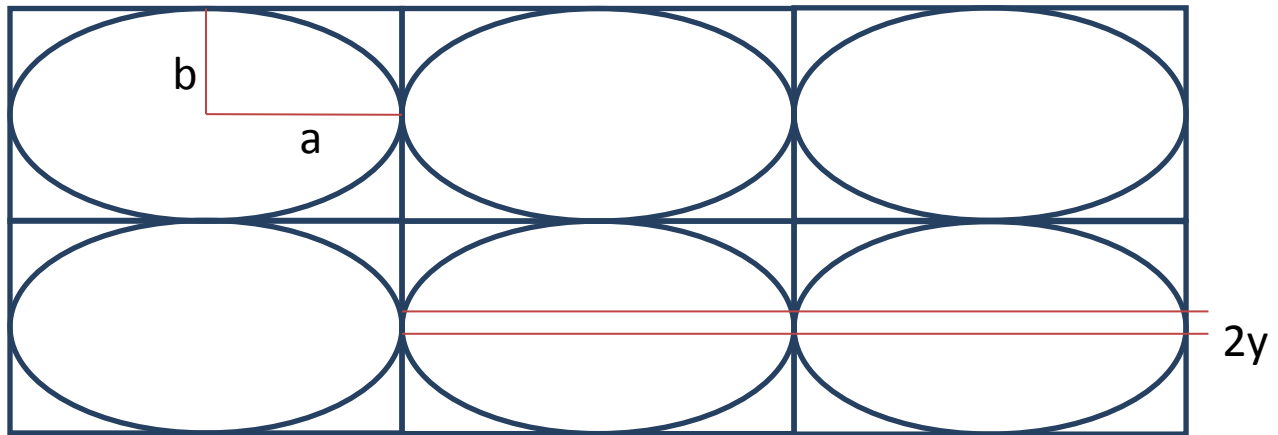


Figure 2-47 Unidirectional, No flattening

This type of pattern can be seen not only in unidirectional roads (0/0 ... 90/90...ect.) but in 45/-45 raster orientations also. Using SolidWorks we created a visual of how the raster will change over cuts in the z-direction (coordinate set-up in Figure 2-48). Looking at a cross-section view at the midplane (Figure 2-49) we find that there is agreement between the models so we can use this basic case to make approximations of the behavior.

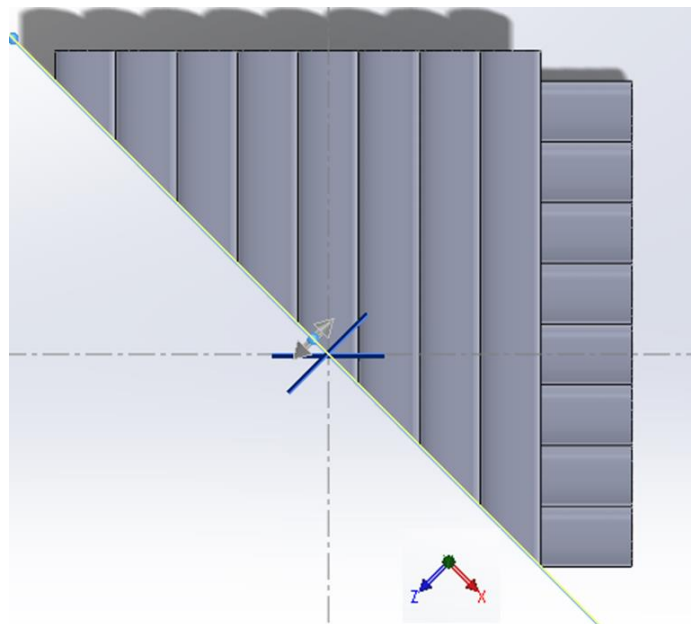


Figure 2-48 Coordinate set-up for 45/-45 raster

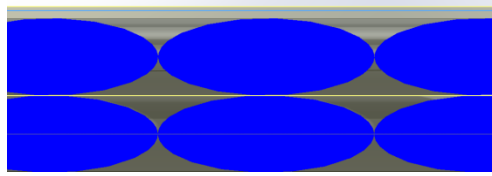


Figure 2-49 Midplane cross-section view 45/-45

The 45/-45 raster orientation does not remain as simple as the unidirectional case. There is a phase shift between the layers where half cycle is governed by a distance of  $a * \cos(\theta)$  in the z-direction. In Figure 2-50 we see the visual effect of this shift and the different stacking pattern that has evolved. Theoretically, in this case we still maintain the same mathematics from Equation 2-5 and Equation 2-6 which is more readily apparent from looking at the geometry presented in Figure 2-51. The ellipses circumscribed in rectangles have not been deformed in the minor or major axis direction but only shifted to reflect a stacked pattern. This means that both the area void density and the linear void ratio have not been affected since the geometry is still fundamentally the same.

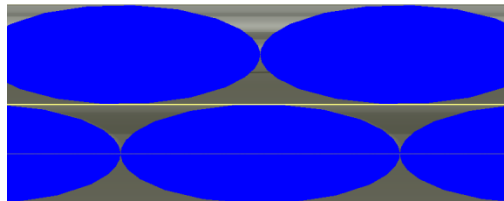


Figure 2-50 Stacking at half cycle for 45/-45

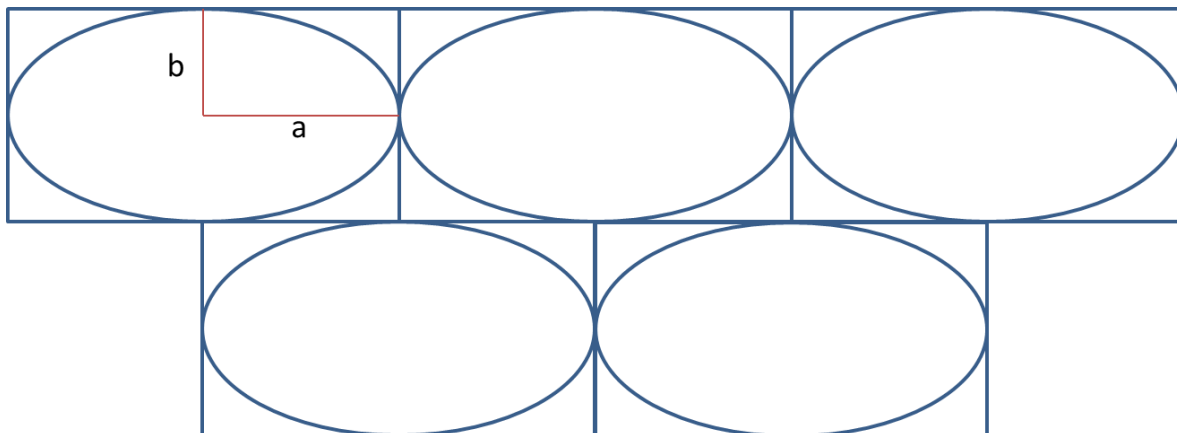


Figure 2-51 Stacked, no flattening

Li et al. (2002) also introduced the flattening effect which occurs due to the weight of other layers and from the settling of layers due to pliability from being heated near the crystalline temperature. This introduces interesting consequences for both a unidirectional and stacked configuration. Flattening means that there is an intersection between layers denoted by Li et al. (2002) as  $\delta$  to represent this distance of intersection. The area void density can be calculated using (Li et al., 2002)

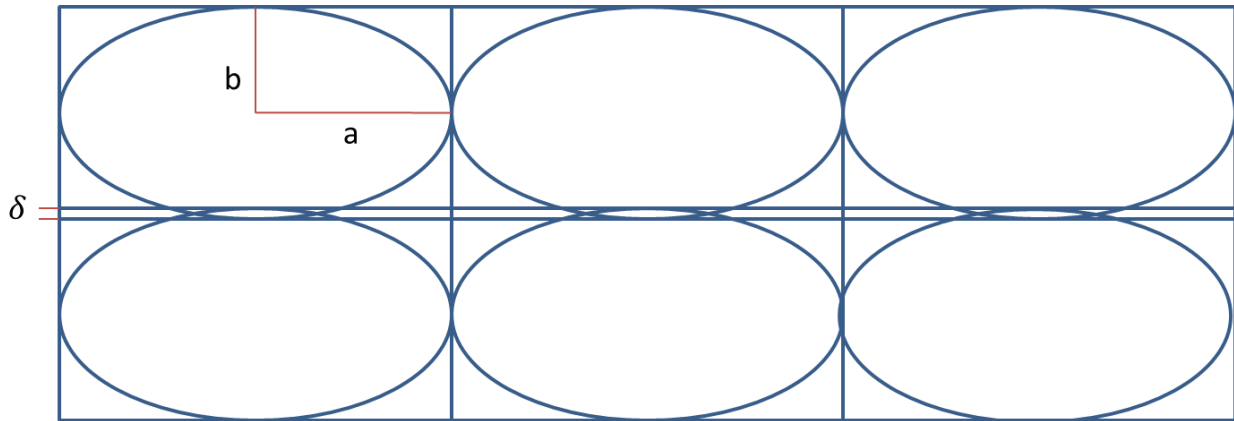


Figure 2-52- Unidirectional and Flattening

$$\rho_1 = 1 - \frac{ab\pi}{2a * (2b - \delta)}$$

Equation 2-7 Area Void Density for Unidirectional and Flattening

Similarly, the stacked pattern arrangement will also follow the same mathematics in Equation 2-7 as apparent from the geometry presented in Figure 2-53. The power of this relationship is important for looking at the different linear shifts along the major axis between layers and having the ability to determine the area void density for each.

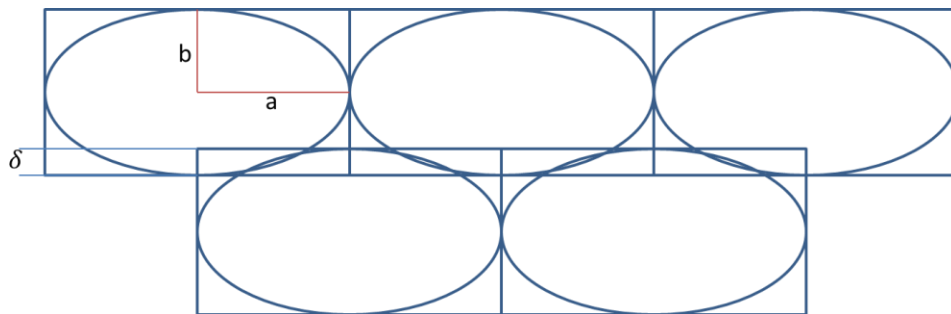


Figure 2-53 Stacked and Flattening

This brings us to an important fundamental understanding of how the stacking arrangement will affect the area void density when taking flattening into consideration. Figure 2-54 elucidates the preferable stacking when the layers are not unidirectional. The stacked arrangement demonstrates that a higher  $\delta$  value is needed to introduce flattening whereas in the unidirectional arrangement  $\delta$  must only be greater than zero. Referring to Equation 2-7, as  $\delta$  increases,  $\rho_1$  decreases meaning that the area void density approaches zero or in physical terms, the cross-section becomes more solid with fewer air gaps. It is expected that if there were no air gaps, then the part's material properties would also approach that of the bulk material.

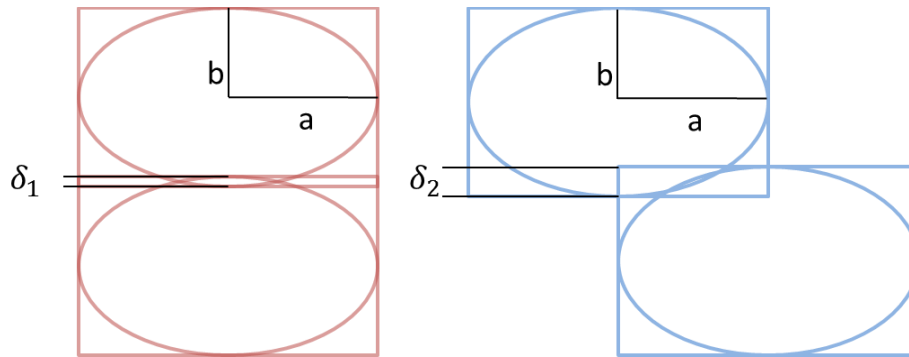


Figure 2-54 Unidirectional (red, left) & Stacked (blue, right)

Because a greater distance must be covered between layers in order to introduce flattening in the stacked case, it is also worthy to note that void area reduction can also happen without flattening as shown in Figure 2-55. We recommend as an approximation to use Equation 2-7 for calculating the area void density; this solution should only garner minor error.

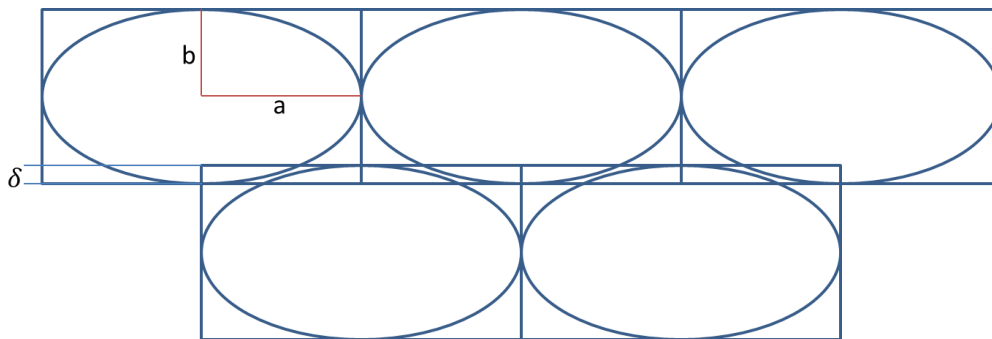


Figure 2-55 Stacked, area reduction without flattening



On the micro-scale, there will be different degrees of flattening for any non-unidirectional raster orientation. The complete solution of this would most likely depend on the thermal cycle for FDM manufacturing, bead width, number of layers, and elasticity. Figure 2-56 shows a basic outline of how the 45/-45 ply behaves in the Z'-Y plane without any form of flattening.<sup>2</sup> If we were to consider flattening here, there would be an expected sagging for the top layer in between the half ellipses causing a wavy pattern. This would invoke almost all of the cases we have defined in this section and a numerical solution would be most appropriate to fully define the micro-scale occurrences.

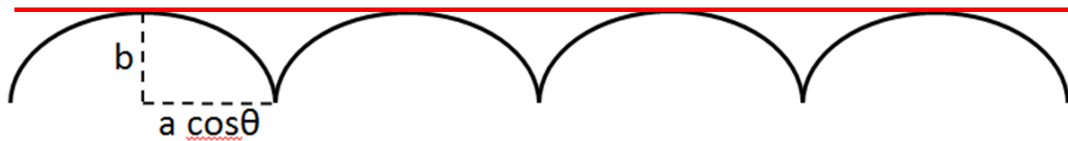


Figure 2-56 Z'-Y plane Bottom Layer (Black) Top Layer (Red)

We can generalize this case with the same geometry we would expect to find in orthogonal plies. Figure 2-57 shows the basic geometry we expect without flattening and the area void density and linear void ratio respectively for this case are described by

$$\rho_1 = 1 - 0.5 * \frac{ab\pi}{2a * 2b}$$

Equation 2-8 Area Void Density for Orthogonal Stack

$$\rho_2 = 0.5 * \frac{b - y}{b}$$

Equation 2-9 Linear Void Ratio for Orthogonal Stack

---

<sup>2</sup> The Z'-axis is rotated counterclockwise from the original Z-axis presented in Figure 2-48 so that X' and Z' are collinear with the bottom and top layers' direction respectively.

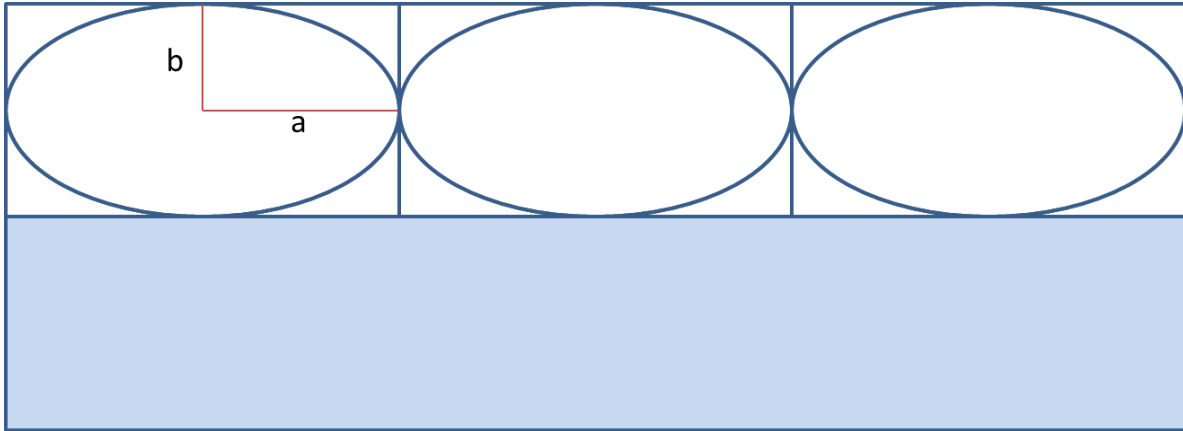


Figure 2-57 Orthogonal, no flattening

In the above, the factor of 0.5 is added to account for the packing difference.

Similarly, by modifying the case shown in Figure 2-52 for an orthogonal case, Figure 2-58 can be used to elucidate the geometry of flattening. The area void density for this case assuming that the wavy pattern previously described does not occur or that its effects are negligible is given by

$$\rho_1 = 1 - 0.5 * \frac{ab\pi}{2a * (2b - \delta)}$$

Equation 2-10 Area Void Density for Orthogonal Stack and Flattening

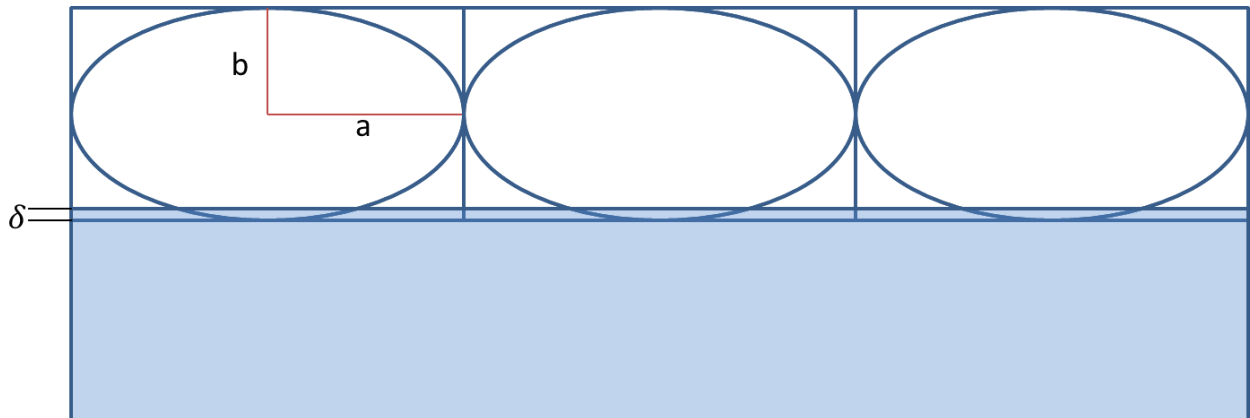


Figure 2-58 Orthogonal and flattening

The linear void ratio is described by Equation 2-9.

### 2.5.2 Stress-Strain Mathematics

The basic relationship relating stress and strain in orthotropic materials is given by

$$\begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{pmatrix} = [\bar{Q}] \begin{pmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{pmatrix}$$

Equation 2-11 Stress Strain relation for plane stress parallel to x-y (Sun, 2006)

where,  $\sigma_{xx}, \sigma_{yy}, \sigma_{zz}$  represent the components of the stress tensor,  $[\bar{Q}]$  is the shear extension coupling, and  $\varepsilon$  is the strain. The shear extension coupling comes from the transformation matrices for stress and strain along with the reduced stiffness matrix, all of which can be seen in 6.4. The relation between them is (Sun, 2006)

$$[\bar{Q}] = [T_\sigma]^{-1}[Q][T_\varepsilon]$$

Equation 2-12 Shear Extension Coupling (Sun, 2006)

where,  $[T_\sigma]$  is the transformation matrix for stress,  $[Q]$  is the reduced stiffness, and  $[T_\varepsilon]$  is the transformation matrix for strain. When strain gauges are arranged along  $0^\circ$ ,  $45^\circ$ , and  $90^\circ$  that data provides the strain vector allowing us to calculate the stress experienced in the x-y plane using these transformative equations. However, in practice with our raster orientations, other steps must be taken to analyze the stress over the differing raster angles as the load is not parallel to all of the rasters. All of our test articles were designed as symmetrical laminates meaning that at the mid-plane, the raster orientations are a mirror image of one another. Therefore, Equation 2-12 must be used for each different raster angle so that it can calculate the stress in each raster direction layer so we alter Equation 2-11 to reflect this minor change (seen in Equation 2-13).

$$\begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{pmatrix}_{n^\circ} = [\bar{Q}]_{n^\circ}(\varepsilon)$$

Equation 2-13 Stress Dependent on Raster Angle (Sun, 2006)

The x-y plane describes the stresses and strains along the plane formed by the loading but it does not describe the stress or strain in the raster directions formed by the rotated plane 1-2.

In order to calculate this simply multiply the stress or strain vector by its respective transformation matrix to determine these values.

One of the objectives for this project is to compare the composite laminate theory with our actual results. In the above method, data from strain gauges is used to determine the stress experienced by the test article. Alternatively, the strain can be predicted by the load experienced. To relate the load with the strain, the extensional stiffness [A] needs to be calculated using

$$A_{ij} = \sum_{k=1}^n \bar{Q}_{ij}^k t_k$$

Equation 2-14- Extensional Stiffness

where,  $t_k$  is the thickness of the raster or the slice height and  $\bar{Q}_{ij}^k$  represents the shear extension coupling value for each raster orientation at the same matrix coordinate. For our purposes, the slice height is the same for every layer thus simplifying the calculation of the extensional stiffness. Then by inverting [A] we can calculate the strain vector as shown in Equation 2-15.

$$\begin{pmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{pmatrix} = [A]^{-1} \begin{pmatrix} N_x \\ N_y \\ N_z \end{pmatrix}$$

Equation 2-15 Calculating Strain from Load

For our purposes,  $N_y$  and  $N_z$  are zero as we are only loading in the x-direction. Then, by using the strain vector in Equation 2-13 the stresses can be calculated and easily transformed to the raster coordinate directions 1-2.

### *Calculating Effective Material Properties using Laminate Theory*

To compare material property values with those determined graphically from tensile tests the following relations presented by Sun (2006) are used:

$$E_x = \frac{1}{hA'_{11}} \quad E_y = \frac{1}{hA'_{22}} \quad \nu_{xy} = -\frac{A'_{12}}{A'_{11}} \quad \nu_{yx} = -\frac{A'_{12}}{A'_{22}} \quad G_{xy} = \frac{1}{hA'_{66}}$$

Equation 2-16 Material Property Relations from Laminate Theory (Sun, 2006)

In the above,  $E_x$  is the Young's modulus in the load direction,  $E_y$  is the Young's Modulus in the orthogonal direction to the load,  $\nu_{xy}$  is Poisson's Ratio,  $\nu_{yx}$  is Poisson's Ratio,  $G_{xy}$  is the Shear Modulus, and  $A'$  is the inverted extensional stiffness matrix. These are the basic material properties that can be determined from our tests for the various printing configurations.

### 2.5.3 ANSYS Modeling of Tensile Tests

ANSYS is a software package based on Finite Element Analysis (FEA). It allows prediction of the deformation, strain, and stress of the dogbone specimens with appropriately defined supports and loads. The Orthotropic nature of FDM parts, which has been discussed in this paper, causes analysis to be difficult and time consuming using pencil and paper. Hambali et al qualified the use of FEA using orthotropic properties to generalize the material properties of FDM parts. Material property data supplied by Stratasys for ABS-M30 was insufficient for analysis, and after further inquiry, our group was rejected due to intellectual property rights. The data presented in Table 2-9 is a combination of data given by Hambali et al. after completing their own testing with FDM parts printed with ABS using a Stratasys Dimension SST and analytical estimate using equations provided through an ANSYS help forum<sup>3</sup>.

Table 2-9 Material Property Data

Material Property	Value	Source
Young's Modulus, $E_{xx}$	1904.000 MPa	Hambali et al.
Young's Modulus, $E_{yy}$	2228.000 MPa	Hambali et al.
Young's Modulus, $E_{zz}$	1822.500 MPa	Hambali et al.
Poisson's Ratio, $\nu_{xy}$	0.157	Hambali et al.

<sup>3</sup> These equations came from a post on XANSYS and were used as a first estimate because of a lack of data. It is understood by the authors that there is no precise way to calculate Shear Modulus from Young's Modulus and Poisson's Ratio as they are independent from one another in Orthotropic Materials.

Poisson's Ratio, $\nu_{yz}$	0.321	Hambali et al.
<b>Material Property</b>	<b>Value</b>	<b>Source</b>
Poisson's Ratio, $\nu_{xz}$	0.127	Hambali et al.
Shear Modulus, $G_{xy}$	896.9 MPa	Calculated (Equation 2-17)
Shear Modulus, $G_{yz}$	740.9 MPa	Calculated (Equation 2-18)
Shear Modulus, $G_{xz}$	824.2 MPa	Calculated (Equation 2-19)

$$G_{xy} = \frac{E_{xx} * E_{yy}}{E_{xx} + E_{yy} + 2 * \nu_{xy} * E_{xx}}$$

Equation 2-17 Shear Modulus  $G_{xy}$  (Patterson & Deibler, 2012)

$$G_{yz} = \frac{E_{yy} * E_{zz}}{E_{yy} + E_{zz} + 2 * \nu_{yz} * E_{yy}}$$

Equation 2-18 Shear Modulus  $G_{yz}$  (Patterson & Deibler, 2012)

$$G_{xz} = \frac{E_{xx} * E_{zz}}{E_{xx} + E_{zz} + 2 * \nu_{xz} * E_{xx}}$$

Equation 2-19 Shear Modulus  $G_{xz}$  (Patterson and Deibler, 2012)

We divided our analysis into three distinct categories in order to see the difference between simulation set-ups and data acquired through tensile testing: Orthotropic, Orthotropic with laminate analysis, and Orthotropic with laminate analysis and experimentally measured densities. In the first two analyses the density is set to  $1040 \text{ kg/m}^3$  (ABS-M30 Spec Sheet), whereas in the last analysis we take the volume data from the Solidworks files and our mass measurements from a scale accurate to two decimal places to calculate the densities. The data is presented in Table 2-10

Table 2-10 Experimental Masses and Densities

Test Article	Mass (g) (Measured)	Volume ( $\text{mm}^3$ ) (SW)	Density ( $\text{kg/m}^3$ )
45/-45 Y-axis	8.82	8666.56	1017.7
45/-45 Large Bead	8.84	8666.56	1020.0

45/-45 Small Bead	8.85	8666.56	1021.2
45/-45 n=1	1.38	1368.40	1008.5
45/-45 n=6	6.20	5929.75	1045.6
Test Article	Mass (g) (Measured)	Volume (mm <sup>3</sup> ) (SW)	Density (kg/m <sup>3</sup> )
0/90 Y-axis	8.74	8666.56	1008.5
Quasi Y-axis	7.22	7403.67	975.2
Quasi Large Bead	7.55	7403.67	1019.8
Quasi Small Bead	7.20	7403.67	972.5
Quasi n=1	4.05	3701.83	1094.1
Quasi n=6	21.80	22208.40	981.6

Each simulation was conducted through ANSYS Workbench using the Static Structural tool. The test articles were recreated in ANSYS using the Geometric modeling tool and dimensions from SolidWorks. The dogbones were drawn as wireframes and then given a surface of  $z = 0\text{mm}$  in order to set-up future laminate analysis. The analysis conditions were set using a fixed support for the bottom three edges (shown in green in Figure 2-59) and the Force was applied to the top three edges (shown in red in Figure 2-59) with its direction in tension to mimic the Instron set-up. The articles were either defined by a thickness (for the purely orthotropic analysis) or defined by layers (for the Laminate analyses). The layer thickness tool was used and each layer was given a thickness of  $0.1778\text{mm}$  to reflect the slicing thickness used by the Fortus 400mc and a raster angle dependent upon the article. Each layer in a given test article was assigned the same bulk material properties definition. For the mesh on each test, we decided to map each face as triangle best split to refine the default quadrilateral meshing and provide a better definition on the radiused corners. For the results, we obtained data on the total deformation, Principal Strain and Stress, and Maximum Shear Strain and Shear Stress.



Figure 2-59 Fixture and Force Set-up for ANSYS

In order to validate the model, we used data from the 45/-45 Y-axis oriented dogbone to provide material property data. This article was chosen because it exhibited smooth data and fracture directly underneath the strain gauge rosette. Therefore the in the loading direction before fracture is measured directly by the gauges. The article also did not have a strain relaxation stage within the stress-strain curve thereby making it easy to extract the data.

The first analysis conducted uses a combination of tensile data from the experiment and data from Habali et al. to investigate the material as purely Orthotropic. A force of 1252 N was applied to the dogbone specimen to correspond with the maximum loading during the tensile test before material failure. We expect with this loading to find a Maximum Principal Strain of 0.011 in the location of the strain gauge as garnered from Figure 2-16. The result of this analysis for Maximum Principal Strain can be seen in Figure 2-60 with an annotation at the approximate location of the strain gauge rosette.



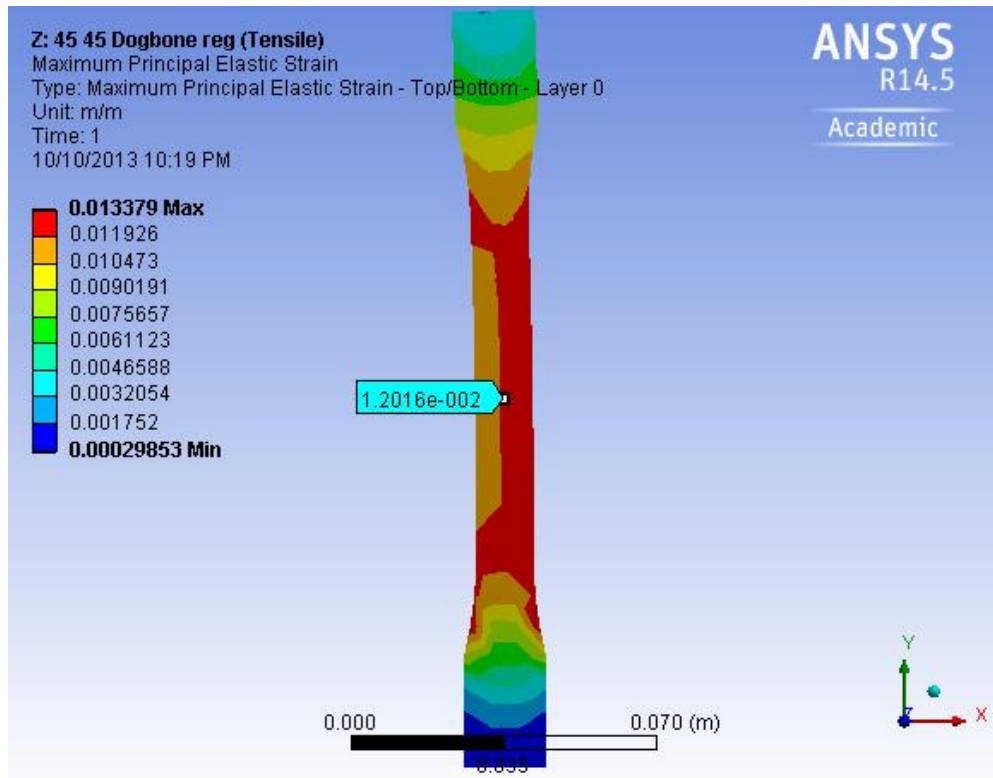


Figure 2-60 ANSYS Orthotropic Analysis for 45/-45 Dogbone using Tensile Data

This data appears to be unstable as there is an odd bend in the test article and the Principal Strain is uneven across the test section. Attempts to remedy this behavior were unsuccessful so we conclude that it may be because of the estimated Shear moduli that we observe this reaction from the applied force.

The next approach we used was modeling the dogbone as an orthotropic composite laminate using the layered tool from the Geometry set-up. This allowed used to input the raster angles and layer thicknesses (0.1778mm) for the 45/-45 dogbone. The Maximum Principal Strain results are show in Figure 2-61.

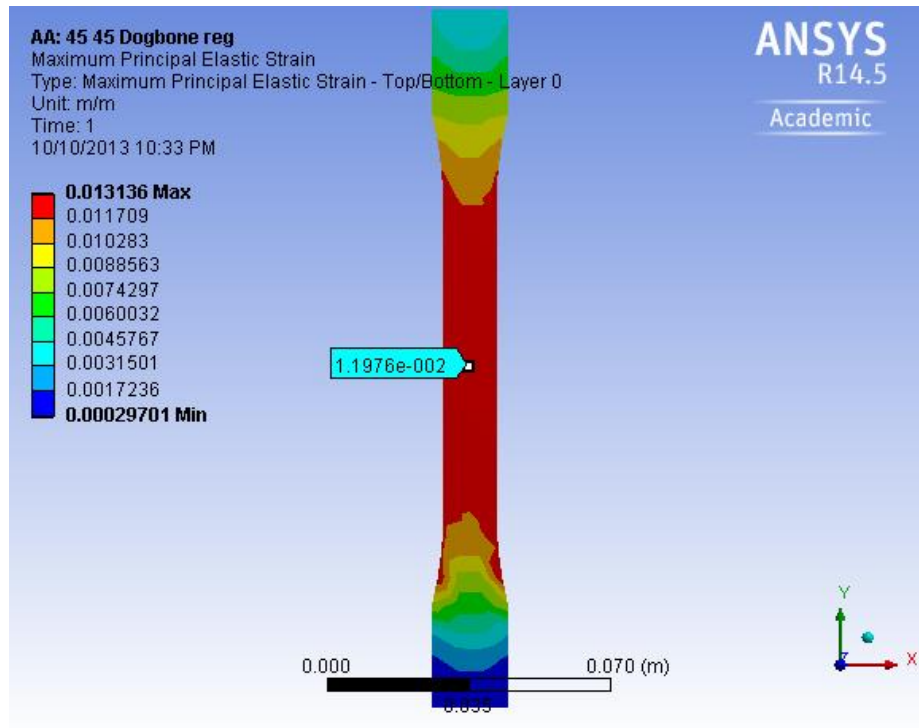


Figure 2-61 ANSYS Orthotropic Composite Laminate Analysis for 45/-45 Dogbone using Tensile Data

In this model we have resolved the bending issue seen within the strictly orthotropic analysis. The Strain data also appears to be consistent throughout the model with no noticeable discontinuities. The geometry of deformation follows that of the observed test article as there were no noticeable signs of necking or out of plane flexure. At the approximate strain gage location a Maximum Normal Strain value of 0.0112 is observed which is about an 8.87% difference from the data in Figure 2-16 thus we are in good agreement between ANSYS and the measured value.

Lastly we investigated the case where we altered the density to reflect the true density of the test article. In the other analyses we used the bulk density of  $1040 \text{ kg/m}^3$  for ABS-M30, but through FDM there will be gaps between layers and beads thus decreasing the density of the part. Using  $1017.7 \text{ kg/m}^3$  for our new density we once again analyzed the dogbone as an Orthotropic Symmetric Composite Laminate in ANSYS. The results of this analysis are show in Figure 2-62.

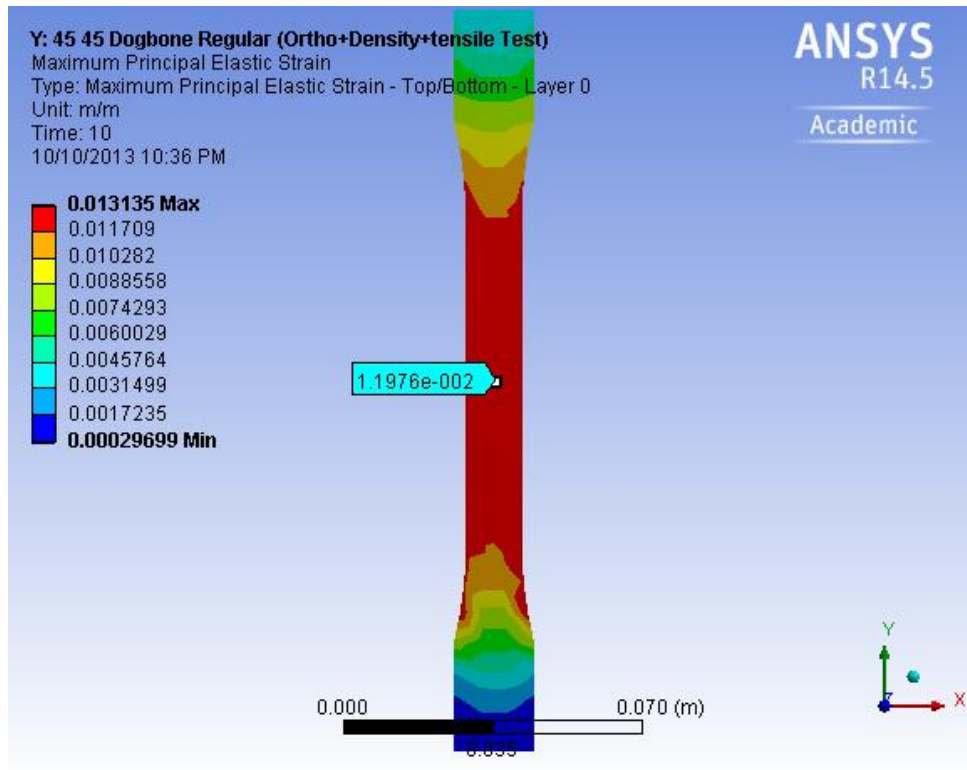


Figure 2-62 ANSYS Orthotropic Composite Laminate Analysis with True Density for 45/-45 Dogbone using Tensile Data

There is only a minute change in the scale for the Maximum Principal Strain and no noticeable difference in the strain distribution throughout the test article. The value at the rosette strain gauge location remains constant. This study would most likely prove more useful if investigating a sparse fill FDM part rather than the regular dense setting. For our purposes, remaining with the regular Orthotropic Symmetric Composite Analysis will attain good results when compared to the testing data.

### 3 Thermal Expansion Testing of FDM Manufactured Parts

This chapter focuses on the method of selection and use of strain gauges for the experiments carried out by the project team. The experimental set-up is explained for the thermal expansion test. The data from the three experiments is presented and analyzed. The chapter concludes with thermal modeling in COMSOL.

#### 3.1 Design of Thermal Expansion Test Articles

At the time of this project for FDM processed ABS or ABS-M30 there are no results in literature for thermal expansion. The existing set-up at MITLL allows for thermal testing across a wide temperature range and with varying heating/cooling rates. Raster orientation was tested independently as shown in Table 3-1. Since the oven at MITLL allows for multiple sensor outputs, it was possible to acquire data from each gauge concurrently. This provided a basis for cataloging the effects of environmental conditions on the test articles' resistance to deformation.

Table 3-1- Thermal Expansion Test Table

Test Article	45/-45	0/90	0/±45/90
1			
2			
3			

#### 3.2 Strain Gauge Decision for Thermal Expansion

In the thermal test, strain gauges were used to measure the elongation of the sample and find the thermal expansion coefficient. Three linear strain gauges were used in the thermal tests on the testing block and reference block as shown in Figure 3-1. The linear strain gauge was used on three sides of the thermal test cube and on the reference block. The 'G' Gauge was connected in series with its respective 'R' gauge<sup>4</sup>. This allowed for the collection of strain measurements in the various directions with respect to the plies to see if there was a difference

---

<sup>4</sup> Refer to 6.3 for a schematic of each Wheatstone Bridge set-up

in the thermal expansion coefficient. For example, data can be recorded parallel and perpendicular to the layers.

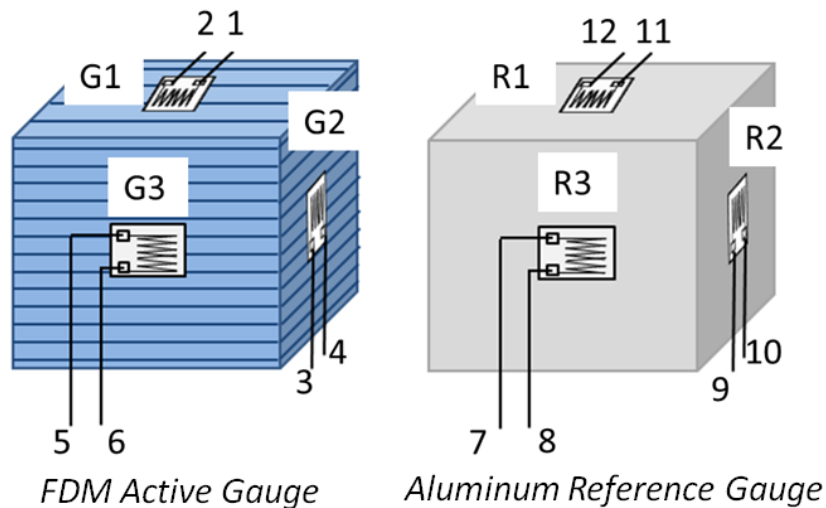


Figure 3-1 Strain gauges on FDM active block and reference block

The maximum strain level that the gauge can experience was also an important aspect due to the plastic applying a larger strain than a metal in the same test. The maximum strain was approximated for the thermal test using the bulk material thermal expansion coefficient and Equation 3-1 relating thermal expansion coefficient to strain:

$$\varepsilon = \alpha \Delta T$$

Equation 3-1 Bulk Strain and CTE

where  $\alpha$  is the thermal expansion coefficient of bulk ABS ( $73.8 * 10^{-6} m/m K$ ) and  $\Delta T$  is the change in temperature for the two extreme cases (85°C and -55°C evaluated in Kelvin). For both extreme temperatures, the estimated strain was 25,000  $\mu\varepsilon$ . The strain gauge used allows for repeated strains of approximately 30,000  $\mu\varepsilon$ , with larger strain for destructive testing.

The external temperature that the gauge is exposed to during testing is also an important factor to consider when designing the test apparatus. External temperatures applied to the gauge will cause the gauge grid to expand or contract and therefore change the resistance. In testing, it was ideal to avoid these types of resistance changes in order to only look at the applied

resistance from the test and not the external temperatures. To avoid this problem, a half Wheatstone bridge and a separate dummy gauge were used. The dummy gauge and the active gauge were installed in two arms of the bridge and both were exposed to the same temperature difference. The active gauge was installed on the test piece and the dummy gauge was not. Because both gauges experience the same temperature difference, the effects of temperature are cancelled out leaving only the strain from the test piece to be measured. In these tests, the dummy gauge was used in the thermal test where the greatest effect of temperature on the gauges was expected.

Based on all of the important factors to consider when using strain gauges, the SGD-3/350-LY41 linear gauge was chosen for the thermal testing from OMEGA Engineering. The gauge is 350Ω and rated for steel ( $V_{EX} \leq 9 V$ ) with a maximum applied strain of 30,000  $\mu\epsilon$ . Lead wires were soldered to the gauges and installed on the sample using SG401 ethyl-based cyanoacrylate adhesive supplied by OMEGA.

### 3.3 Thermal Expansion Test

Group 77 has a wide variety of resources for laboratory use including a thermal oven with a feed through port allowing for data acquisition during the heat cycle. The oven is programmable enabling precise control of the heating rates and temperatures as this is not a dedicated thermal expansion testing apparatus. Thermocouples and orthogonally placed strain gauges attached to the specimen will provide data that can be used to determine the linear thermal expansion coefficients.

The heat expansion test closely follows ASTM E831 for the testing standards. The furnace was programmed to raise the temperature by 10°C and maintain the temperature for 10 minutes in order to allow the thermal block to reach equilibrium with the ambient temperature. This ensures that we are taking data in sync with the temperature increase; likewise, the temperature was lowered by 10°C during the cooling cycle and maintained for the same amount of time. Data from the thermocouple was sufficient in accuracy and the strain gauge provided the data designated as the sensing element for the standard. In order to reduce the time needed for testing, the same test articles were used for both a cooling cycle and heating

cycle. In the both cycles, the material was brought from room temperature ( $\sim 25^{\circ}\text{C}$ ) to  $-55^{\circ}\text{C}$  and back to room temperature. After the first data was processed from this cycle, it was determined if there was plastic deformation from this test from a visual perspective and from the strain gauge reading<sup>5</sup>. If it was deemed that there was no residual strain or deformation from the cooling cycle, the same specimens were then be heated from room temperature to  $85^{\circ}\text{C}$  and reduced back to room temperature. This test method will save time in removing test articles and reconfiguring instrumentation. Stratasys advertises that ABS-M30 is thermally resistant so only fully-elastic deformation was expected in each test article.

In order to read the apparent thermal strain from the test samples, a combination of software and hardware from National Instruments was used. The gages were attached to a NI 9237 Bridge Module and the data was processed with LabView Signal Express software. The strain measurements were taken for approximately 10 seconds and the data averaged after finishing the measurements in Microsoft Excel. Figure 3-2 shows the thermal test set up within the chamber. The active and reference gage have three strain gauges installed to them. The thermocouple and two blocks are kept within the same proximity to each other within the chamber in order to ensure that they are exposed to the same temperature difference.



Figure 3-2: Image of thermal test set-up

---

<sup>5</sup> The strain gauge should register near 0 volts taking its accuracy into account and any noise in the data.

### 3.4 Thermal Expansion Test Results

Due to time limitations of this project, only one sample of each raster orientation was tested. The results from experimentation and modeling in Comsol are presented in this section.

#### 3.4.1 Experimental Results for Thermal Expansion Coefficient

The strain input into Signal Express is a combination of the strain on the ABS test sample as well as the aluminum reference sample due to the geometry of the half Wheatstone bridge. If two strain gauges are configured in a half Wheatstone bridge and exposed to strain with the same sign, then the strain from one gauge will be subtracted from the other in the final output strain. Therefore, the strain from the reference gauge needs to be added on to the final strain in order to isolate the strain on the ABS sample. To accomplish this, a theoretical thermal expansion equation is derived from the known thermal expansion coefficient of Aluminum 6061 (24.5  $\mu\text{m}/\text{mK}$ ) and one data point  $(T, \mu\epsilon) = (25, 0)$  following

$$y = 24.5x - 612.5$$

Equation 3-2 Expected Strain

In the above,  $y$  is the apparent thermal strain and  $x$  is the Temperature in  $^{\circ}\text{C}$ . This equation is used to interpolate the expected strain and add that the output strain measured by Signal Express in order to isolate the apparent thermal strain on the ABS sample. The results of this analysis are shown in Figure 3-3, Figure 3-4, Figure 3-5, Figure 3-6, Figure 3-7, Figure 3-8, and Figure 3-9.

#### *[45/-45] Block*

The experimental results for the [45/-45] block are shown in Figure 3-3 and Figure 3-4.



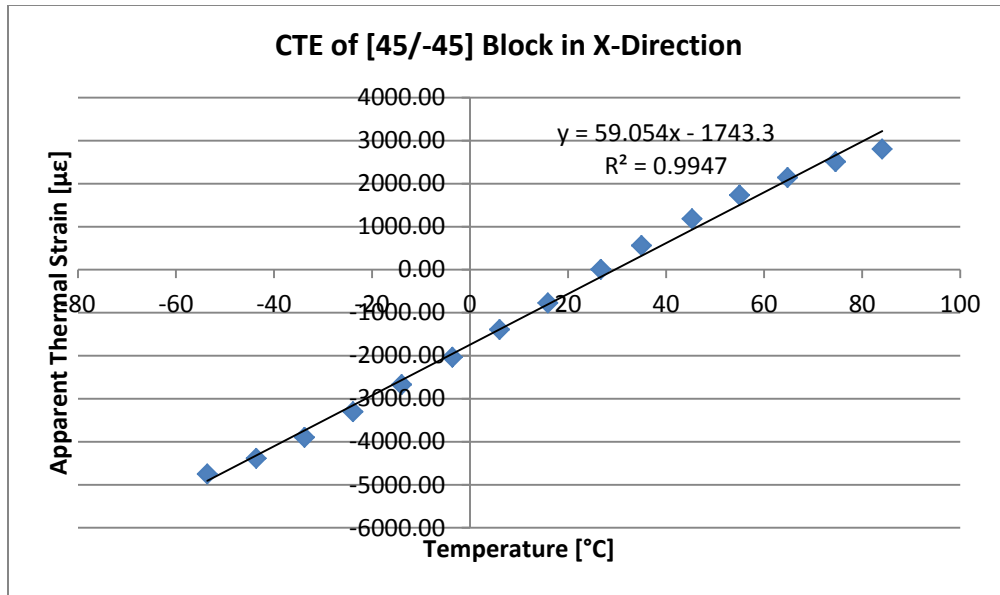


Figure 3-3: CTE of [45/-45] block in X direction

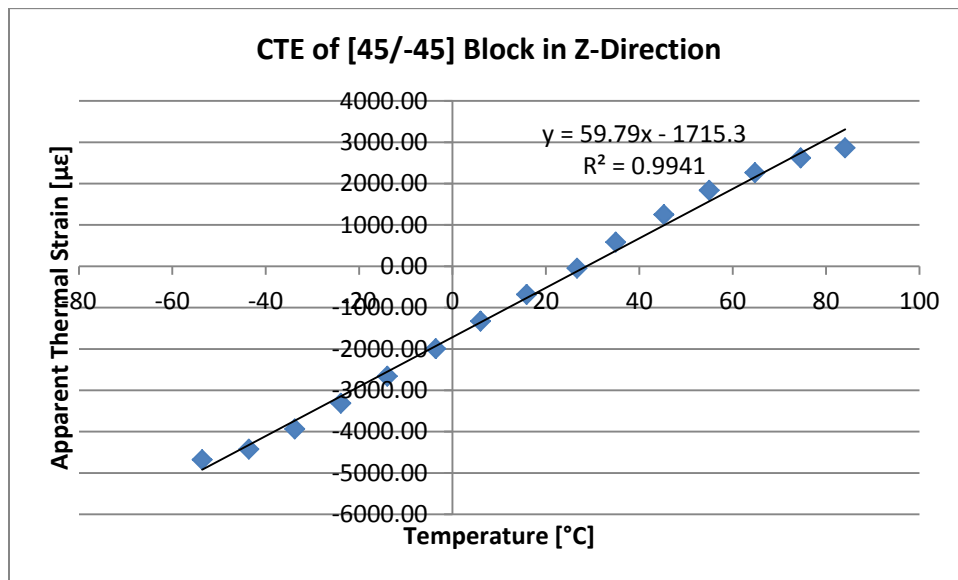


Figure 3-4: CTE of [45/-45] block in the Z direction

The results of this test show a CTE value in the X direction of  $59.1 \frac{\mu m}{mK}$  and in the Z direction of  $59.5 \frac{\mu m}{mK}$ . As expected, it was shown in the experimental results that the CTE in the Z direction between the layers was larger than the CTE in the rasters. The average CTE in both directions is  $59.5 \frac{\mu m}{mK}$ , approximately 19% lower than the coefficient of thermal expansion for bulk ABS.

[0/90] Block

The coefficient of thermal expansion test results for the [0/90] sample is shown in Figure 3-5: CTE of [0/90] block in the XY direction and Figure 3-6.

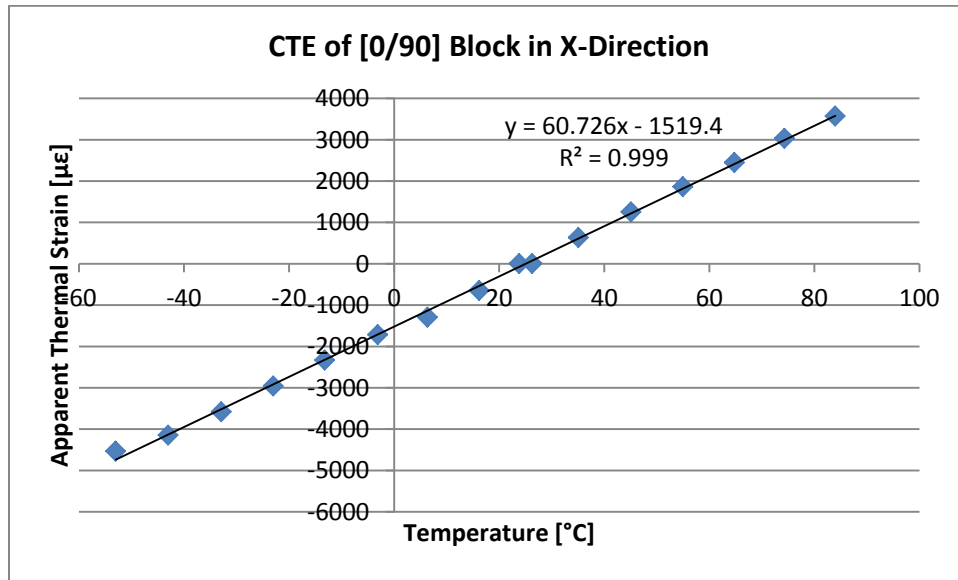


Figure 3-5: CTE of [0/90] block in the XY direction

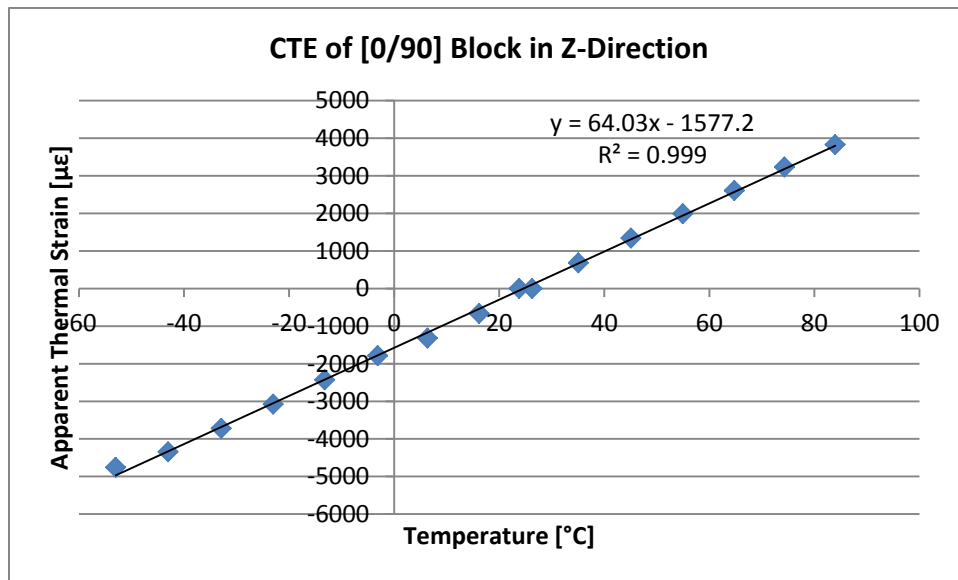


Figure 3-6: CTE of [0/90] block in the Z direction

The results of this test show a CTE value in the X direction of  $60.7 \frac{\mu m}{mK}$  and in the Z direction of  $64.0 \frac{\mu m}{mK}$ . As expected, it was shown in the experimental results that the CTE in the Z direction

between the layers was larger than the CTE in the rasters. The average CTE in both directions is  $62.4 \frac{\mu m}{mK}$ , approximately 15% lower than the coefficient of thermal expansion for bulk ABS.

*[0/90/±45]<sub>2</sub> Quasi Isotropic Block*

The coefficient of thermal expansion test results for the quasi-isotropic sample is shown in Figure 3-7, Figure 3-8, and Figure 3-9.

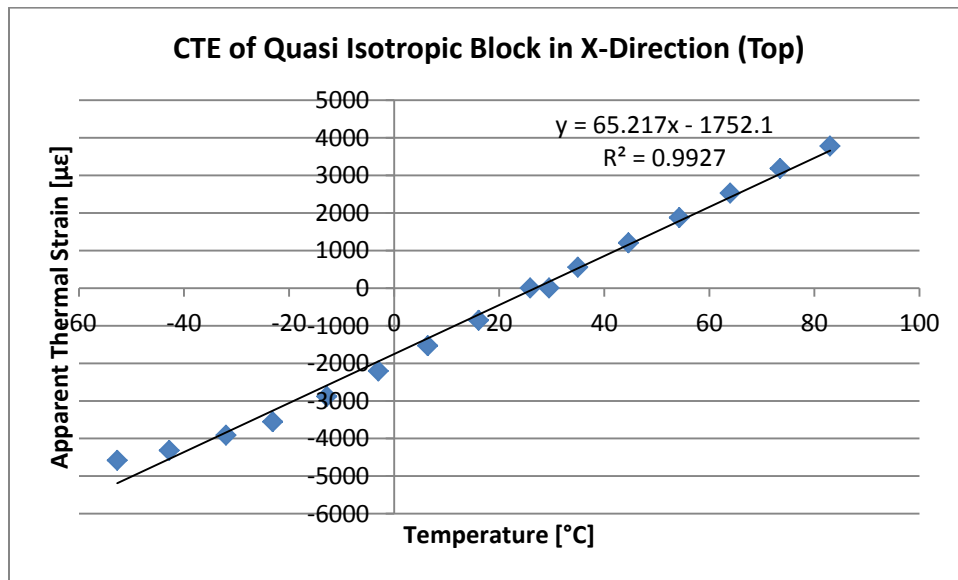


Figure 3-7: CTE of quasi-isotropic block in the X direction (Gauge 1)

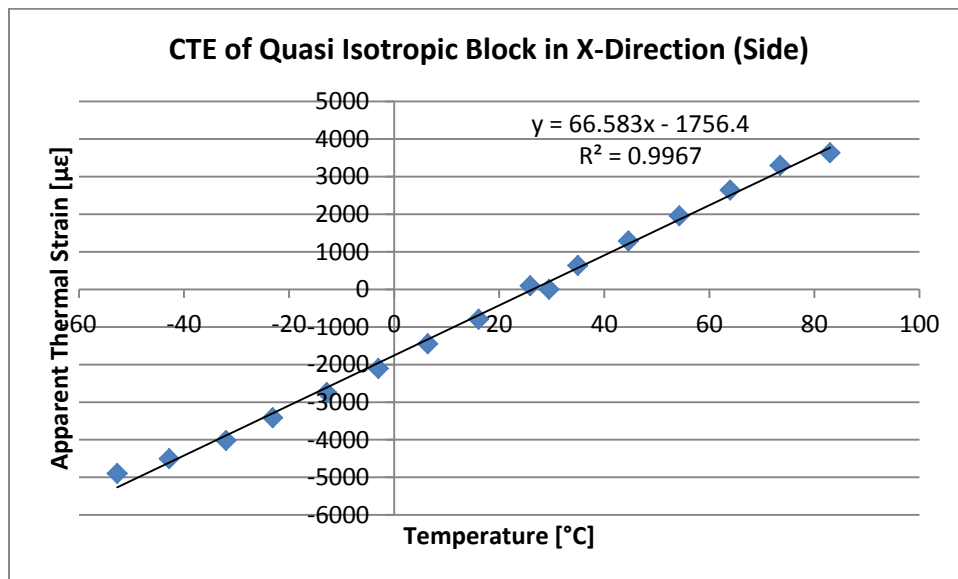


Figure 3-8: CTE of quasi-isotropic block in the X direction (Gauge 3)

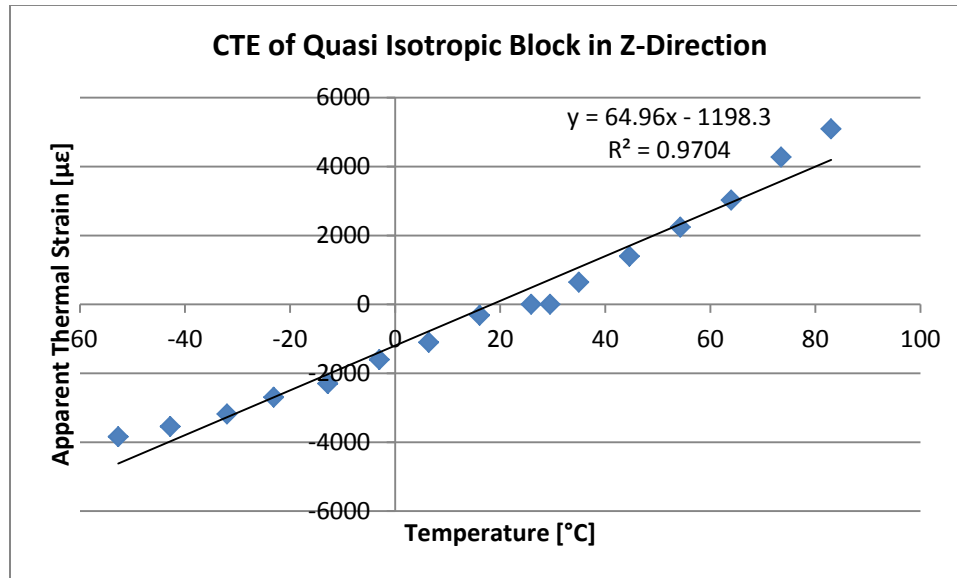


Figure 3-9: CTE of quasi-isotropic block in the Z direction

The results of this test show a CTE value in the X direction from gauge 1 of  $65.2 \frac{\mu m}{mK}$  and from gauge 3 of  $66.6 \frac{\mu m}{mK}$  with an average value of  $65.9 \frac{\mu m}{mK}$ . The CTE value in the Z direction was calculated as  $64.9 \frac{\mu m}{mK}$ . In this test, the average CTE value in the Z direction was not larger than in the X direction, as seen in previous tests. This may be due to the Z direction gauge recording data that did not show as linear as the other two gauges. The trendline for the Z direction gauge has a  $R^2$  value of 0.9704 instead of 0.993 or 0.997 as shown in the X direction gauges. This test must be conducted again in order to verify these results. Also, the average CTE in both directions is  $65.6 \frac{\mu m}{mK}$ , approximately 11% lower than the coefficient of thermal expansion for bulk ABS. The quasi-isotropic showed a CTE closest to the ABS bulk material properties out of all the samples, even taking into consideration that the data from the Z direction was not as accurate.

### 3.4.2 COMSOL Modeling of Thermal Expansion Tests

COMSOL is a multiphysics finite element analysis program that allows analysis of the thermal expansion of samples heated and cooled in a thermal chamber as well as the heat transfer behavior through the samples (Product Suite, 2013). COMSOL uses a thermal stress module to study the internal stresses and the temperature gradient within the block. Using COMSOL, the

samples were modeled first using the isentropic bulk material properties and then compared to the orthotropic samples from our experiments to analyze how the thermal properties change.

The thermal chamber uses a 500W heater and a vertical-down airflow system to maintain temperature uniformity, as shown in Figure 3-10 (Thermal Product Solutions, 2010). Due to our samples being such a small size relative to the interior of the thermal chamber, we simplified the model by assuming the hot air at ideal temperature will blow from one side of the block to the other side in a straight line, heating only one side. This assumption allows for simplification of the geometry used in COMSOL as shown in Figure 3-11.

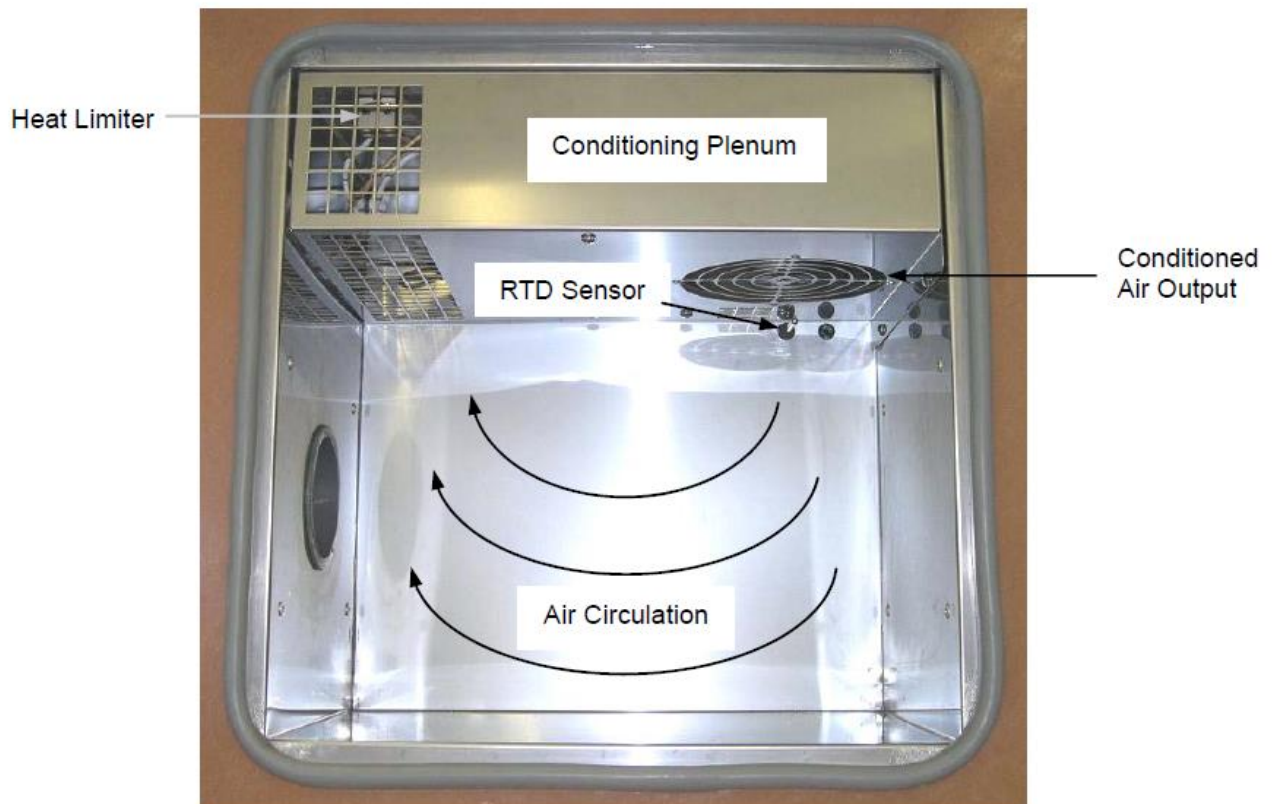


Figure 3-10 - Air circulation diagram in thermal oven. Picture provided by Thermal Product Solutions.

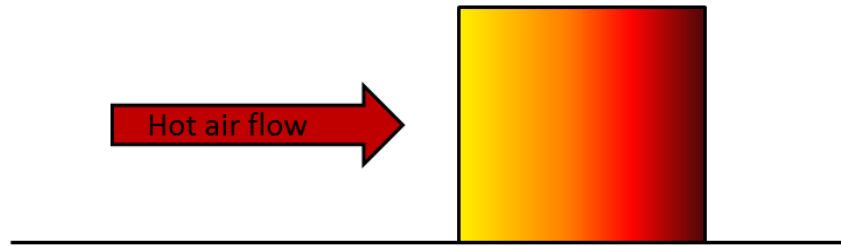


Figure 3-11 - Simplified model geometry for Comsol analysis

The COMSOL model analyzes the effects of conduction through the thermal block and neglects forced or free convection and any radiation present. In order to compare the bulk material properties to the properties of the printed parts, the density and the thermal expansion coefficients for the x, y, and z directions were changed for the three thermal samples. Each sample was modeled in the heat transfer and solid mechanics modules available in COMSOL. The constant material properties were manually input using the following parameters in Table 3-2.

Table 3-2 Parameters for Comsol

Description	Value	Parameter Name
Block height	10 [mm]	H
Block length	10 [mm]	L
Specific heat	1400 [J/(kg/K)]	Cp
Young's modulus	1.8 [GPa]	E
Poisson's ratio	0.33	$\nu$
Thermal conductivity	0.17 [W/(m*K)]	k

The block is heated on one surface and insulated on the other 5 surfaces allowing for heat conduction through the block. The block was heated at a constant temperature until the temperature gradient had a difference of no more than 0.5°C and the block was not under any thermal stresses. One surface had to be fixed due to the requirements of the program. In order to avoid fixing one side of the cube, a 0.05mm diameter post was added to the bottom of the cube, and the surface where the block and post intersected was the fixed surface. This allowed

COMSOL to calculate the expansion in the X, Y, and Z directions with minimal interference from the fixed surface. The geometry of the block is shown in Figure 3-12.

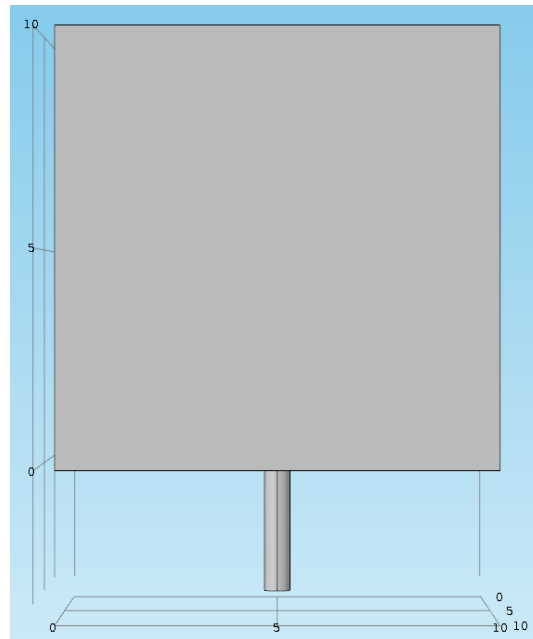


Figure 3-12 - Geometry of block modeled in Comsol Multiphysics

### *Heat Transfer Analysis*

The densities of the ABS-M30 samples were calculated and used to analyze the conductive heat transfer through the blocks. This test was executed using a two dimensional conductive heat transfer model and analyzing a block heated for one minute. Three models were developed for the blocks with different raster angle orientations and densities. After one minute, we choose an arbitrary temperature, 320K, and calculated the distance into the block that was heated to this temperature. A diagram of this process is shown in Figure 3-13 and the specific results from each test are shown in Figure 3-14.

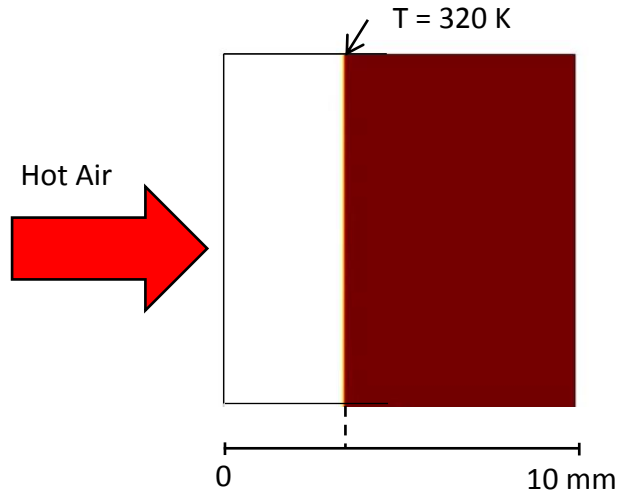


Figure 3-13 - Diagram of heat transfer analysis

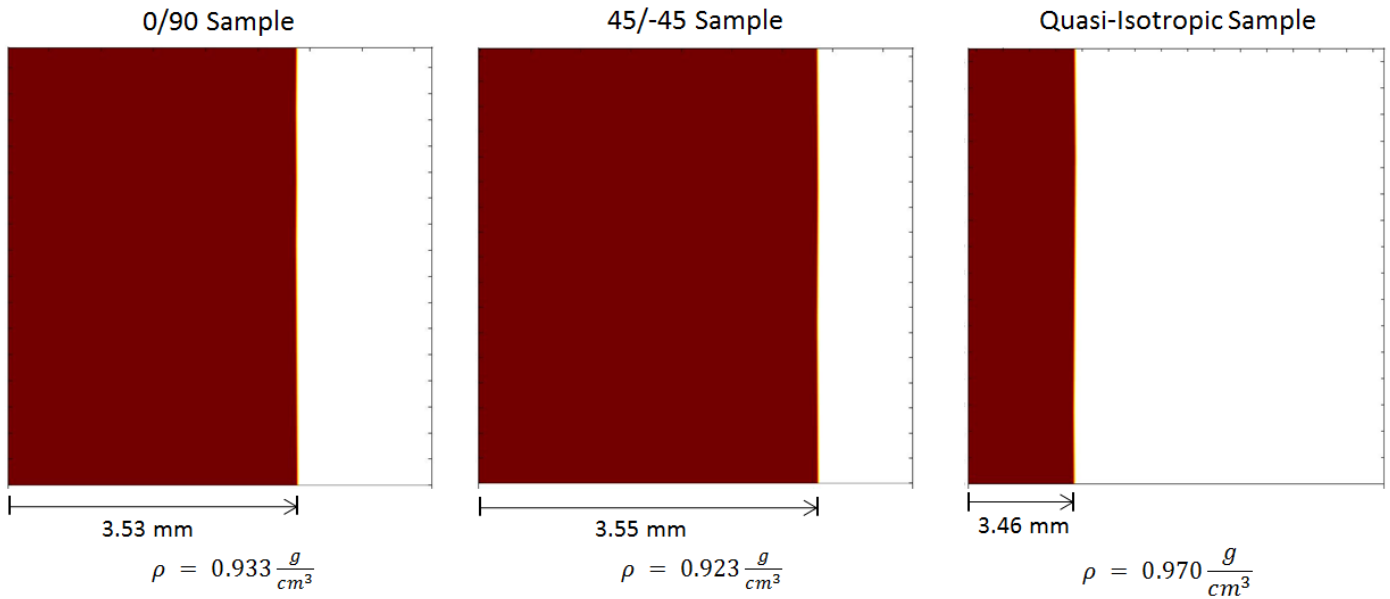


Figure 3-14 - Modeling results for conductive heat transfer through samples

Figure 3-15 shows the isothermal contours in the cube. It was found that the difference in density did not make a significant change conductive heat transfer based on the results found in COMSOL. However, we wanted to validate that the heat will travel faster through the less dense samples using COMSOL.



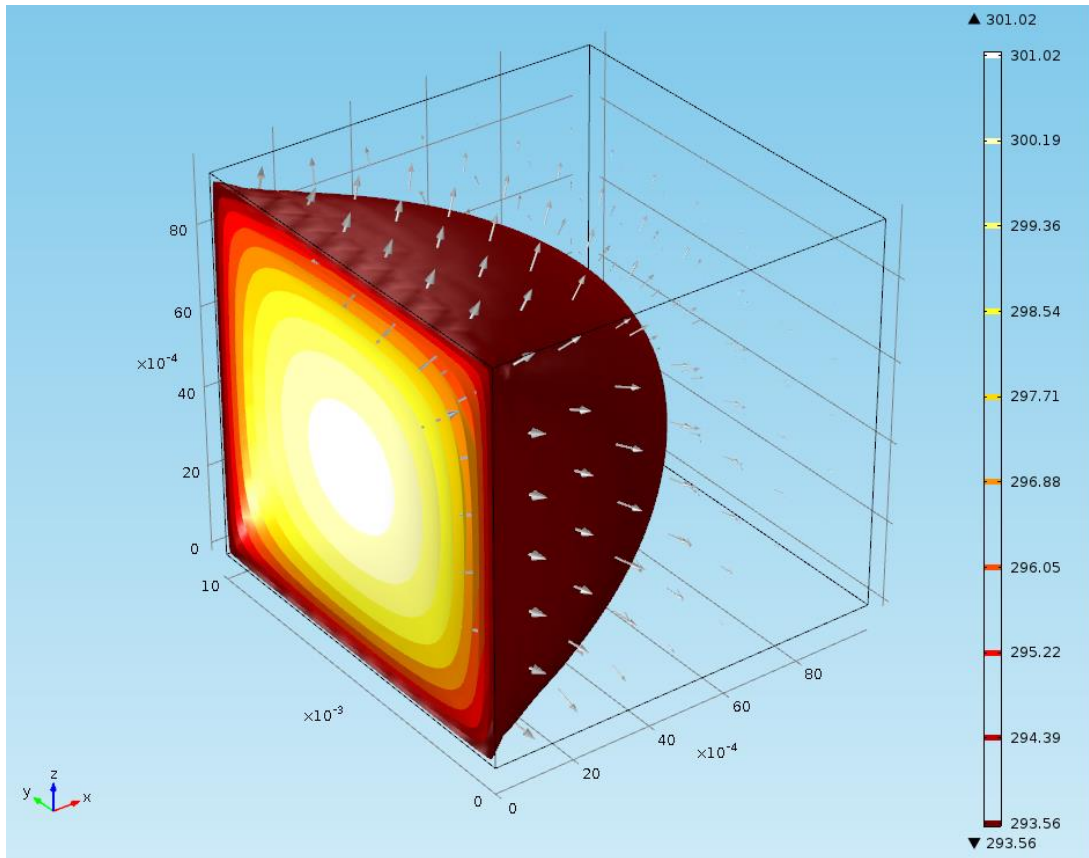


Figure 3-15 – Isothermal contours for heat transfer through sample

The results of this analysis validate that heat traveled the slowest through the quasi-isotropic block with the highest density and faster through the low density blocks. The heat transfer behavior based on density is a useful topic to know when designing end use parts with FDM.

### *Thermal Expansion Analysis*

The thermal expansion was analyzed using the Solid Mechanics module available in Comsol Multiphysics software. The specific densities and thermal expansion coefficients were input for each model based on the experimental results and calculations. The model was simplified to a two dimensional model based on the assumption that the X and Y thermal expansion coefficients will be approximately equal. On the two dimensional model, the total deformation in the X and Y directions was modeled and the results analyzed. Figure 3-16 and Figure 3-17 shows that the total deformation model looks like in the Comsol window. The deformation in

the Z direction is read directly off the chart as 0.0587 mm. The XY direction is calculated by adding the deformation in the  $-X$  and  $X$  direction to calculate a value of 0.0589 mm. The results for all the samples were compiled and are shown in Table 3-3.

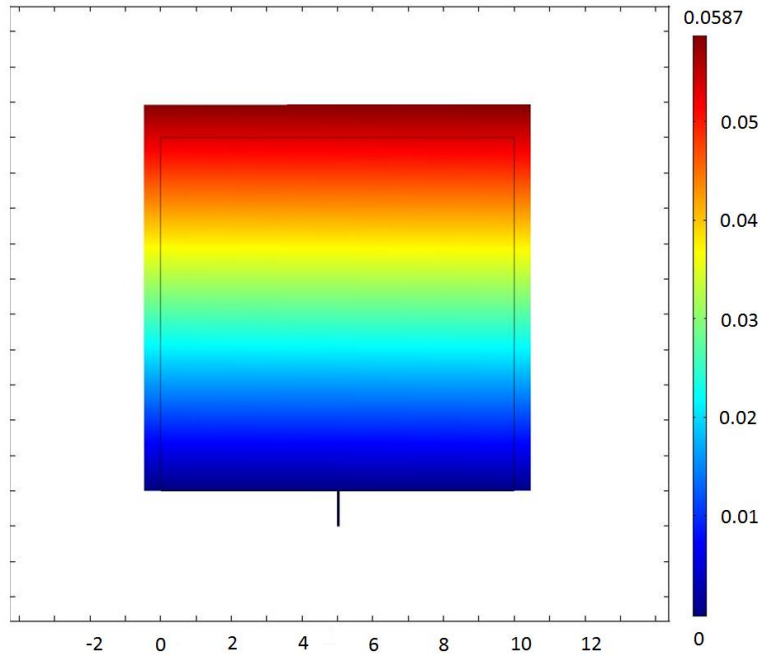


Figure 3-16 - Total deformation in the Y direction for an isotropic sample

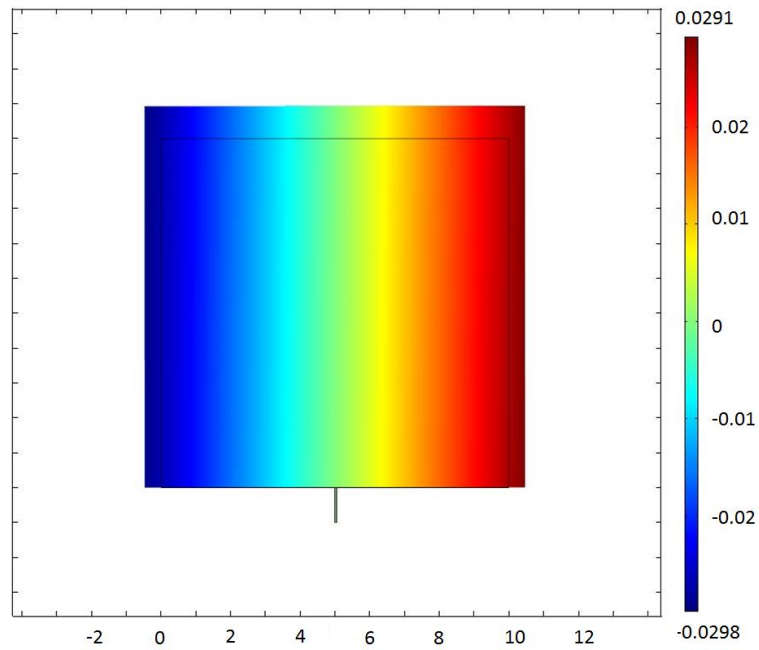


Figure 3-17 - The total deformation model in X direction for isotropic sample

Table 3-3 - Density, coefficient of thermal expansion, and total deformation of FDM printed samples

	Isotropic	45/-45	0/90	Quasi-isotropic
$\rho \left( \frac{g}{cm^3} \right)$	1.04	0.923	0.933	0.97
$\alpha_{xy} \left( \frac{\mu m}{mK} \right)$	74.3	59.1	60.7	65.9
$\alpha_z \left( \frac{\mu m}{mK} \right)$		59.8	64.0	65.0
$\Delta l_{xy} (mm)$	0.0589	0.0475	0.0487	0.0530
$\Delta l_z (mm)$	0.0587	0.0477	0.0507	0.0522

The expansion calculated from Comsol shows that of the FDM samples, the Quasi-Isotropic sample has the largest overall deformation. Results also show that the thermal expansion coefficient is also generally larger in the Z direction than in the XY direction.

## 4 Summary, Conclusions and Recommendations for Future Work

### 4.1 Tensile Testing and Modeling

Polymers processed through Fused Deposition Modeling exhibit interesting characteristics due to the build parameters. Our objective was to determine the effect build parameters have on the material properties for tensile loading. The tensile test articles did not exhibit any necking as would be expected from thermoplastics. This means that FDM parts do not behave like normal polymers as they do not exhibit large regions of plastic deformation before fracture; this is corroborated by the majority of articles failing due to brittle fracture. Unfortunately, brittle fracture means that parts printed with FDM will fracture at an unpredictable stress near the Ultimate Tensile Stress, and as a result, the designed stress must be much lower to avoid catastrophic failure.

Overall, the 45/-45 raster provides the largest Ultimate Tensile Strength out of the three raster orientations tested for this project. This also comes with some drawbacks as the ductility of the 45/-45 raster is the least out of the printing rasters. It will not hold up to larger strain states such as the Quasi Isotropic raster which has the highest ductility. By having a larger ductility, the transition strain is also more likely to be higher as is the case in the dogbones tested in this project. Though the Quasi Isotropic raster has a lower UTS, it performs more similarly to the bulk material than any of the other rasters.

Another way to increase ductility is by increasing the width of the bead. As we observed, there was a noticeable difference in ductility with the increased bead size without a sacrifice to the UTS (comparison graphs are included in Appendix 6.2 refer to Figure 6-12 and Figure 6-13). Further experiments would be needed to confirm these results, but preliminary data would suggest that flexibility of printed parts could be improved by using a larger bead width. By increasing the ductility, we once again approach the bulk material properties of ABS-M30 from that of the other brittle dogbones.

A desire for weight savings while maintaining good strength was a priority for Group 77 for their UAV project. We found that reasonable strength is maintained at a single ply for both the

45/-45 and Quasi Isotropic rasters. A potential problem with using the 45/-45 single ply is its highly brittle nature; validation of the strength of this build parameter acting with other structures would be necessary.

In order to model FDM processed ABS-M30, a symmetric composite laminate analysis is necessary for good agreement between physical observations and computational modeling for an article in tension. A purely orthotropic analysis will not provide accurate data for an article in tension.

## 4.2 Thermal Testing and Modeling

Group 77 also has an interest in investigating how the FDM parts perform in environmental conditions such as extreme temperatures. We experimented with the three raster orientations in a temperature range of -55°C to 85°C, a range often specified by government specifications, and compared the thermal expansion coefficients with the bulk material properties. The results of these tests showed that the FDM printed parts will have a slightly lower thermal expansion coefficient than the bulk ABS-M30. Bulk ABS-M30 has a coefficient of thermal expansion of  $73.4 \frac{\mu m}{mK}$  and we calculated the FDM parts to be approximately 15% lower at  $62.9 \frac{\mu m}{mK}$ . Therefore, if the expansion in an isotropic ABS-M30 part is acceptable for a part's requirement, then the FDM printed part will expand less and also perform acceptably. Similarly to the tensile tests, the quasi-isotropic raster orientation performed the closest to the bulk material properties.

In conclusion to this project, we have verified that the build parameters have an effect on the material properties of FDM printed parts and investigated the parameters relation to specific material properties. Group 77 may use this information to help guide their choices in build parameters when printing end-use parts and also as a basis for further testing and investigation.

## 4.3 Benefits to MITLL and Recommendations for Future Work

The data that this project provides to MITLL presents engineers with material property trends for several different build parameters. It defines the fracture trends for different rasters, plies, bead widths, and orientations in tensile testing. This data compares the FDM produced parts to

the bulk material properties where the findings suggest that additively manufactured parts are weaker than the material used in printing. Results also show the thermal expansion trends of various raster orientations are lower than the bulk material properties. It suggests that FDM-printed parts are more resistive to temperature fluctuations and will maintain their dimensions and geometry better than an extruded part.

The amount of tensile and thermal expansion experiments was limited due to budget and time constrictions for this project. We recommend a study that would test several articles of each build parameter to determine statistical significance for our findings. We also recommend a three-point bend test be done for the different build parameters to determine which materials would be ideal for the internal spars of the wings for the UAV project. Further testing should also compare the data to a machined ABS-M30 dogbone to determine the bulk properties in full including the transition strain and strain state at fracture.

Thermal expansion tests should also cover full-size parts with complex geometries as a flight test. This would help validate future UAVs for use in hot climates and high altitudes without risk of mission failure from internal thermal stresses on the structure. Different size bead widths should be integrated into the thermal tests to determine if this build parameter change will introduce unwanted thermal stresses. Larger beads were shown to be good for ductility but this could also mean that this material property change could signify a larger coefficient of thermal expansion.

## 5 References

- ABS-M30 Spec Sheet. (n.d.). *ABS-M30*. Retrieved August 26, 2013, from <http://www.stratasys.com/materials/fdm/abs-m30>
- Ahn, S. H., Montero, M., Odell, D., Roundy, S., & Wright, P. (2002). Anisotropic material properties of fused deposition modeling ABS. *Rapid Prototyping Journal*, 8(4), 248-257. doi:10.1108/13552540210441166
- Askeland, D. R., Fulay, P. P., & Wright, W. J. (2011). *Polymers. The science and engineering of materials* (6th ed., pp. 619-625). Stamford, CT: Cengage Learning.
- ASTM Standard D3039, 2008, "Standard Test Method for Tensile Properties of Polymer Matrix Composite Materials," ASTM International, West Conshohocken, PA, 2003, DOI:10.1520/D3039\_D3039M-08, [www.astm.org](http://www.astm.org)
- ASTM Standard D638, 2010, "Standard Test Method for Tensile Properties of Plastics," ASTM International, West Conshohocken, PA, 2003, DOI: 10.1520/D0638-10, [www.astm.org](http://www.astm.org)
- ASTM Standard E831, 2012, "Standard Test Method for Linear Thermal Expansion of Solid Materials by Thermomechanical Analysis," ASTM International, West Conshohocken, PA, 2003, DOI: 10.1520/E0831-12, [www.astm.org](http://www.astm.org)
- Fatimatuzahraa, A. W., Farahaina, B., & Yusoff, W. A. Y. (2011). The effect of employing different raster orientations on the mechanical properties and microstructure of fused deposition modeling parts. 22-27. doi:10.1109/ISBEIA.2011.6088811
- Fused Deposition Modeling (FDM). (n.d.). *CustomPart.Net*. Retrieved September 24, 2013, from <http://www.custompartnet.com/wu/fused-deposition-modeling>
- Hambali, Ruzy Haryati and Celik, Kursat and Smith, Paul and Rennie, Allan and Ucar, M (2010) *Effect of build orientation on FDM parts:a case study for validation of deformation behaviour by FEA*. In: International Conference on Design and Concurrent Engineering (iDECEN 2010). Penerbit Universiti (UTeM), pp. 224-228. ISBN 983-2948-84-1

Hoffmann, K. Applying the Wheatstone Bridge (White paper). Hottinger Baldwin Messtechnik GmbH, Darmstadt, Germany: Tips & Tricks for Experimental Stress Analysis.

Kalita, S. J., Bose, S., Hosick, H., & Bandyopadhyay, A. (2003). Development of controlled porosity polymer-ceramic composite scaffolds via fused deposition modeling. *Materials Science and Engineering*. 23(5) 611-620. DOI: 10.1016/S0928-4931(03)00052-3

Kouhi, E., Masood, S., Morsi, Y. (2008). Design and fabrication of reconstructive mandibular models using fused deposition modeling. *Assembly Automation*. 28(3), 246-254. DOI: 10.1108/01445150810889501

Li, L., Sun, Q., Bellehumeur, C., & Gu, P. (2002). Composite modeling and analysis for fabrication of FDM prototypes with locally controlled properties. *Journal of Manufacturing Processes*, 4(2), 129-141. doi:10.1016/S1526-6125(02)70139-4

Mechanical Properties. (n.d.). *IMPRESS*. Retrieved August 26, 2013, from [www.spaceflight.esa.int/impress/text/education/MechanicalProperties/](http://www.spaceflight.esa.int/impress/text/education/MechanicalProperties/)

Micro-Measurements (2010). *Precision Strain Gauges* [White Paper]. Retrieved October 15, 2013, from <http://www.vishaypg.com/doc?11201>.

NI 9944, NI 9945 . (n.d.). *National Instruments: Test, Measurement, and Embedded Systems*. Retrieved October 15, 2013, from <http://sine.ni.com/nips/cds/view/p/lang/en/nid/211299>

Patterson, J., & Deibler, J. (2012, April 16). Orthotropic Material. *XANSYS*. Retrieved September 26, 2013, from [www.xansys.org/forum/viewtopic.php?p=84501&sid=8e8f7e024187d300f50610cbfa897cce](http://www.xansys.org/forum/viewtopic.php?p=84501&sid=8e8f7e024187d300f50610cbfa897cce)

Product Suite (2013). *Comsol.com*. Retrieved October 17, 2013, from [http://www.comsol.com/products?utm\\_expid=1422053-2.VoUKHhUoRBKFVuvbN4hlBg.0&utm\\_referrer=http%3A%2F%2Fwww.comsol.com%2F](http://www.comsol.com/products?utm_expid=1422053-2.VoUKHhUoRBKFVuvbN4hlBg.0&utm_referrer=http%3A%2F%2Fwww.comsol.com%2F)



Scalea, F. Lanza di. (1998). Measurement of thermal expansion coefficients of composites using strain gages.

Strain Gage Rosettes - Selection, Application and Data Reduction. (2010, September 12). *Vishay Precision Group*. Retrieved September 16, 2013, from <http://www.vishaypg.com/doc?11065>

Strain Gauge Measurement- A Tutorial. (n.d.). *ME 3901 Engineering Experimentation*. Retrieved September 13, 2013, from <https://users.wpi.edu/~sullivan/ME3901/index.html>

Sun, C. T. (2006). Analysis of Composite Laminates. *Mechanics of aircraft structures* (2nd ed., pp. 271-292). Hoboken, N.J.: J. Wiley.

*Tip Selection* (White Paper) Stratasys Incorporated, Eden Prairie, MN

## 6 Appendix

Included in the Appendix are Data, Engineering Specs, and Analysis Programs used within this Major Qualifying Project. This is provided to guide the reader through the entire technical process of this project.

### 6.1 Appendix A- Design and Manufacturing

Below is the Control Center Layout for the Tensile and Heat Expansion Specimens.

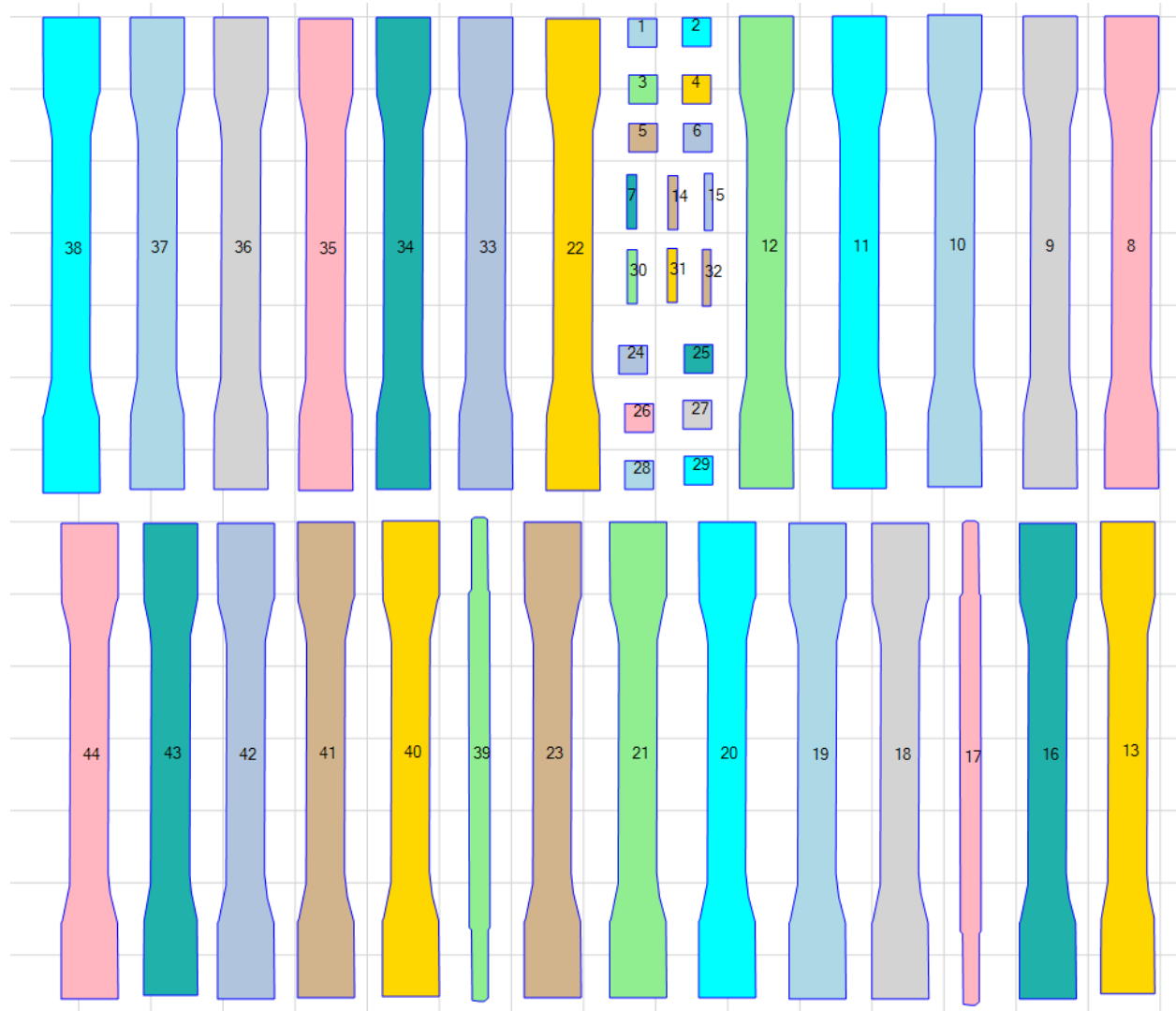


Figure 6-1- Printing Arrangement

Article	Description
1	Thermal 45/-45
2	Thermal 45/-45
3	Thermal 0/90
4	Thermal 0/90
5	Thermal Quasi
6	Thermal Quasi
7	Dogbone 45/-45 Z-axis
8	Dogbone 45/-45 Y-axis
9	Dogbone 45/45 Thin n=1
10	Dogbone 45/-45 Thick n=6
11	Dogbone 45/-45 Small Bead
12	Dogbone 45/-45 Large Bead
13	Dogbone 0/90 Y-axis
14	Dogbone 0/90 Z-axis
15	Dogbone Quasi Z-axis
16	Dogbone Quasi Y-axis
17	Dogbone Quasi X-axis
18	Dogbone Quasi Thin n=1
19	Dogbone Quasi Thick n=6
20	Dogbone Quasi Small Bead
21	Dogbone Quasi Large Bead
22	Dogbone 45/45 Thin n=1
23	Dogbone Quasi Thin n=1
24	Thermal 45/-45
25	Thermal 45/-45
26	Thermal 0/90
27	Thermal 0/90
28	Thermal Quasi
29	Thermal Quasi
30	Dogbone 45/-45 Z-axis
31	Dogbone 0/90 Z-axis
32	Dogbone Quasi Z-axis
33	Dogbone 45/-45 Y-axis
34	Dogbone 45/-45 Thick n=6
35	Dogbone 45/-45 Small Bead
36	Dogbone 45/-45 Large Bead
37	Dogbone 0/90 Y-axis
38	Dogbone Quasi Y-axis
39	Dogbone Quasi X-axis
40	Dogbone Quasi Thick n=6
41	Dogbone Quasi Small Bead

42	Dogbone Quasi Large Bead
43	Dogbone 45/45 Thin n=1
44	Dogbone Quasi Thin n=1

Table 6-1- Printing Arrangement Key

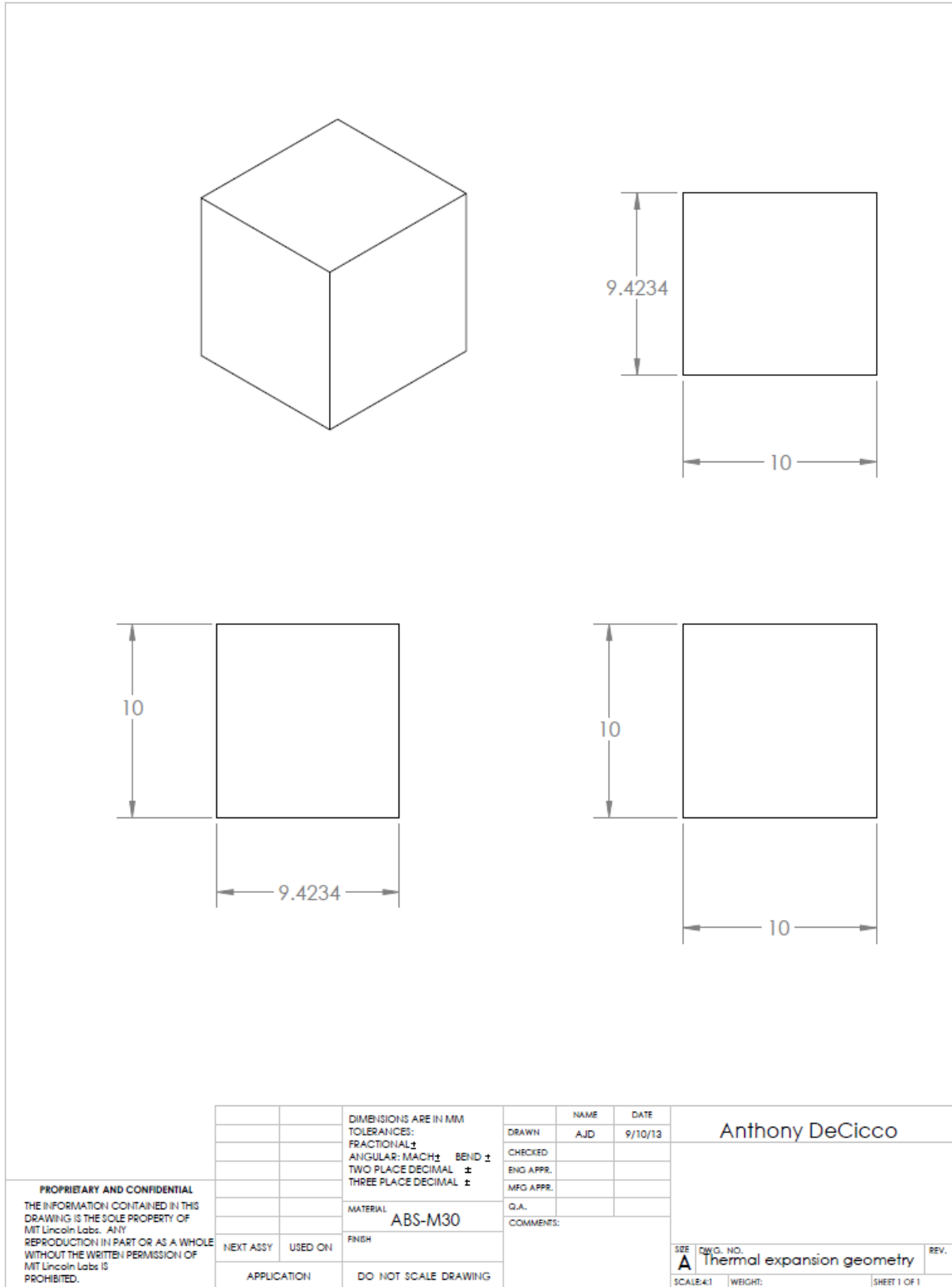


Figure 6-2-Thermal Geometry for 0/90 and 45/-45

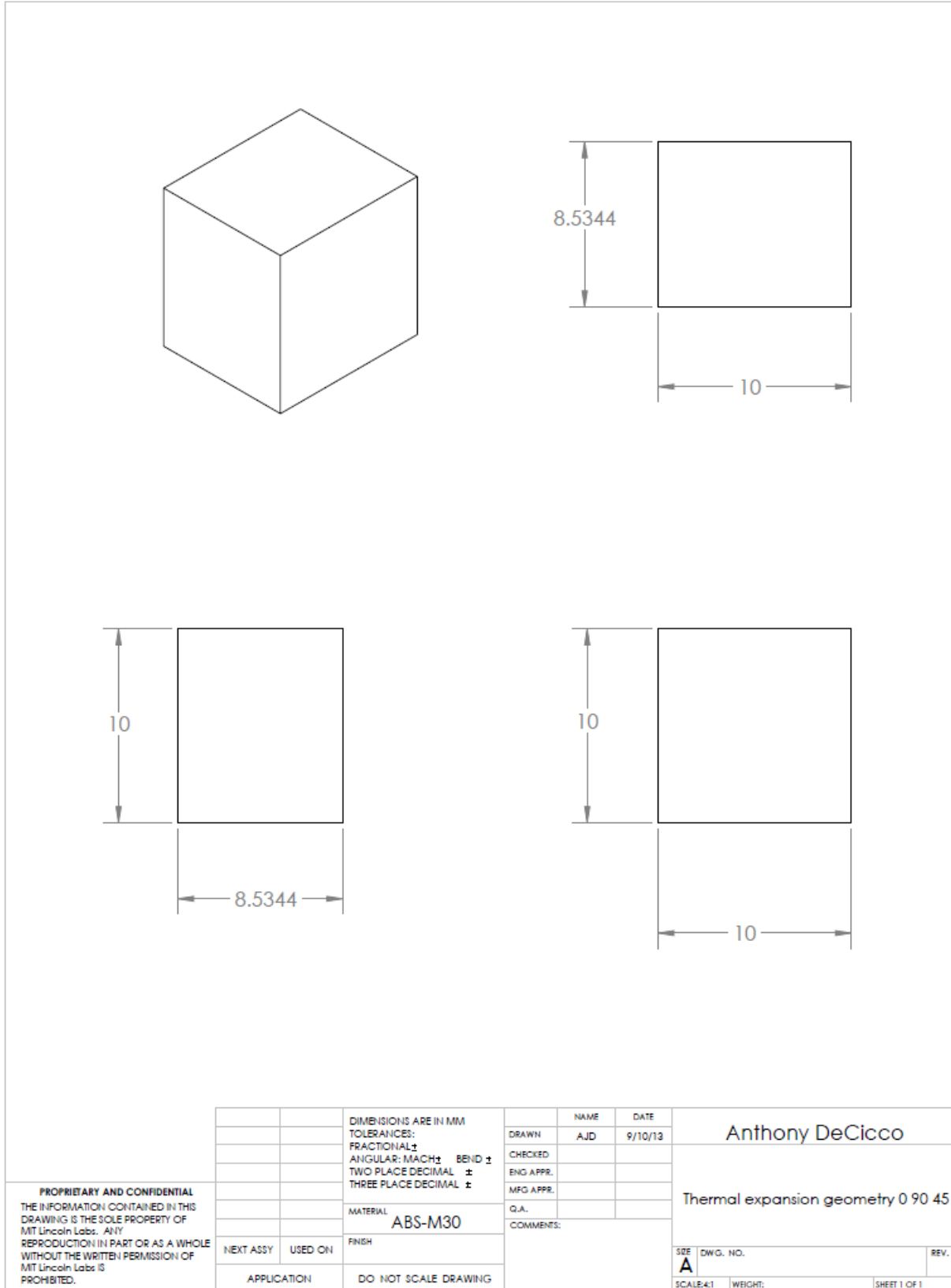


Figure 6-3- Thermal Geometry for Quasi Isotropic

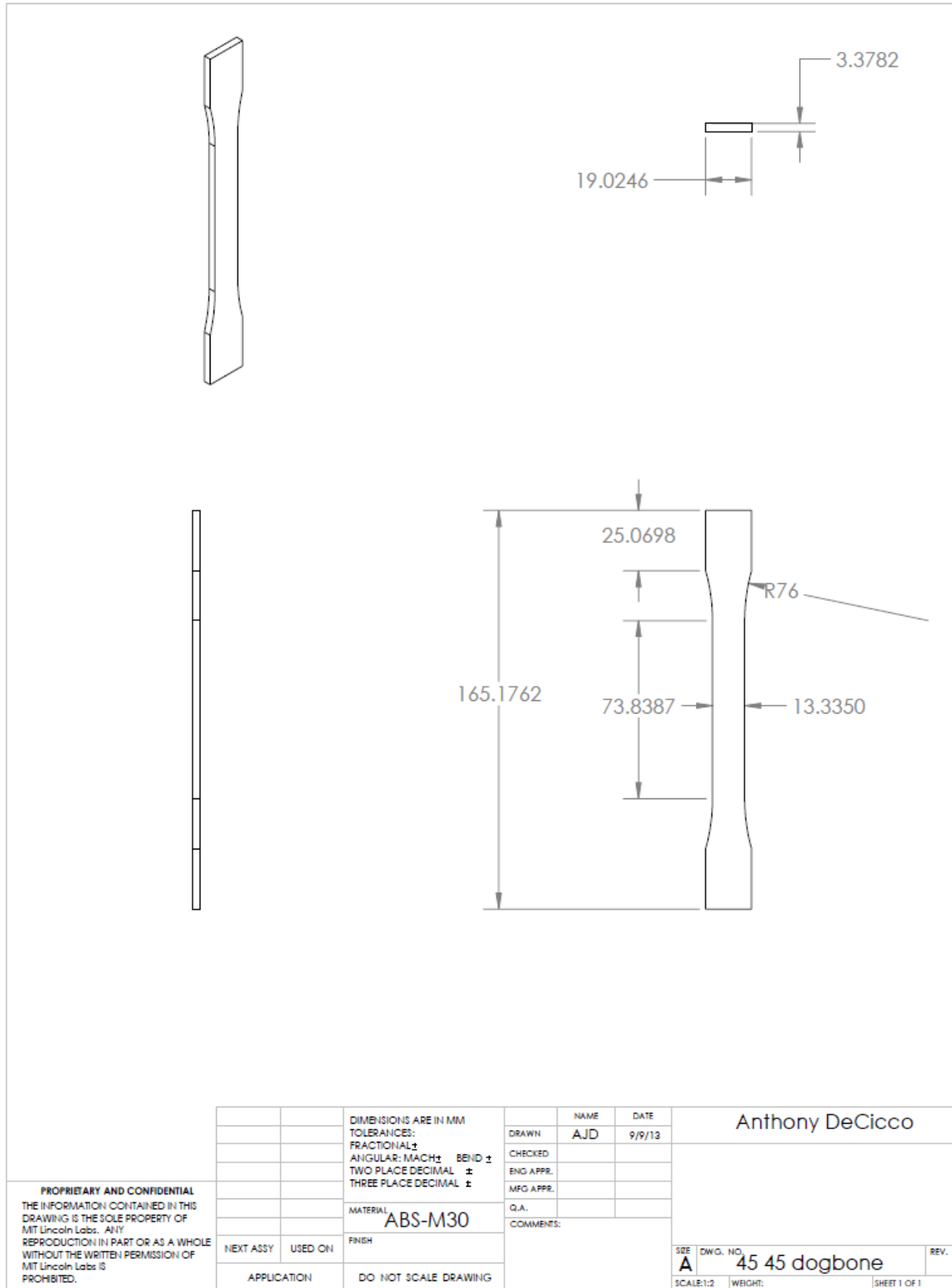


Figure 6-4- Geometry for 45/45 Dogbone

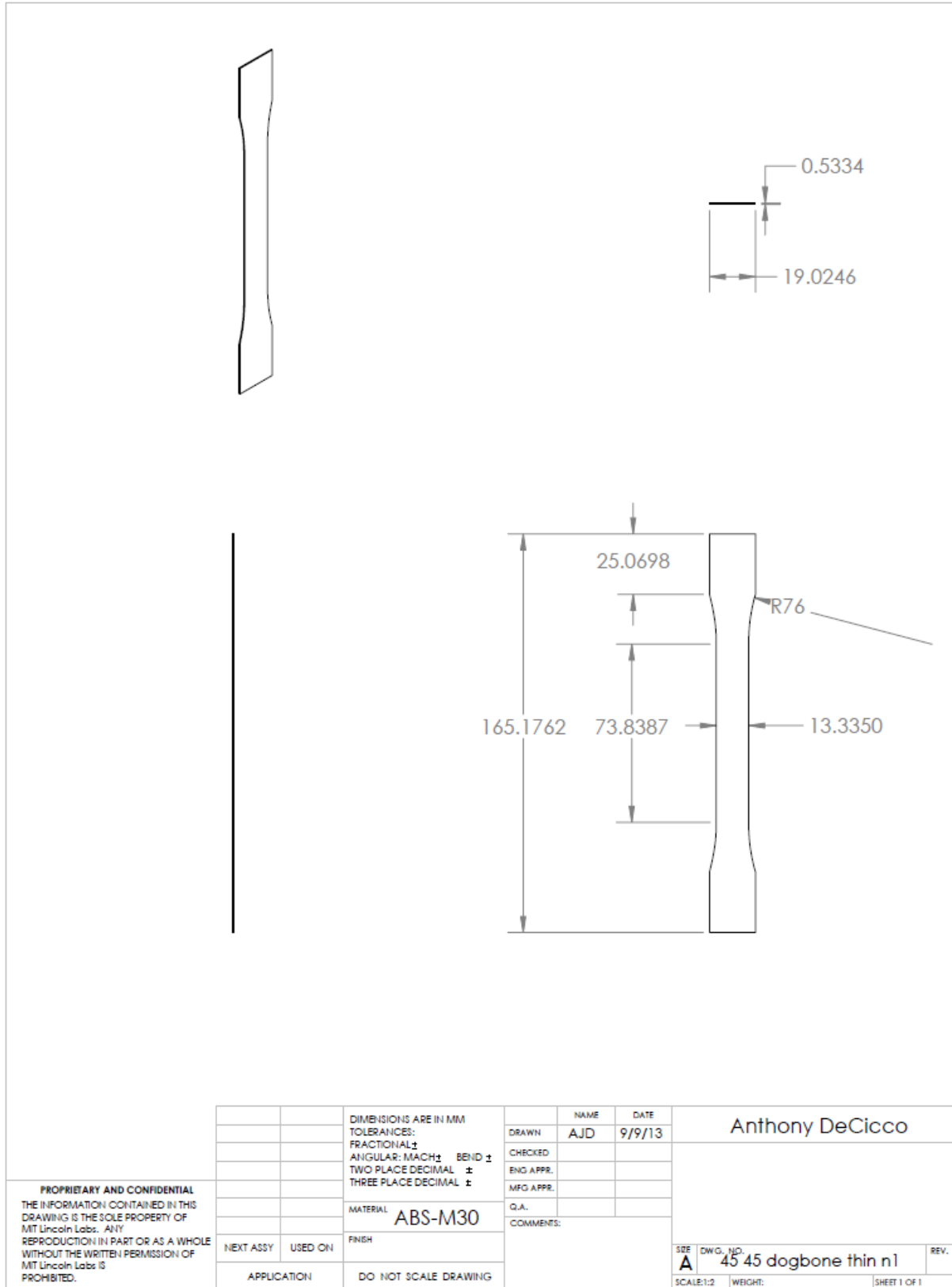


Figure 6-5- Geometry for Thin Specimen of 45/-45 Dogbone



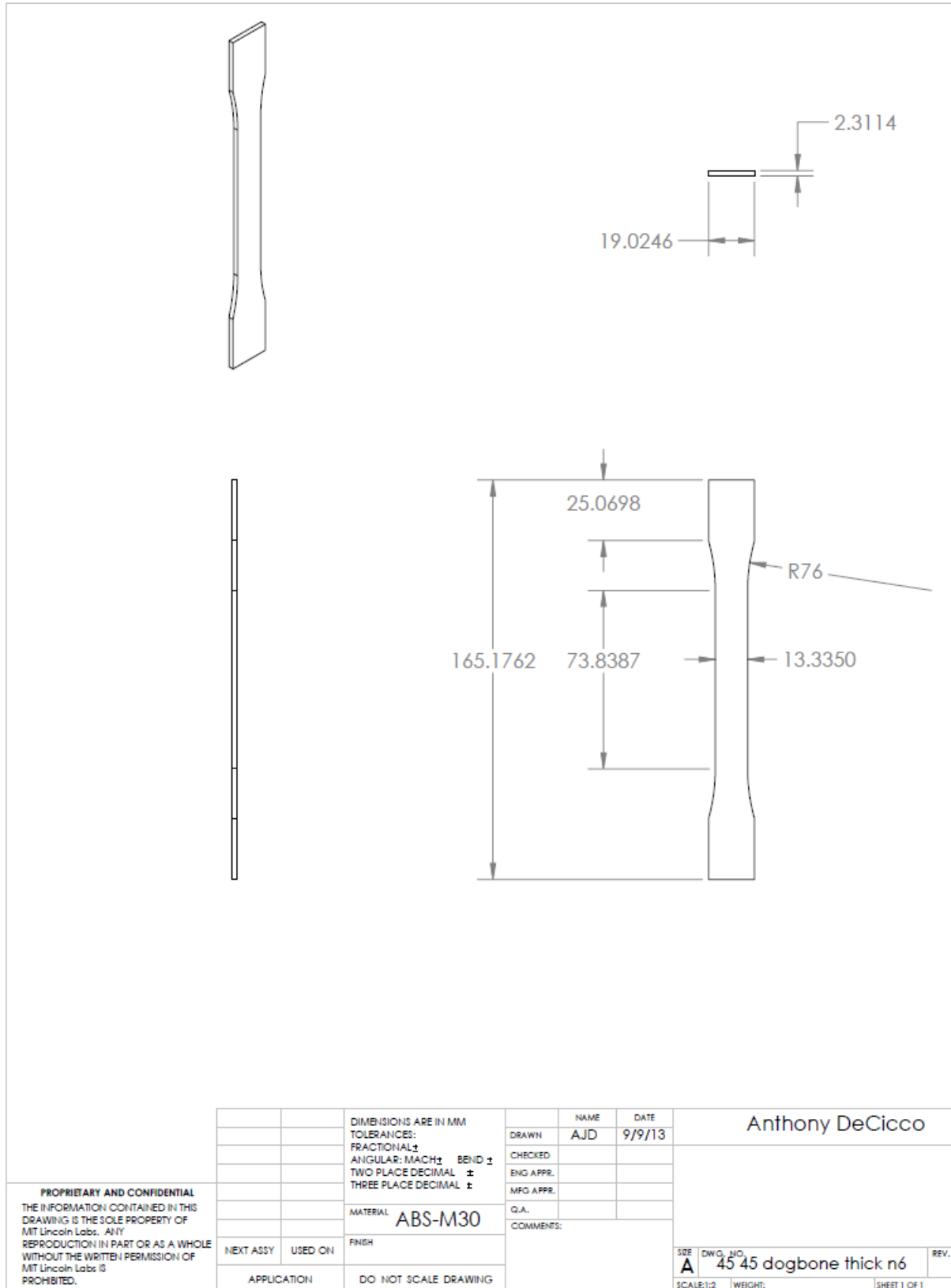


Figure 6-6- Geometry for Thick Specimen of 45/-45 Dogbone

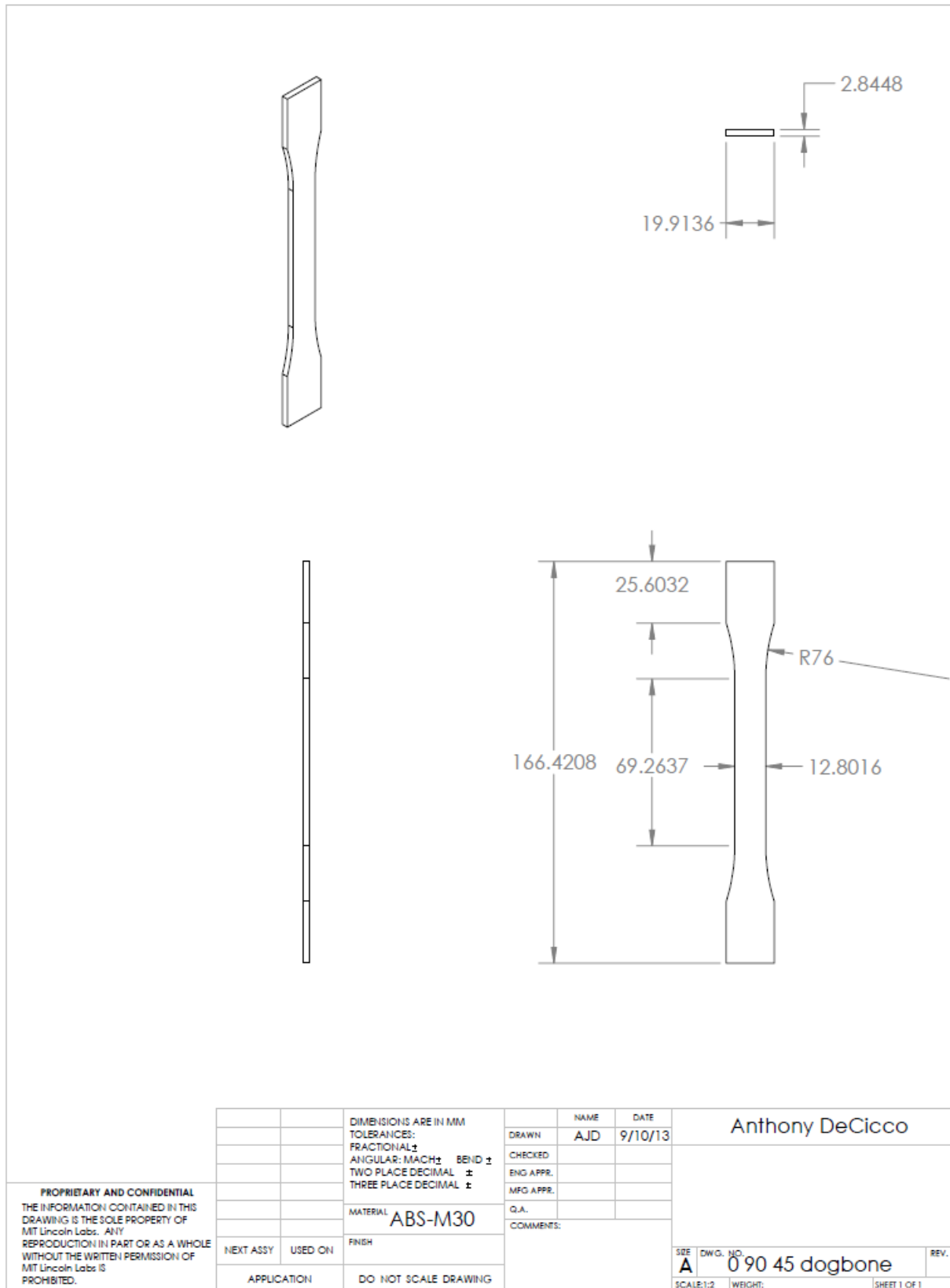


Figure 6-7- Geometry for Quasi Isotropic Dogbone Specimen

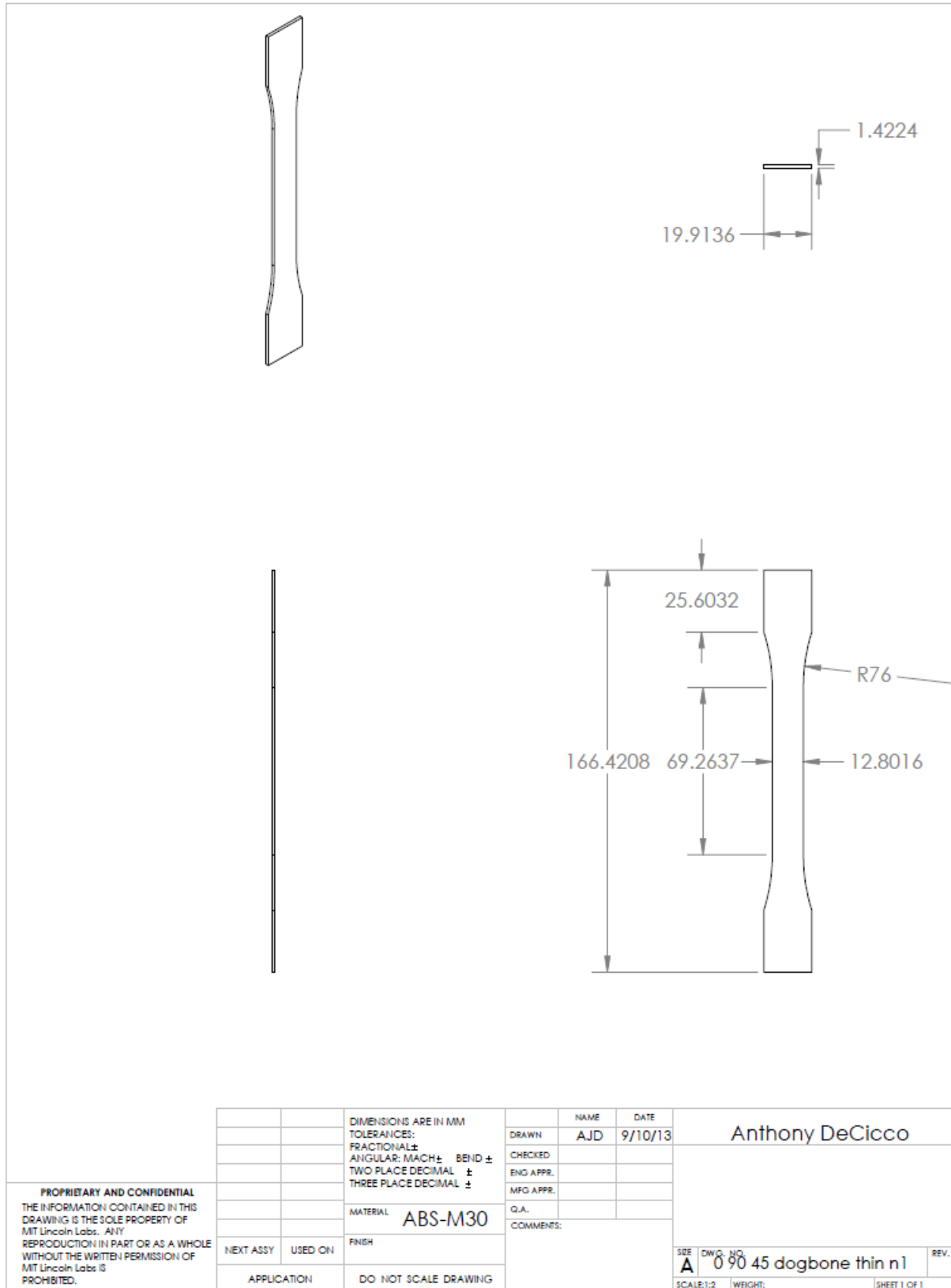


Figure 6-8- Geometry for Thin Quasi Isotropic Dogbone Specimen

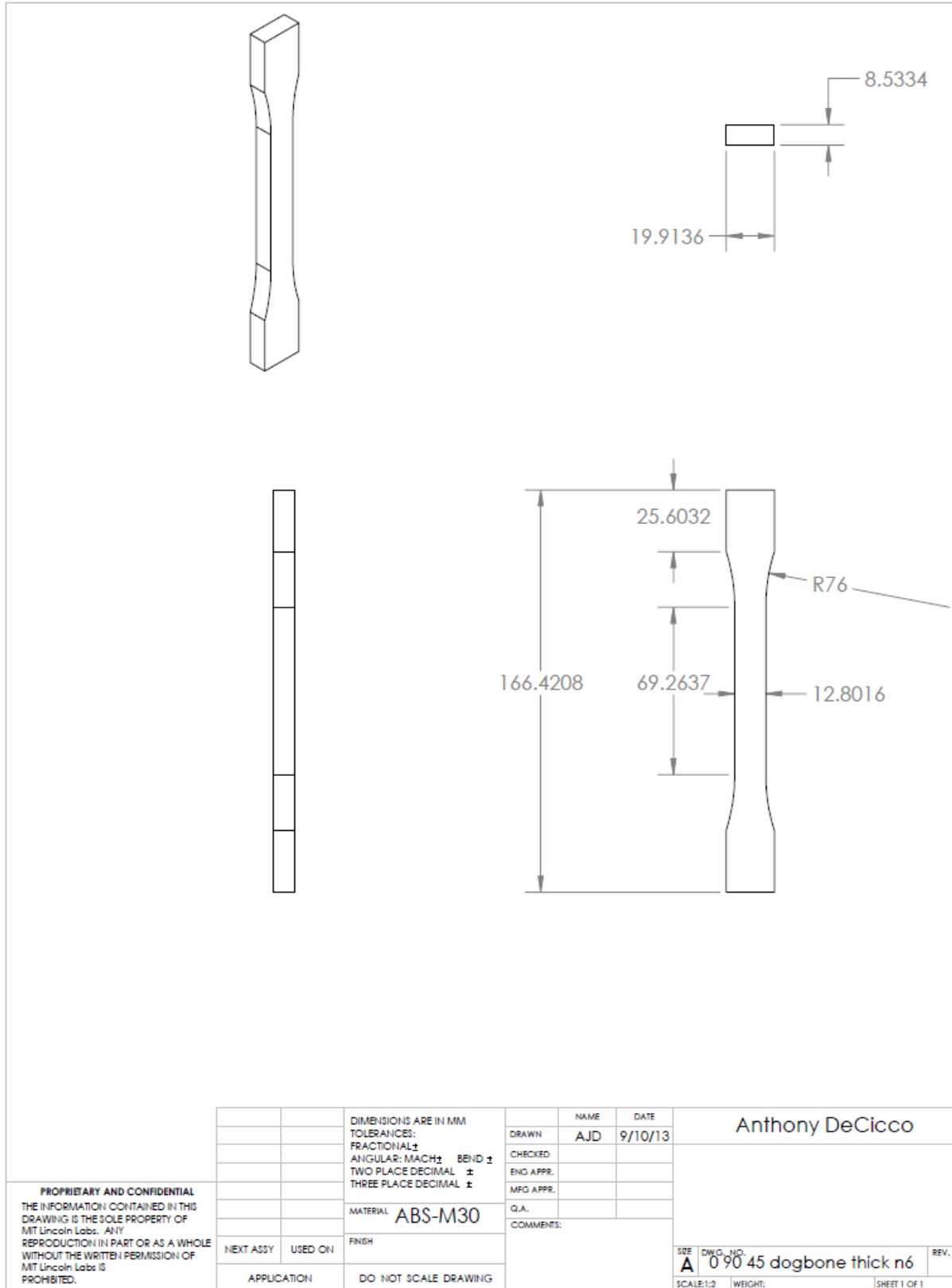


Figure 6-9-Geometry for Thick Quasi Isotropic Dogbone Specimen

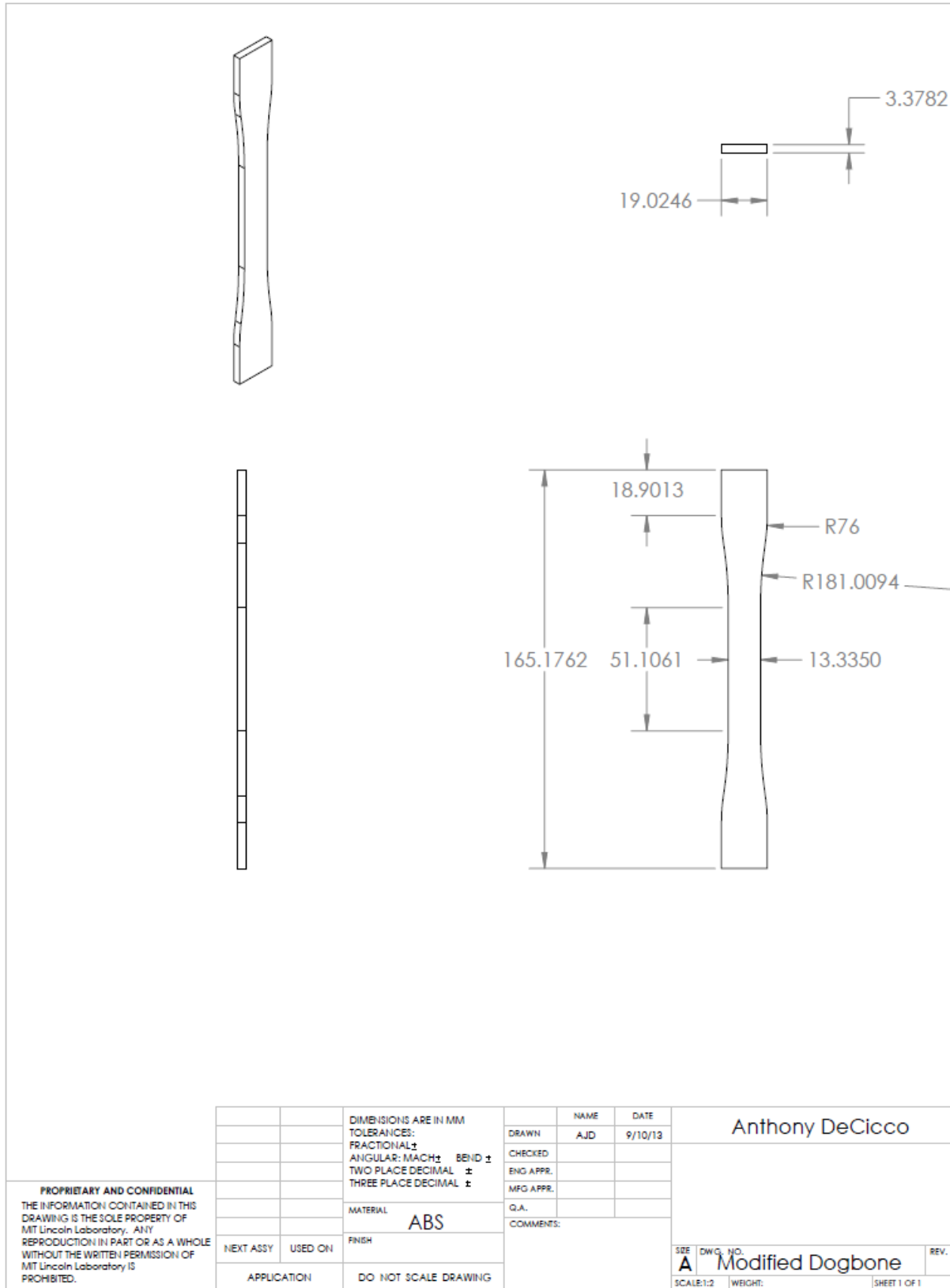


Figure 6-10 Modified Dogbone

## 6.2 Appendix B- Comparing Tensile Results

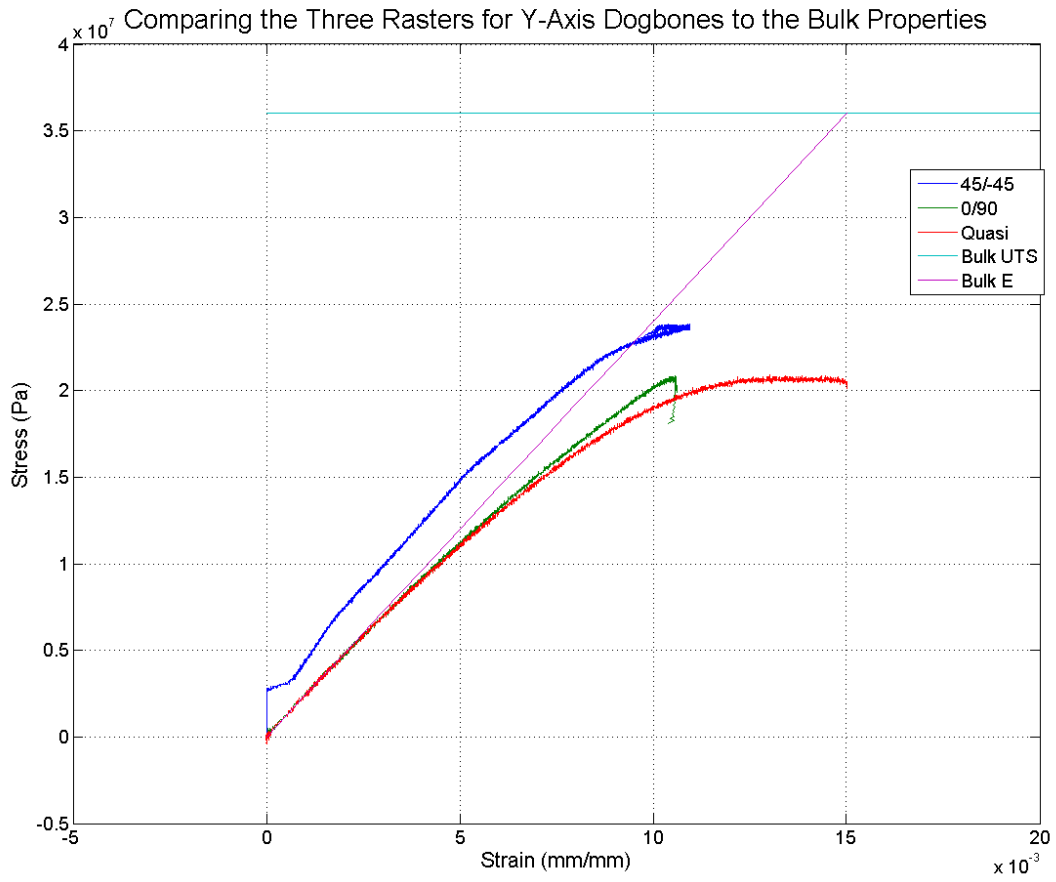


Figure 6-11 Comparing three rasters for Y-Axis Dogbone

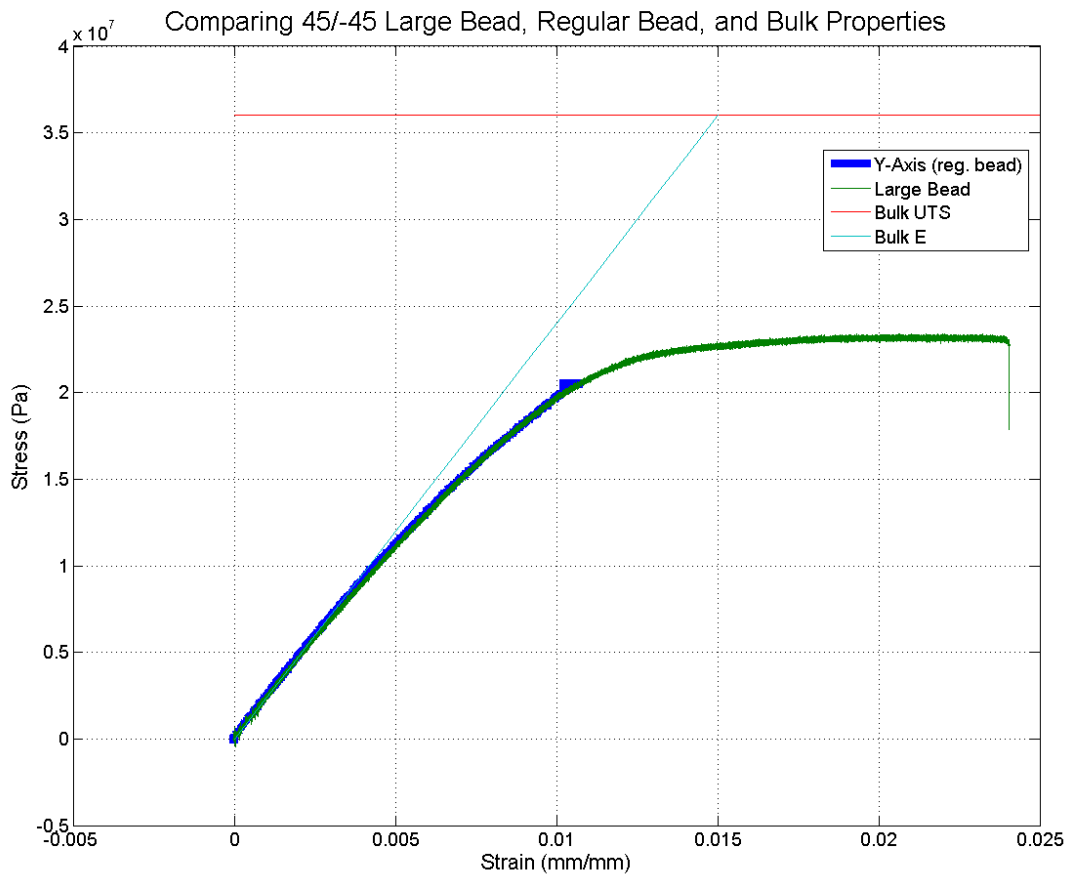


Figure 6-12 Comparing 45/-45 Y-axis with Large Bead (Disclaimer: Strain in Plastic region for n=6 is a result of slip)

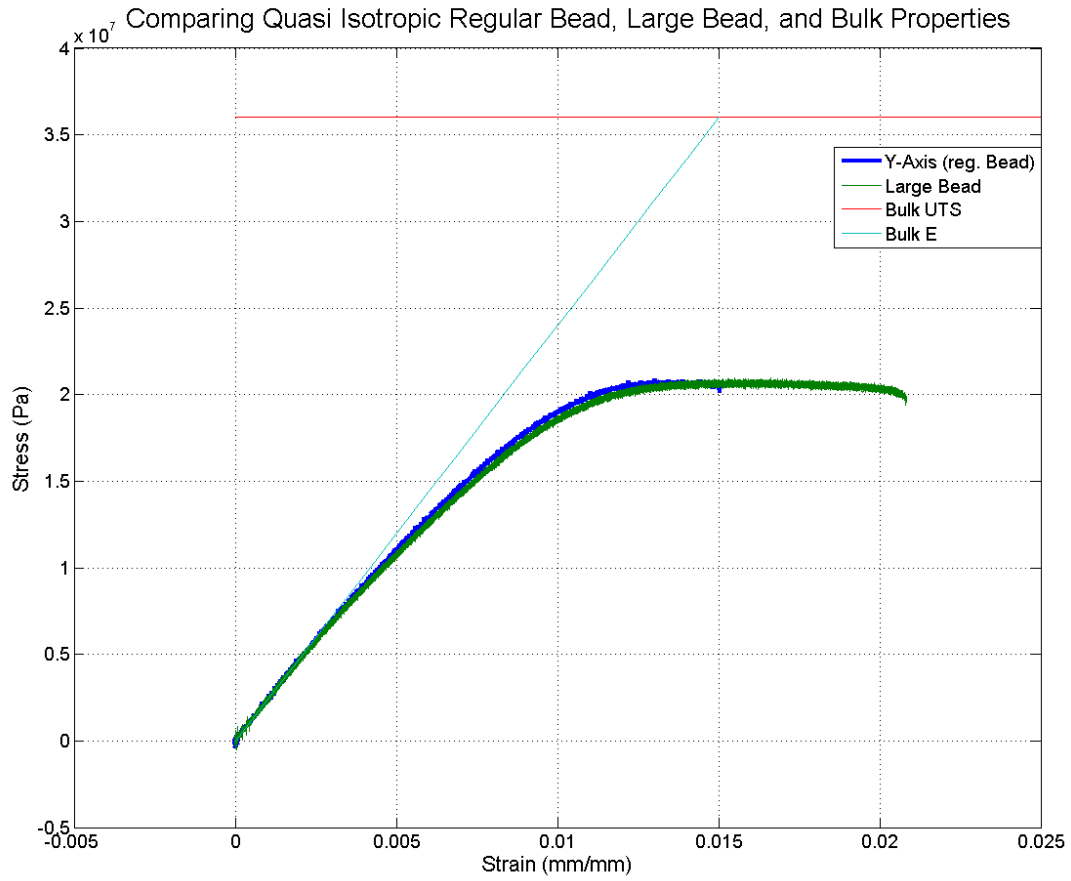


Figure 6-13 Comparing Quasi Isotropic Y-axis with Large Bead



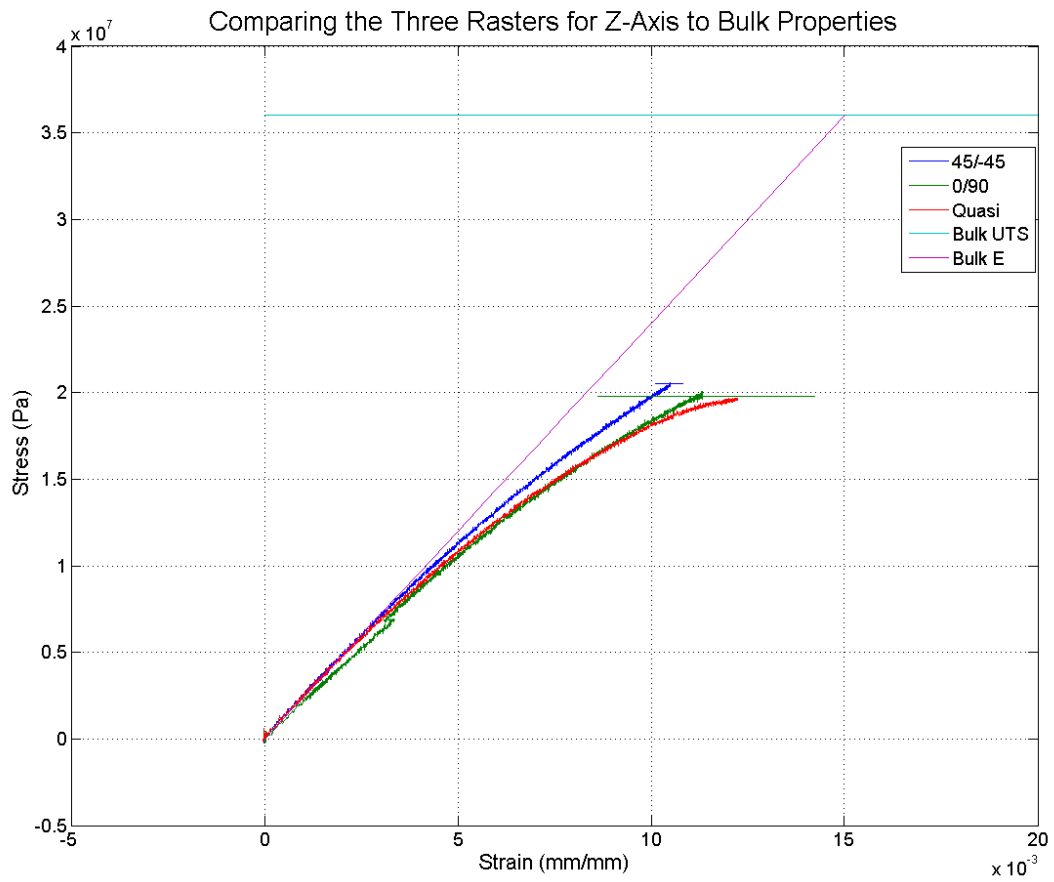


Figure 6-14 Comparing three rasters for Z-axis Dogbones

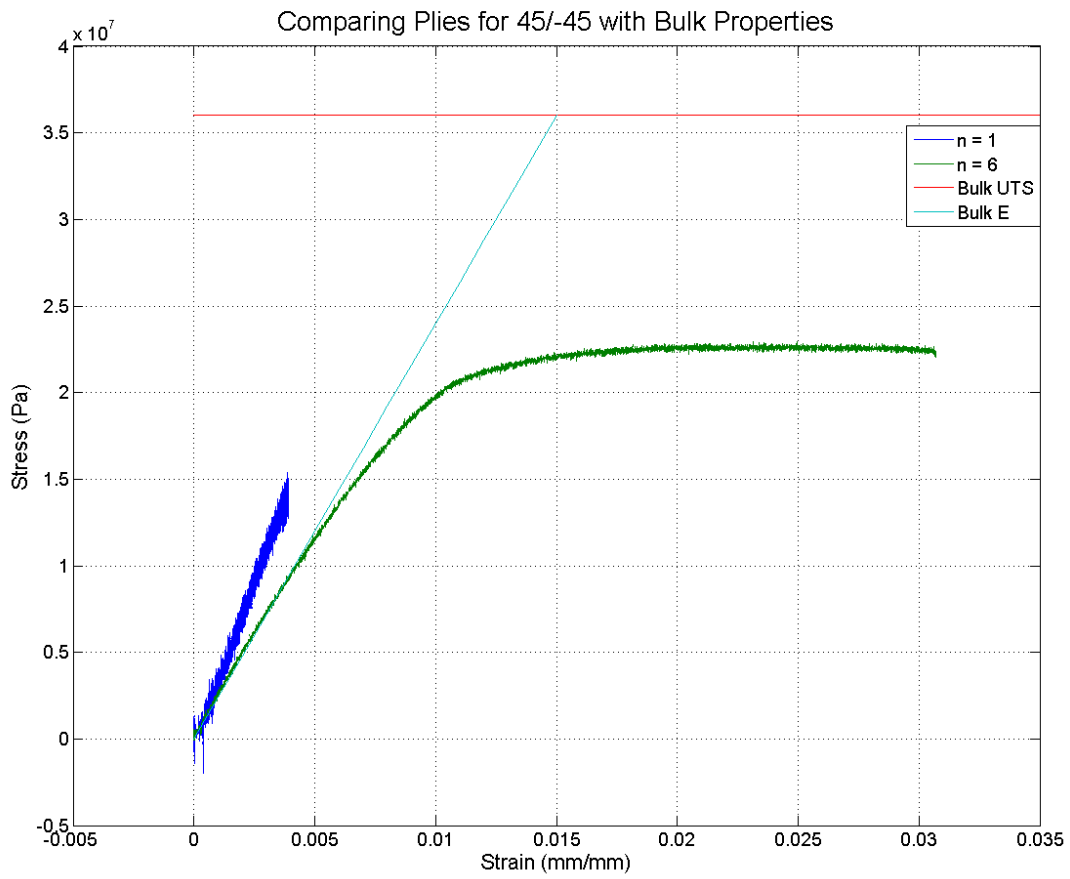


Figure 6-15 Comparing Ply Thickness for 45/-45

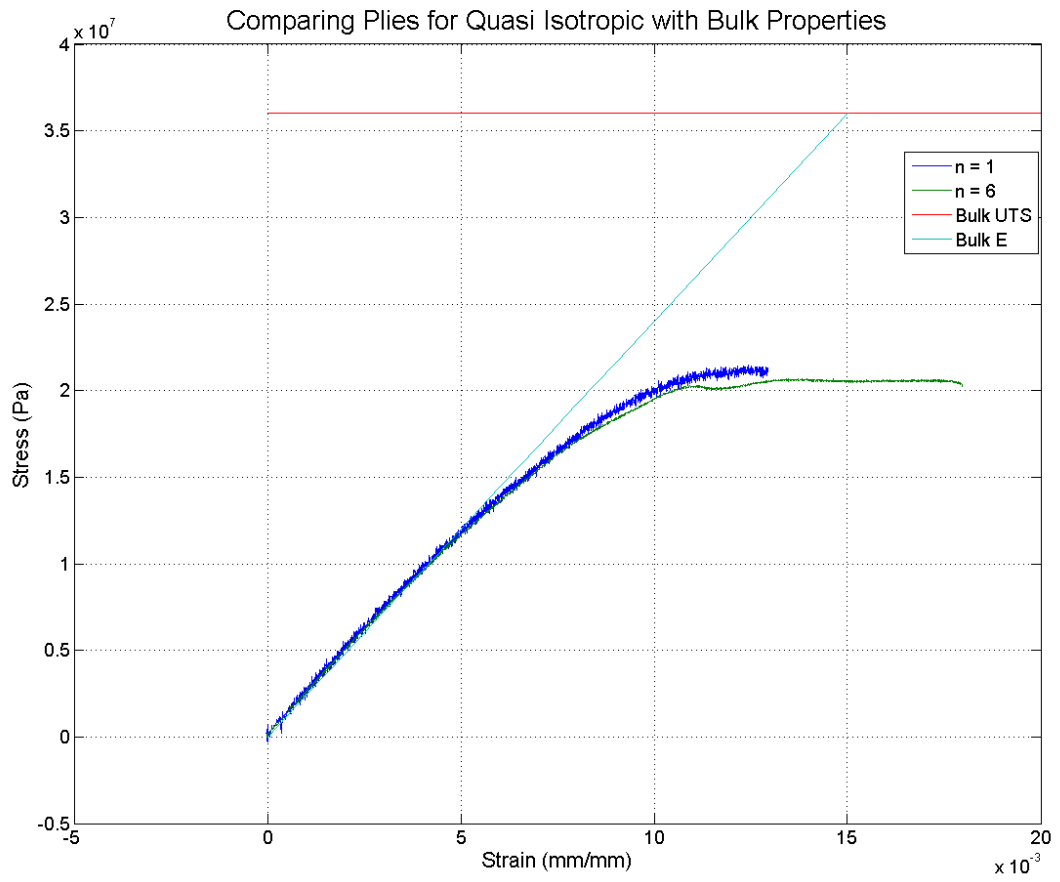


Figure 6-16 Comparing Ply Thickness for Quasi Isotropic

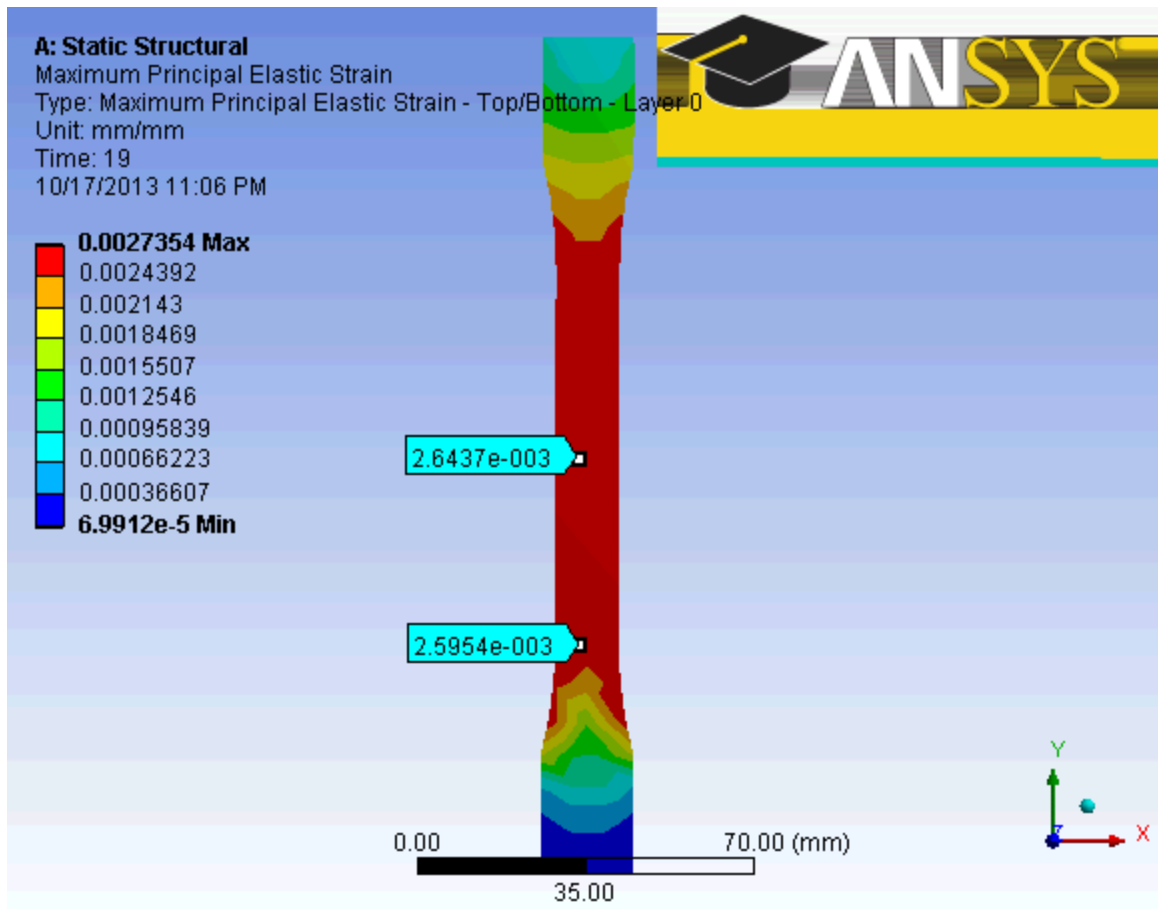


Figure 6-17 0 90 Y-Axis Dogbone at UTS displaying Max Normal Strain

### 6.3 Appendix C- Thermal Test Set-up and Schematics

In this section of the Appendix, we present the full labeled thermal test set up with schematics.

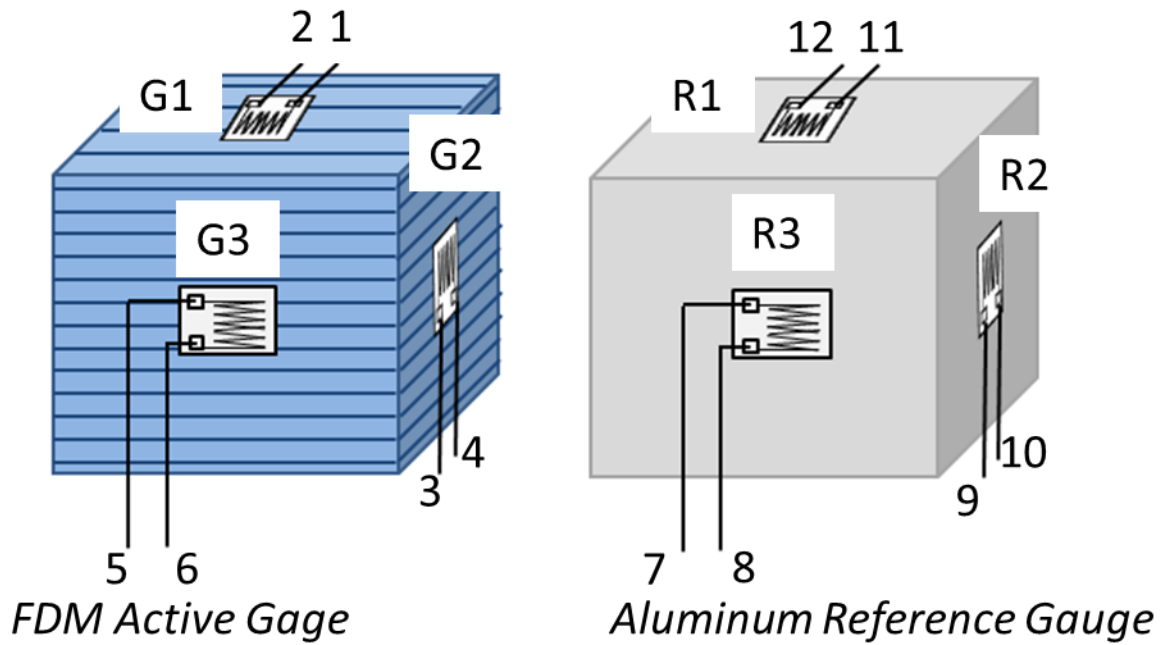


Figure 6-18 Thermal Block with Labeled Wires and Gauges

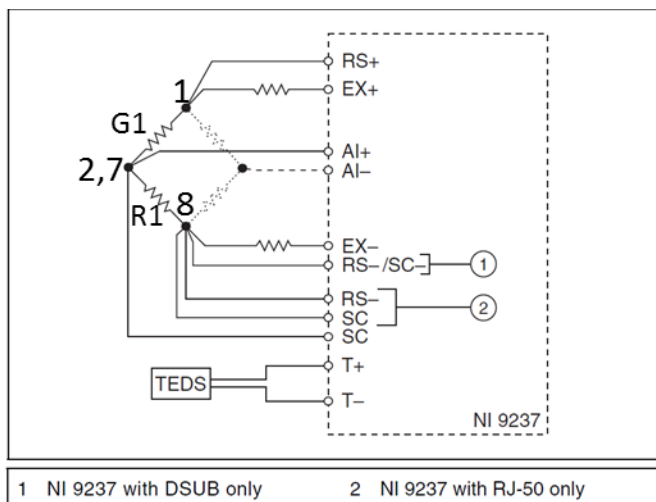


Figure 3. Connecting a Half or Full Bridge to the NI 9237

Figure 6-19 Half Bridge for Gauge 1 Schematic (Modeled using NI)

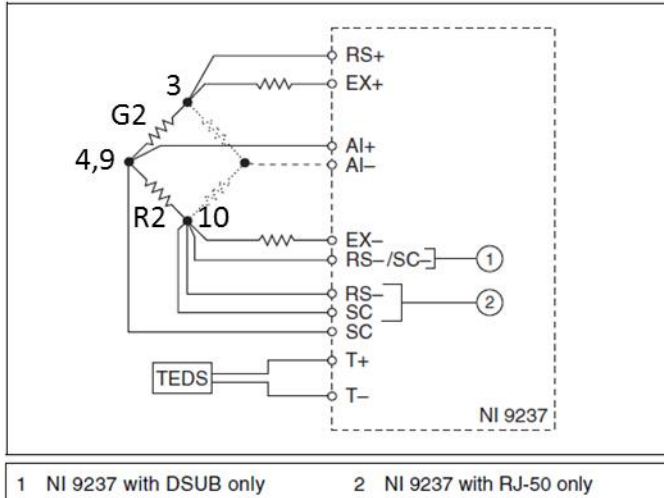


Figure 3. Connecting a Half or Full Bridge to the NI 9237

Figure 6-20 Half Bridge for Gauge 2 Schematic (Modeled using NI)

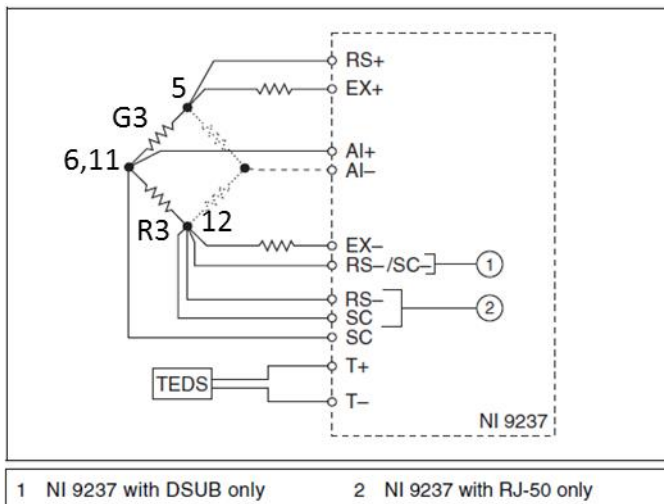


Figure 3. Connecting a Half or Full Bridge to the NI 9237

Figure 6-21 Half Bridge for Gauge 3 Schematic (Modeled using NI)

## 6.4 Appendix B- Matlab functions

Matlab® was used for all data analysis as it was provided by Lincoln Labs and was familiar to the Project Team. In this section, the main functions, sub-functions, and graphing functions are presented. Note that the Graphing functions were generated by Matlab® using the Graphing Editor Interface within the program. Note also that code is repeated between the three main functions for Laminate Analysis to allow the reader ease in understanding the entire process of the code without having to reference previous portions of the Appendix.

### 6.4.1 Stress-Strain Plot Function

```
function PlotStressStrain(X1, Y1)
%CREATEFIGURE1(X1, Y1)
% X1: Strain
% Y1: Stress

% Auto-generated by MATLAB on 06-Oct-2013 18:15:18

% Create figure
figure1 = figure;

% Create axes
axes1 = axes('Parent',figure1,'YGrid','on','XGrid','on');

xlim(axes1,[0 0.025]);
box(axes1,'on');
hold(axes1,'all');

% Create plot
plot(X1,Y1);

% Create xlabel
xlabel('Strain');

% Create ylabel
ylabel('Stress (Pa)');
```

### 6.4.2 Mohr's Circle Plot Function

```
function PlotMohrsCircle (s1, s2, s3)

% Sxx is the Strain From Gage 1
% Syy is the Strain From Gage 2
% Sxy is the Strain From Gage 3

% Create figure
figure2 = figure;

% Create axis
axes2 = axes('Parent',figure2,'YGrid','on','XGrid','on');

% Create xlabel
xlabel('Normal Strain');

% Create ylabel
ylabel('Shear Strain');

% Equations from Vishay
Principa11 = ((s1 + s3)/2) + (1/sqrt(2))*sqrt((s1-s2)^2 + (s2-s3)^2);
Principa12 = ((s1 + s3)/2) - (1/sqrt(2))*sqrt((s1-s2)^2 + (s2-s3)^2);
```

```

theta = (1/2)*atand((S1-2*S2+S3)/(S1-S3));
AVERAGE = (abs(Principa11)+abs(Principa12))/2;
CENTER = [Principa11-AVERAGE,0];
RADIUS = abs(AVERAGE);

% Draw Circle
viscircles(CENTER, RADIUS)

```

### 6.4.3 Plotting Data for 45/-45 Z-axis

#### *Strain*

```

% Data for 45/-45 Z-axis
% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
1.xlsx', '10_04_2013 03_...cDAQ1Mod1_ai2', 'B7995:B42479');
raw(cellfun(@(x) isempty(x) && isnumeric(x) && isnan(x), raw)) = {' '};

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
TestSerial1xx = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw R;

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
1.xlsx', '10_04_2013 03_...cDAQ1Mod1_ai0', 'B7995:B42479');
raw(cellfun(@(x) isempty(x) && isnumeric(x) && isnan(x), raw)) = {' '};

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
TestSerial1yy = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw R;

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
1.xlsx', '10_04_2013 03_...cDAQ1Mod1_ai1', 'B7995:B42479');
raw(cellfun(@(x) isempty(x) && isnumeric(x) && isnan(x), raw)) = {' '};

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

```



```

% Create output variable
TestSerial1xy = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw R;

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
1.xlsx','10_04_2013_03_...cDAQ1Mod1_ai2','A11:A34495');
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {' '};

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
TimeStrain1 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw R;

Strain = [TestSerial1xx TestSerial1xy TestSerial1yy]; % Strain Vector exx,eyy,Gammaxy

```

## Stress

```

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.1.xlsx','dogboneabstensiletest_ser.1','G3:G2141');

% Create output variable
LoadSerial1 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw;

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.1.xlsx','dogboneabstensiletest_ser.1','B3:B2141');

% Create output variable
timeload1 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw;

resampleload1 = interp1(timeload1, LoadSerial1, TimeStrain1); % resample data to match recording
frequencies

StressAVG1 = resampleload1*4.4482/5.24686e-5;% 5.24686e-5 is the average cross-sectional area in
m^2 for Serial 1 and 4.4482 is the conversion factor from lb to N

```

```
UTS1 = max(StressAVG1)% calculate Ultimate Tensile Strength
```

### *Average Stress Strain Plot*

```
X1 = Strain(:,1);
Y1 = StressAVG1(:,1);
PlotStressStrain(X1,Y1)
title('45/-45 Z-axis Average Stress-Strain')
```

### *Principal Strain at Failure*

```
S1 = Strain(2139,3);
S2 = Strain(2139,2);
S3 = Strain(2139,1);

PlotMohrsCircle(S1,S2,S3)
title('45/-45 Z-axis Mohrs Circle at Failure')
```

## 6.4.4 Plotting Data for 45/-45 Y-Axis

### *Strain*

```
% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
2.xlsx', '10_04_2013 01_...cDAQ1Mod1_ai2', 'B15692:B81773', '', @convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans, dateNums, raw) & cellfun('isclass', raw, 'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial2xx = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
2.xlsx', '10_04_2013 01_...cDAQ1Mod1_ai1', 'B15692:B81773', '', @convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};
```

```

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial2xy = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
2.xlsx','10_04_2013_01_...cDAQ1Mod1_ai0','B15692:B81773','',@convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial2yy = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
2.xlsx','10_04_2013_01_...cDAQ1Mod1_ai2','A11:A66092','',@convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
TimeStrain2 = reshape([raw{:}],size(raw));

```

```
% Clear temporary variables
clearvars raw dateNums R;

Strain = [StrainSerial2xx StrainSerial2xy StrainSerial2yy];% exx, eyy, Gammaxy
```

## Stress

```
% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.2.xlsx', 'dogboneabstensiletest_ser.2', 'G3:G4100');

% Create output variable
LoadSerial2 = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw;

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.2.xlsx', 'dogboneabstensiletest_ser.2', 'B3:B4100');

% Create output variable
timeload2 = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw;

resampleLoad2 = interp1(timeload2, LoadSerial2, TimeStrain2); % Resample to match Recording
Frequencies

StressAVG2 = resampleLoad2*4.4482/5.24686e-5;% 5.24686e-5 is the average cross-sectional area in
m^2 and 4.4482 is the conversion factor from lb to N

UTS2 = max(StressAVG2) % Calculate the Ultimate Tensile Strength
```

## Average Stress Strain Plot

```
X1 = Strain(:,1);
Y1 = StressAVG2(:,1);
PlotStressStrain(X1,Y1)
title('45/-45 Y-axis Average Stress-Strain')
```

## Principal Strain at Failure

```

S1 = Strain(66082,3);
S2 = Strain(66082,2);
S3 = Strain(66082,1);

PlotMohrsCircle(S1,S2,S3)
title('45/-45 Y-axis Mohrs Circle at Failure')

```

### 6.4.5 Plotting Data for 0/90 Y-axis

#### Strain

```

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~,~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
3.xlsx','10_04_2013 12_...cDAQ1Mod1_ai2','B47585:B94424','',@convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial3xx = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~,~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
3.xlsx','10_04_2013 12_...cDAQ1Mod1_ai1','B47585:B94424','',@convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial3xy = reshape([raw{:}],size(raw));

```

```

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
3.xlsx', '10_04_2013 12_...cDAQ1Mod1_ai0', 'B47585:B94424', '', @convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans, dateNums, raw) & cellfun('isclass', raw, 'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial3yy = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
3.xlsx', '10_04_2013 12_...cDAQ1Mod1_ai2', 'A11:A46850', '', @convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans, dateNums, raw) & cellfun('isclass', raw, 'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
TimeStrain3 = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw dateNums R;

Strain = [StrainSerial3xx StrainSerial3xy StrainSerial3yy]; % exx, eyy, Gammaxy

```

## Stress

```

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.3.xlsx', 'dogboneabstensiletest_ser.3', 'G3:G2907');

```

```

% Create output variable
LoadSerial3 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw;

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.3.xlsx','dogboneabstensiletest_ser.3','B3:B2907');

% Create output variable
timeLoad3 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw;

resampleLoad3 = interp1(timeLoad3, LoadSerial3, TimeStrain3); % resample Data to match recording
frequencies

StressAVG3 = resampleLoad3*4.4482/5.24686e-5;% 5.24686e-5 is the average cross-sectional area in
m^2 and 4.4482 is the conversion factor from lb to N

UTS3 = max(StressAVG3)% calculate Ultimate Tensile Strength

```

### *Average Stress Strain Plot*

```

X1 = Strain(:,1);
Y1 = StressAVG3(:,1);

PlotStressStrain(X1,Y1)
title('0/90 Y-axis Average Stress-Strain')

```

### *Principal Strain at Failure*

```

S1 = Strain(46840,3);
S2 = Strain(46840,2);
S3 = Strain(46840,1);

PlotMohrsCircle(S1,S2,S3)
title('0/90 Y-axis Mohrs Circle at Failure')

```

## 6.4.6 Plotting Data for 45/-45 n = 1

### Strain

```

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
4.xlsx','10_04_2013 02_...cDAQ1Mod1_ai2','B16531:B87337','',@convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial4xx = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
4.xlsx','10_04_2013 02_...cDAQ1Mod1_ai1','B16531:B87337','',@convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial4xy = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
4.xlsx','10_04_2013 02_...cDAQ1Mod1_ai0','B16531:B87337','',@convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates

```



```

raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial4yy = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
4.xlsx','10_04_2013_02_...cDAQ1Mod1_ai2','A11:A70817','',@convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
TimeStrain4 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

Strain = [StrainSerial4xx StrainSerial4xy StrainSerial4yy];% exx, eyy, Gammaxy

% Stress

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiilettest_ser.4.xlsx','dogboneabstensiilettest_ser.4','G3:G4393');
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
LoadSerial4= reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw R;

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of

```

```

Tensile\dogboneabstensiletest_ser.4.xlsx', 'dogboneabstensiletest_ser.4', 'B3:B4393');
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
timeload4 = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw R;

resampleLoad4 = interp1(timeload4, LoadSerial4, TimeStrain4); % Resample to match recording
frequencies

StressAVG4 = resampleLoad4*4.4482/8.2845e-6;% 8.2845e-6 is the average cross-sectional area in
m^2 for Serial 4 and 4.4482 is the conversion factor from lb to N

UTS4 = max(StressAVG4)% Calculate Ultimate Tensile Strength

```

### *Average Stress Strain Plot*

```

X1 = Strain(:,1);
Y1 = StressAVG4(:,1);

PlotStressStrain(X1,Y1)
title('45/-45 n=1 Average Stress-Strain')

```

### *Principal Strain at Failure*

```

S1 = Strain(70807,3);
S2 = Strain(70807,2);
S3 = Strain(70807,1);

PlotMohrsCircle(S1,S2,S3)
title('45/-45 n=1 Mohrs Circle at Failure')

```

## 6.4.7 Plotting data for 45/-45 n = 6

### *Strain*

```

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.5.xlsx', 'dogboneabstensiletest_ser.5', 'L3:L6088');
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

```

```

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
TensileTest5xx = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw R;

```

## Import Time Scales

```

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
5.xlsx','10_04_2013_02_...cDAQ1Mod1_ai2','A11:A98156','',@convertSpreadsheetDates);
raw(cellfun(@(x) isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
TimeStrain5 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.5.xlsx','dogboneabstensiletest_ser.5','B3:B6088');
raw(cellfun(@(x) isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
timeload5 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw R;

resamplestrain5 = interp1(timeload5, TensileTest5xx, TimeStrain5); % Resample to match recording
frequencies

Strain = [resamplestrain5];% exx

```

## Stress

```

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.5.xlsx','dogboneabstensiletest_ser.5','G3:G6088');
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
LoadSerial5 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw R;

resampleLoad5 = interp1(timeLoad5, LoadSerial5, TimeStrain5);

StressAVG5 = resampleLoad5*4.4482/3.5900e-5; % 3.5900e-5 is the average cross-sectional area in
m^2 and 4.4482 is the conversion factor from lb to N

UTS5 = max(StressAVG5) % Calculate Ultimate Tensile Strength

```

## Average Stress Strain Plot

```

X1 = Strain(:,1);
Y1 = StressAVG5(:,1);
PlotStressStrain(X1,Y1)

title('45/-45 n=6 Average Stress-Strain')

```

## 6.4.8 Plotting Data for 45/-45 Large Bead

### Strain

```

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.7.xlsx','dogboneabstensiletest_ser 7','M3:M119124');
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
TensileTest7xx = reshape([raw{:}],size(raw));

```

```
% Clear temporary variables
clearvars raw R;

Strain = [TensileTest7xx]; % exx
```

## Stress

```
% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.7.xlsx', 'dogboneabstensiletest_ser 7', 'G3:G119124');
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
LoadSerial7 = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw R;

StressAVG7 = LoadSerial7*4.4482/5.24686e-5;% 5.24686e-5 is the average cross-sectional area in
m^2 and 4.4482 is the conversion factor from lb to N

UTS7 = max(StressAVG7) % Calculate Ultimate Tensile Stress
```

## Average Stress Strain Plot

```
X1 = Strain(:,1);
Y1 = StressAVG7(:,1);

PlotStressStrain(X1,Y1)
title('45/-45 Large Bead Average Stress-Strain')
```

## 6.4.9 Plotting Data for Quasi Isotropic Y-axis

### Strain

```
% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.8.xlsx', 'dogboneabstensiletest_ser.8', 'L3:L3005');

% Create output variable
StrainSerial8xx = reshape([raw{:}], size(raw));
```

```

% Clear temporary variables
clearvars raw;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
8.xlsx', '10_04_2013 01_...cDAQ1Mod1_ai1', 'B24669:B73088', '', @convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans, dateNums, raw) & cellfun('isclass', raw, 'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial8xy = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
8.xlsx', '10_04_2013 01_...cDAQ1Mod1_ai0', 'B24669:B73088', '', @convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans, dateNums, raw) & cellfun('isclass', raw, 'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial8yy = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw dateNums R;

```

### *Importing Time scales*

```

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
8.xlsx', '10_04_2013 01_...cDAQ1Mod1_ai2', 'A11:A48430', '', @convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans, dateNums, raw) & cellfun('isclass', raw, 'char'); % Find

```

```

spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
TimeStrain8 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiilettest_ser.8.xlsx','dogboneabstensiilettest_ser.8','B3:B3005');

% Create output variable
timeLoad8 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw;

resampleStrain8xx = interp1(timeLoad8, StrainSerial8xx, TimeStrain8); % Resample data to match
recording frequencies

Strain = [resampleStrain8xx StrainSerial8xy StrainSerial8yy];% exx, eyy, Gammaxy

% Stress

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiilettest_ser.8.xlsx','dogboneabstensiilettest_ser.8','G3:G3005');

% Create output variable
LoadSerial8 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw;

resampleLoad8 = interp1(timeLoad8, LoadSerial8, TimeStrain8);

StressAVG8 = resampleLoad8*4.4482/4.44876e-5;% 4.44876e-5 is the average cross-sectional area in
m^2 for Serial 8, 4.4482 converts lbs to Newtons

UTS8 = max(StressAVG8)% Calcualte Ultimate Tensile Strength

```

### *Average Stress Strain Plot*

```

X1 = Strain(:,1);
Y1 = StressAVG8(:,1);

```

```
PlotStressStrain(X1,Y1)
title('Quasi Y-axis Average Stress-Strain')
```

### *Principal Strain at Failure*

```
S1 = Strain(48420,3);
S2 = Strain(48420,2);
S3 = Strain(48420,1);

PlotMohrsCircle(S1,S2,S3)
title('Quasi Y-axis Mohrs Circle at Failure')
```

## 6.4.10 Plotting Data for Quasi Isotropic n = 1

### *Time Scale*

```
% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstest_ser.10.xls', 'dogboneabstest_ser.10', 'B3:B2585');

% Create output variable
loadtime10 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
10.xlsx', '10_04_2013_01_...cDAQ1Mod1_ai2', 'A11:A41656', '@convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithNaNs,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainTime10 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;
```



## Strain

```

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabststentest_ser.10.xls', 'dogboneabststentest_ser.10', 'L3:L2585');

% Create output variable
StrainSerial10xx = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw;

resampleStrain10xx = interp1(loadtime10, StrainSerial10xx, StrainTime10);

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
10.xlsx', '10_04_2013 01_...cDAQ1Mod1_ai1', 'B129927:B171572', '', @convertspreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans, dateNums, raw) & cellfun('isclass', raw, 'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial10xy = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
10.xlsx', '10_04_2013 01_...cDAQ1Mod1_ai0', 'B129927:B171572', '', @convertspreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans, dateNums, raw) & cellfun('isclass', raw, 'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial10yy = reshape([raw{:}], size(raw));

```

```
% Clear temporary variables
clearvars raw dateNums R;

Strain10 = [resampleStrain10xx StrainSerial10xy StrainSerial10yy];% exx, eyy, Gammaxy
```

## Stress

```
% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.10.xls', 'dogboneabstensiletest_ser.10', 'G3:G2585');

% Create output variable
LoadSerial10 = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw;

resampleload10 = interp1(loadtime10, LoadSerial10, StrainTime10); % Resample to match recording
frequencies

StressAVG10 = resampleload10*4.4482/2.2244e-5;% 2.2244e-5 is the average cross-sectional area in
m^2 and 4.4482 is the conversion factor from lb to N

UTS10 = max(StressAVG10)% Calculate Ultimate Tensile Strength
```

## Average Stress Strain Plot

```
X1 = Strain10(:,1);
Y1 = StressAVG10(:,1);

PlotStressStrain(X1,Y1)
title('Quasi Isotropic n=1 Average Stress-Strain')
```

## Principal Strain at Failure

```
S1 = Strain10(41646,3);
S2 = Strain10(41646,2);
S3 = Strain10(41646,1);

PlotMohrsCircle(S1,S2,S3)
title('Quasi Isotropic n=1 Mohrs Circle at Failure')
```

## 6.4.11 Plotting Data for Quasi Isotropic n = 6

### Import Time Scales

```

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.11.xlsx', 'dogboneabstensiletest_ser.11', 'B3:B3592');

% Create output variable
loadtime11 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
11.xlsx', '10_04_2013 02_...cDAQ1Mod1_ai2', 'A11:A57898', '', @convertSpreadsheetDates);
raw(cellfun(@(x) isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainTime11 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

```

### Strain

```

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.11.xlsx', 'dogboneabstensiletest_ser.11', 'L3:L3592');

% Create output variable
StrainSerial11xx = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
11.xlsx', '10_04_2013 02_...cDAQ1Mod1_ai1', 'B57513:B115400', '', @convertSpreadsheetDates);
raw(cellfun(@(x) isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

```

```

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial11xy = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
11.xlsx','10_04_2013 02_...cDAQ1Mod1_ai0','B57513:B115400','',@convertSpreadsheetDates);
raw(cellfun(@(x) isempty(x) && isnumeric(x) && isnan(x),raw)) = {' '};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial11yy = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

resampleStrain11xx = interp1(loadtime11, StrainSerial11xx, StrainTime11);% Resample to match non-
equal recording frequencies

Strain11 = [resampleStrain11xx StrainSerial11xy StrainSerial11yy];% exx, eyy, Gammaxy

```

## Stress

```

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.11.xlsx','dogboneabstensiletest_ser.11','G3:G3592');

% Create output variable
LoadSerial11 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw;

```

```

resampleLoad11 = interp1(loadtime11, LoadSerial11, StrainTime11);% Resample data to match

StressAVG11 = resampleLoad11*4.4482/13.34473e-5;% 13.34473e-5 is the average cross-sectional area
in m^2 for Serial 11 and 4.4482 is the conversion factor from lb to N

UTS11 = max(StressAVG11)% Calculate Ultimate Tensile Strength

```

### *Average Stress Strain Plot*

```

X1 = Strain11(:,1);
Y1 = StressAVG11(:,1);

PlotStressStrain(X1,Y1)
title('Quasi n=6 Average Stress-Strain')

```

### *Principal Strain at Failure*

```

S1 = Strain11(57888,3);
S2 = Strain11(57888,2);
S3 = Strain11(57888,1);

PlotMohrsCircle(S1,S2,S3)
title('Quasi n=6 Mohrs Circle at Failure')

```

## 6.4.12 Plotting Data for Quasi Isotropic Large Bead

### *Strain*

```

% Import the data
[~,~,raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.13.xlsx','dogboneabstensiletest_ser 13 (2','L3:L103882');
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial13xx = reshape([raw{:}],size(raw));

```

```

% Clear temporary variables
clearvars raw R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
13 missing lateral.xlsx','10_07_2013
04_...cDAQ1Mod1_a11','B13883:B117762','',@convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial13xy = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
13 missing lateral.xlsx','10_07_2013
04_...cDAQ1Mod1_a10','B13883:B117762','',@convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial13yy = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

Strain13 = [StrainSerial13xx StrainSerial13xy StrainSerial13yy];% exx, eyy, Gammaxy

```

## Stress

```

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.13.xlsx','dogboneabstensiletest_ser 13 (2','G3:G103882');

```

```

raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
LoadSerial13 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw R;

StressAVG13 = LoadSerial13*4.4482/4.4488e-5;% 4.4488e-5 is the average cross-sectional area in
m^2 and 4.4482 is the conversion factor from lb to N

UTS13 = max(StressAVG13)% calculate Ultimate Tensile Strength

```

### *Average Stress Strain Plot*

```

X1 = Strain13(:,1);
Y1 = StressAVG13(:,1);

PlotStressStrain(X1,Y1)
title('Quasi Isotropic Large Bead Average Stress-Strain')

```

### *Principal Strain at Failure*

```

S1 = Strain13(103880,3);
S2 = Strain13(103880,2);
S3 = Strain13(103880,1);

PlotMohrsCircle(S1,S2,S3)
title('Quasi Isotropic Large Bead Mohrs Circle at Failure')

```

## 6.4.13 Plotting Data for Quasi Isotropic Z-axis

### Strain

```

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
14.xlsx', '10_04_2013 03_...cDAQ1Mod1_ai2', 'B27501:B65081', '', @convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans, dateNums, raw) & cellfun('isclass', raw, 'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial14xx = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
14.xlsx', '10_04_2013 03_...cDAQ1Mod1_ai1', 'B27501:B65081', '', @convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans, dateNums, raw) & cellfun('isclass', raw, 'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial14xy = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
14.xlsx', '10_04_2013 03_...cDAQ1Mod1_ai0', 'B27501:B65081', '', @convertSpreadsheetDates);
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans, dateNums, raw) & cellfun('isclass', raw, 'char'); % Find
spreadsheet dates

```



```

raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial14yy = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

% Import the data, extracting spreadsheet dates in MATLAB serial date number format (datenum)
[~, ~, raw, dateNums] = xlsread('C:\Users\anthony.decicco\Desktop\Results of Tensile\Test Serial
14.xlsx','10_04_2013 03_...cDAQ1Mod1_ai2','A11:A37591','',@convertSpreadsheetDates);
raw(cellfun(@(x) isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace date strings by MATLAB serial date numbers (datenum)
R = ~cellfun(@isequalwithequalnans,dateNums,raw) & cellfun('isclass',raw,'char'); % Find
spreadsheet dates
raw(R) = dateNums(R);

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
TimeStrain14 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw dateNums R;

Strain = [StrainSerial14xx StrainSerial14xy StrainSerial14yy]; % exx, eyy, Gammaxy

```

## Stress

```

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.14.xlsx','dogboneabstensiletest_ser.14','G3:G2333');

% Create output variable
LoadSerial14 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw;

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.14.xlsx','dogboneabstensiletest_ser.14','B3:B2333');

```

```

% Create output variable
timeLoad14 = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw;

resampleLoad14 = interp1(timeLoad14, LoadSerial14, TimeStrain14);% resample to match recording
frequencies

StressAVG14 = resampleLoad14*4.4482/4.44876e-5;% 4.44876e-5 is the average cross-sectional area
in m^2 for Serial 8, 4.4482 converts lbs to Newtons

UTS14 = max(StressAVG14)% Calculate Ultimate Tensile Strength

```

### *Average Stress Strain Plot*

```

X1 = Strain(:,1);
Y1 = StressAVG14(:,1);

PlotStressStrain(X1,Y1)
title('Quasi Z-axis Average Stress-Strain')

```

### *Principal Strain at Failure*

```

S1 = Strain(27581,3);
S2 = Strain(27581,2);
S3 = Strain(27581,1);

PlotMohrsCircle(S1,S2,S3)
title('Quasi Z-axis Mohrs Circle at Failure')

```

## 6.4.14 Plotting data for 0/90 Z-axis

### *Strain*

```

% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.15.xlsx', 'dogboneabstensiletest_ser.15', 'L3:L2422');
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
StrainSerial15xx = reshape([raw{:}],size(raw));

```

```
% Clear temporary variables
clearvars raw R;

Strain15 = [StrainSerial15xx]; % exx
```

## Stress

```
% Import the data
[~, ~, raw] = xlsread('C:\Users\anthony.decicco\Desktop\Results of
Tensile\dogboneabstensiletest_ser.15.xlsx', 'dogboneabstensiletest_ser.15', 'G3:G2422');
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x), raw)) = {''};

% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x), raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

% Create output variable
LoadSerial15 = reshape([raw{:}], size(raw));

% Clear temporary variables
clearvars raw R;

StressAVG15 = LoadSerial15*4.4482/5.2469e-5; % 5.2469e-5 is the average cross-sectional area in
m^2 for Serial 8, 4.4482 converts lbs to Newtons

UTS15 = max(StressAVG15) % calculate Ultimate Tensile Strength
```

## Average Stress Strain Plot

```
X1 = Strain15(:,1);
Y1 = StressAVG15(:,1);

PlotStressStrain(X1,Y1)
title('0/90 Z-axis Average Stress-Strain')
```

## 6.4.15 Determining the Stress and Strain Relationships in a Quasi-Isotropic Laminate

### Constants

```
E_1 = 1904e6; % Young's Modulus in the 1-direction (Pa)
E_2 = 2228e6; % Young's Modulus in the 2-direction (Pa)
G_12 = 896.9e6; % Shear Modulus in the 1-2 (Pa)
Poisson12 = 0.157; % Poisson's Ratio in the 1-2
t = 0.0001778; % Ply Thickness (m)
```

```
Poisson21 = Poisson12*E_2/E_1; % Poisson's Ratio in the 2-1
height = 0.0028448 ; % total height in (m)
```

### Calculating the Stiffness Matrix Q

```
Q11 = E_1/(1-Poisson12*Poisson21); % Stiffness in 11
Q12 = Poisson12*E_2/(1-Poisson12*Poisson21); % Stiffness in 12 and 21
Q22 = E_2/(1-Poisson12*Poisson21); % Stiffness in 22
Q66 = G_12; % Stiffness in 66
```

### Calculating Directional Stiffness matrix Qbar

```
Qbar0 = Qbar(Q11, Q12, Q22, Q66, 0);
Qbar90 = Qbar(Q11, Q12, Q22, Q66, 90);
Qbar45 = Qbar(Q11, Q12, Q22, Q66, 45);
Qbar_45 = Qbar(Q11, Q12, Q22, Q66, -45);
```

### Calculating Extensional Stiffness Aij

```
plies = height/(t*8); % number of plies for the material

A11 = plies*(Qbar_45(1,1)+Qbar45(1,1)+Qbar90(1,1)+Qbar0(1,1))*t;
A12 = plies*(Qbar_45(1,2)+Qbar45(1,2)+Qbar90(1,2)+Qbar0(1,2))*t;
A16 = plies*(Qbar_45(1,3)+Qbar45(1,3)+Qbar90(1,3)+Qbar0(1,3))*t;
A21 = plies*(Qbar_45(2,1)+Qbar45(2,1)+Qbar90(2,1)+Qbar0(2,1))*t;
A22 = plies*(Qbar_45(2,2)+Qbar45(2,2)+Qbar90(2,2)+Qbar0(2,2))*t;
A26 = plies*(Qbar_45(2,3)+Qbar45(2,3)+Qbar90(2,3)+Qbar0(2,3))*t;
A61 = plies*(Qbar_45(3,1)+Qbar45(3,1)+Qbar90(3,1)+Qbar0(3,1))*t;
A62 = plies*(Qbar_45(3,2)+Qbar45(3,2)+Qbar90(3,2)+Qbar0(3,2))*t;
A66 = plies*(Qbar_45(3,3)+Qbar45(3,3)+Qbar90(3,3)+Qbar0(3,3))*t;

A = [A11 A12 A16;A21 A22 A26;A61 A62 A66];
```

### Calculating the Inverse Extensional Stiffness [A]<sup>-1</sup>

```
Ap11 = (A22*A66-A26^2)/det(A);
Ap12 = (A16*A26-A12*A66)/det(A);
Ap22 = (A11*A66-A16^2)/det(A);
Ap16 = (A12*A26-A22*A16)/det(A);
Ap26 = (A12*A16-A11*A26)/det(A);
Ap66 = (A11*A22-A12^2)/det(A);

Ap = [Ap11 Ap12 Ap16;Ap12 Ap22 Ap26;Ap16 Ap26 Ap66];
```

*Calculating the Stress Vector*Importing Data

```
[~, ~, raw] = xlsread('R:\MQP\Strain Data.xlsx','Sheet1','B1:D1000');
```

Create output variable

```
StrainData = reshape([raw{:}],size(raw));
```

Clear temporary variables

```
clearvars raw;
```

*Calculating the Stresses in XY and 12*

```
Strain = StrainData; % Strain Vector Exx,Eyy,Gammaxy

for i= 1:1000
Stress0(i,,:,)= Qbar0*Strain(i,,:,)';
% Stress for 0 ply in xy plane
end
for i = 1:1000
Stress012(i,,:,) = stresstransformation(0)*Stress0(i,,:,)';
% Stress for 0 Ply in 12 Plane
end
for i = 1:1000
Stress45(i,,:,)= Qbar45*Strain(i,,:,)';
% Stress for 45 Ply
end
for i = 1:1000
Stress4512(i,,:,) = stresstransformation(45)*Stress45(i,,:,)';
% Stress for 45 Ply in 12 Plane
end
for i = 1:1000
Stress_45(i,,:,) = Qbar_45*Strain(i,,:,)';
% Stress for -45 ply
end
for i= 1:1000
Stress_4512(i,,:,) = stresstransformation(-45)*Stress_45(i,,:,)';
% Stress for -45 Ply in 12 Plane
end
```

```

for i = 1:1000
Stress90(i, :, :) = Qbar90*Strain(i, :,:);
% Stress for 90 ply
end
for i = 1:1000
Stress9012(i, :, :) = stresstransformation(90)*Stress90(i, :,:);
% Stress for 90 Ply in 12 Plane
end

```

### *Calculating the Strains for 12*

```

for i = 1:1000
    Strain012(i, :, :) = straintransformation(0)*Strain(i, :,:);
    % Strain for 0 Ply in 12 Plane
end
for i = 1:1000
    Strain9012(i, :, :) = straintransformation(90)*Strain(i, :,:);
    % Strain for 90 Ply in 12 Plane
end
for i = 1:1000
    Strain4512(i, :, :) = straintransformation(45)*Strain(i, :,:);
    % Strain for 45 Ply in 12 Plane
end
for i = 1:1000
    Strain_4512(i, :, :) = straintransformation(-45)*Strain(i, :,:);
    % Strain for -45 Ply in 12 Plane
end

```

### *Material Properties from Laminate Theory*

```

E_x = 1/(height*Ap11);% Young's Modulus in the x-direction
E_y = 1/(height*Ap22); % Young's Modulus in the y-direction
Poissonxy = -Ap12/Ap11; % Poisson's Ratio in the xy
Poissonyx = -Ap12/Ap22; % Poisson's Ratio in the yx
Shearxy = 1/(height*Ap66);% Shear Modulus in xy

```

### *Laminate Theory for a Given Load*

#### Import the data

```

[~, ~, raw] = xlsread('R:\MQP\Strain Data.xlsx', 'Sheet1', 'A1:A1000');

```

#### Create output variable

```
columnA = reshape([raw{:}],size(row));
```

### Clear temporary variables

```
clearvars row;
```

### *Calculating the Stresses in XY and 12 for LT*

```
Load = [columnA zeros(size(columnA,1),1) zeros(size(columnA,1),1)];% Load from Instron Data
for i = 1:1000
StrainLT(i,,:) = Ap*Load(i,,:)' ;
% Strain from Laminate Theory
end
for i = 1:1000
Stress0LT(i,,:) = Qbar0*StrainLT(i,,:)' ;
% Stress for 0 Ply from LT
end
for i = 1:1000
Stress0LT12(i,,:) = stresstransformation(0)*Stress0LT(i,,:)' ;
% Stress for 0 Ply from LT in 12 Plane
end
for i = 1:1000
Stress45LT(i,,:) = Qbar45*StrainLT(i,,:)' ;
% Stress for 45 ply from LT
end
for i = 1:1000
Stress45LT12(i,,:) = stresstransformation(45)*Stress45LT(i,,:)' ;
% Stress for 45 Ply from LT in 12 Plane
end
for i = 1:1000
Stress_45LT(i,,:) = Qbar_45*StrainLT(i,,:)' ;
% Stress for -45 ply from LT
end
for i = 1:1000
Stress_45LT12(i,,:) = stresstransformation(-45)*Stress_45LT(i,,:)' ;
% Stress for -45 Ply from LT in 12 Plane
end
for i = 1:1000
Stress90LT(i,,:) = Qbar90*StrainLT(i,,:)' ;
% Stress for 90 ply from LT
end
for i = 1:1000
Stress90LT12(i,,:) = stresstransformation(90)*Stress90LT(i,,:)' ;
% Stress for 90 Ply from LT in 12 Plane
end
```

### *Calculating the Strains for LT 12*

```

for i = 1:1000
    Strain0LT12(i,::) = straintransformation(0)*StrainLT(i,::)';
    % Strain for 0 Ply from LT in 12 Plane
end
for i = 1:1000
    Strain90LT12(i,::) = straintransformation(90)*StrainLT(i,::)';
    % Strain for 90 Ply from LT in 12 Plane
end
for i = 1:1000
    Strain45LT12(i,::) = straintransformation(45)*StrainLT(i,::)';
    % Strain for 45 Ply from LT in 12 Plane
end
for i = 1:1000
    Strain_45LT12(i,::) = straintransformation(-45)*StrainLT(i,::)';
    % Strain for -45 Ply from LT in 12 Plane
end

```

### *Plots for XY Coordinate System*

#### Stress in 0 Ply from Strain Data

```

X1 = Strain(:,1);
Y1 = Stress0(:,1);
X2 = Strain(:,2);
Y2 = Stress0(:,2);
X3 = Strain(:,3);
Y3 = Stress0(:,3);

createfigure(X1,Y1,X2,Y2,X3,Y3);
title('Stress in 0 ply v. Strain');

```

#### Stress in 0 Ply from Laminate Theory using Load

```

X4 = StrainLT(:,1);
Y4 = Stress0LT(:,1);
X5 = StrainLT(:,2);
Y5 = Stress0LT(:,2);
X6 = StrainLT(:,3);
Y6 = Stress0LT(:,3);

createfigure(X4,Y4,X5,Y5,X6,Y6);
title('Stress in 0 ply from Laminate Theory v. Strain');

```



### Stress in 45 Ply from Strain Data

```
x7 = Strain(:,1);
y7 = Stress45(:,1);
x8 = Strain(:,2);
y8 = Stress45(:,2);
x9 = Strain(:,3);
y9 = Stress45(:,3);

createfigure(x7,y7,x8,y8,x9,y9);
title('Stress in 45 ply v. Strain');
```

### Stress in 45 Ply from Laminate Theory using Load

```
x10 = StrainLT(:,1);
y10 = Stress45LT(:,1);
x11 = StrainLT(:,2);
y11 = Stress45LT(:,2);
x12 = StrainLT(:,3);
y12 = Stress45LT(:,3);

createfigure(x10,y10,x11,y11,x12,y12);
title('Stress in 45 ply from Laminate Theory v. Strain');
```

### Stress in -45 Ply from Strain Data

```
x13 = Strain(:,1);
y13 = Stress_45(:,1);
x14 = Strain(:,2);
y14 = Stress_45(:,2);
x15 = Strain(:,3);
y15 = Stress_45(:,3);

createfigure(x13,y13,x14,y14,x15,y15);
title('Stress in -45 ply v. Strain');
```

### Stress in -45 Ply from Laminate Theory using Load

```
x16 = StrainLT(:,1);
y16 = Stress_45LT(:,1);
x17 = StrainLT(:,2);
y17 = Stress_45LT(:,2);
x18 = StrainLT(:,3);
y18 = Stress_45LT(:,3);
```

```
createfigure(X16,Y16,X17,Y17,X18,Y18);
title('Stress in -45 ply from Laminate Theory v. Strain');
```

### Stress in 90 Ply from Strain Data

```
X19 = Strain(:,1);
Y19 = Stress90(:,1);
X20 = Strain(:,2);
Y20 = Stress90(:,2);
X21 = Strain(:,3);
Y21 = Stress90(:,3);

createfigure(X19,Y19,X20,Y20,X21,Y21);
title('Stress in 90 ply v. Strain');
```

### Stress in 90 Ply from Laminate Theory using Load

```
X22 = StrainLT(:,1);
Y22 = Stress90LT(:,1);
X23 = StrainLT(:,2);
Y23 = Stress90LT(:,2);
X24 = StrainLT(:,3);
Y24 = Stress90LT(:,3);

createfigure(X22,Y22,X23,Y23,X24,Y24);
title('Stress in 90 ply from Laminate Theory v. Strain')
```

### *Plots for 12 coordinate system*

### Stress in 0 Ply from Strain Data

```
X25 = Strain012(:,1);
Y25 = Stress012(:,1);
X26 = Strain012(:,2);
Y26 = Stress012(:,2);
X27 = Strain012(:,3);
Y27 = Stress012(:,3);

createfigure12(X25,Y25,X26,Y26,X27,Y27);
title('Stress in 0 ply in the 12 plane v. Strain');
```

Stress in 0 Ply from Laminate Theory using Load

```

x28 = Strain0LT12(:,1);
y28 = Stress0LT12(:,1);
x29 = Strain0LT12(:,2);
y29 = Stress0LT12(:,2);
x30 = Strain0LT12(:,3);
y30 = Stress0LT12(:,3);

createfigure12(x28,y28,x29,y29,x30,y30);
title('Stress in 0 ply from Laminate Theory in the 12 Plane v. Strain');

```

Stress in 45 Ply from Strain Data

```

x31 = Strain4512(:,1);
y31 = Stress4512(:,1);
x32 = Strain4512(:,2);
y32 = Stress4512(:,2);
x33 = Strain4512(:,3);
y33 = Stress4512(:,3);

createfigure12(x31,y31,x32,y32,x33,y33);
title('Stress in 45 ply in the 12 Plane v. Strain');

```

Stress in 45 Ply from Laminate Theory using Load

```

x34 = Strain45LT12(:,1);
y34 = Stress45LT12(:,1);
x35 = Strain45LT12(:,2);
y35 = Stress45LT12(:,2);
x36 = Strain45LT12(:,3);
y36 = Stress45LT12(:,3);

createfigure12(x34,y34,x35,y35,x36,y36);
title('Stress in 45 ply from Laminate Theory in the 12 Plane v. Strain');

```

Stress in -45 Ply from Strain Data

```

x37 = Strain_4512(:,1);
y37 = Stress_4512(:,1);
x38 = Strain_4512(:,2);
y38 = Stress_4512(:,2);
x39 = Strain_4512(:,3);
y39 = Stress_4512(:,3);

```

```
createfigure12(x37,y37,x38,y38,x39,y39);  
title('Stress in -45 ply in the 12 plane v. Strain');
```

### Stress in -45 Ply from Laminate Theory using Load

```
x40 = Strain_45LT12(:,1);  
y40 = Stress_45LT12(:,1);  
x41 = Strain_45LT12(:,2);  
y41 = Stress_45LT12(:,2);  
x42 = Strain_45LT12(:,3);  
y42 = Stress_45LT12(:,3);  
  
createfigure12(x40,y40,x41,y41,x42,y42);  
title('Stress in -45 ply from Laminate Theory in the 12 plane v. Strain');
```

### Stress in 90 Ply from Strain Data

```
x43 = Strain9012(:,1);  
y43 = Stress9012(:,1);  
x44 = Strain9012(:,2);  
y44 = Stress9012(:,2);  
x45 = Strain9012(:,3);  
y45 = Stress9012(:,3);  
  
createfigure12(x43,y43,x44,y44,x45,y45);  
title('Stress in 90 ply in 12 Plane v. Strain');
```

### Stress in 90 Ply from Laminate Theory using Load

```
x46 = Strain90LT12(:,1);  
y46 = Stress90LT12(:,1);  
x47 = Strain90LT12(:,2);  
y47 = Stress90LT12(:,2);  
x48 = Strain90LT12(:,3);  
y48 = Stress90LT12(:,3);  
  
createfigure12(x46,y46,x47,y47,x48,y48);  
title('Stress in 90 ply from Laminate Theory in 12 Plane v. Strain');
```

### *Bland Altman Plots for Strain*

The purpose of these plots is to compare the correlation between the two data sets from Laminate Theory and Data Acquisition for Stresses and Strains experienced

```

strainBA((X4+X1)/2, X4-X1, (X5+X2)/2, X5-X2, (X6+X3)/2, X6-X3)
title('Strain Bland-Altman Plot for 0 Degree Raster')

strainBA((X10+X7)/2, X10-X7, (X11+X8)/2, X11-X8, (X12+X9)/2, X12-X9)
title('Strain Bland-Altman Plot for 45 Degree Raster')

strainBA((X16+X13)/2, X16-X13, (X17+X14)/2, X17-X14, (X18+X15)/2, X18-X15)
title('Strain Bland-Altman Plot for -45 Degree Raster')

strainBA((X22+X19)/2, X22-X19, (X23+X20)/2, X23-X20, (X24+X21)/2, X24-X21)
title('Strain Bland-Altman Plot for 90 Degree Raster')

strainBA12((X28+X25)/2, X28-X25, (X29+X26)/2, X29-X26, (X30+X27)/2, X30-X27)
title('Strain Bland-Altman Plot for 0 Degree Raster 12 Coordinate')

strainBA12((X34+X31)/2, X34-X31, (X35+X32)/2, X35-X32, (X36+X33)/2, X36-X33)
title('Strain Bland-Altman Plot for 45 Degree Raster 12 Coordinate')

strainBA12((X40+X37)/2, X40-X37, (X41+X38)/2, X41-X38, (X42+X39)/2, X42-X39)
title('Strain Bland-Altman Plot for -45 Degree Raster 12 Coordinate')

strainBA12((X46+X43)/2, X46-X43, (X47+X44)/2, X47-X44, (X48+X45)/2, X48-X45)
title('Strain Bland-Altman Plot for 90 Degree Raster 12 Coordinate')

```

### *Bland-Altman Plots for Stress*

```

stressBA((Y4+Y1)/2, Y4-Y1, (Y5+Y2)/2, Y5-Y2, (Y6+Y3)/2, Y6-Y3)
title('Stress Bland-Altman Plot for 0 Degree Raster')

stressBA((Y10+Y7)/2, Y10-Y7, (Y11+Y8)/2, Y11-Y8, (Y12+Y9)/2, Y12-Y9)
title('Stress Bland-Altman Plot for 45 Degree Raster')

stressBA((Y16+Y13)/2, Y16-Y13, (Y17+Y14)/2, Y17-Y14, (Y18+Y15)/2, Y18-Y15)
title('Stress Bland-Altman Plot for -45 Degree Raster')

stressBA((Y22+Y19)/2, Y22-Y19, (Y23+Y20)/2, Y23-Y20, (Y24+Y21)/2, Y24-Y21)
title('Stress Bland-Altman Plot for 90 Degree Raster')

stressBA12((Y28+Y25)/2, Y28-Y25, (Y29+Y26)/2, Y29-Y26, (Y30+Y27)/2, Y30-Y27)
title('Stress Bland-Altman Plot for 0 Degree Raster 12 Coordinate')

stressBA12((Y34+Y31)/2, Y34-Y31, (Y35+Y32)/2, Y35-Y32, (Y36+Y33)/2, Y36-Y33)
title('Stress Bland-Altman Plot for 45 Degree Raster 12 Coordinate')

stressBA12((Y40+Y37)/2, Y40-Y37, (Y41+Y38)/2, Y41-Y38, (Y42+Y39)/2, Y42-Y39)

```

```

title('Stress Bland-Altman Plot for -45 Degree Raster 12 Coordinate')

stressBA12((Y46+Y43)/2, Y46-Y43, (Y47+Y44)/2, Y47-Y44, (Y48+Y45)/2, Y48-Y45)
title('Stress Bland-Altman Plot for 90 Degree Raster 12 Coordinate')

```

## 6.4.16 Determining the stress and strain relationships in a 0/90 Ply Laminate

### *Constants*

```

E_1 = 1904e6; % Young's Modulus in the 1-direction (Pa)
E_2 = 2228e6; % Young's Modulus in the 2-direction (Pa)
G_12 = 896.9e6; % Shear Modulus in the 1-2 (Pa)
Poisson12 = 0.157; % Poisson's Ratio in the 1-2
t = 0.0001778; % Ply Thickness (m)
Poisson21 = Poisson12*E_2/E_1; % Poisson's Ratio in the 2-1
height = 0.0033782 ; % total height in (m)

```

### *Calculating the Stiffness Matrix Q*

```

Q11 = E_1/(1-Poisson12*Poisson21); % Stiffness in 11
Q12 = Poisson12*E_2/(1-Poisson12*Poisson21); % Stiffness in 12 and 21
Q22 = E_2/(1-Poisson12*Poisson21); % Stiffness in 22
Q66 = G_12; % Stiffness in 66

```

### *Calculating Directional Stiffness matrix Qbar*

```

Qbar0 = Qbar(Q11, Q12, Q22, Q66, 0);
Qbar90 = Qbar(Q11, Q12, Q22, Q66, 90);

```

### *Calculating Extensional Stiffness Aij*

```

plies = height/(t*8); % number of plies for the material

A11 = plies*(Qbar90(1,1)+Qbar0(1,1))*t;
A12 = plies*(Qbar90(1,2)+Qbar0(1,2))*t;
A16 = plies*(Qbar90(1,3)+Qbar0(1,3))*t;
A21 = plies*(Qbar90(2,1)+Qbar0(2,1))*t;
A22 = plies*(Qbar90(2,2)+Qbar0(2,2))*t;
A26 = plies*(Qbar90(2,3)+Qbar0(2,3))*t;
A61 = plies*(Qbar90(3,1)+Qbar0(3,1))*t;

```

```
A62 = plies*(Qbar90(3,2)+Qbar0(3,2))*t;
A66 = plies*(Qbar90(3,3)+Qbar0(3,3))*t;

A = [A11 A12 A16;A21 A22 A26;A61 A62 A66];
```

### *Calculating the Inverse Extensional Stiffness $[A]^{-1}$*

```
Ap11 = (A22*A66-A26^2)/det(A);
Ap12 = (A16*A26-A12*A66)/det(A);
Ap22 = (A11*A66-A16^2)/det(A);
Ap16 = (A12*A26-A22*A16)/det(A);
Ap26 = (A12*A16-A11*A26)/det(A);
Ap66 = (A11*A22-A12^2)/det(A);

Ap = [Ap11 Ap12 Ap16;Ap12 Ap22 Ap26;Ap16 Ap26 Ap66];
```

### *Calculating the Stress Vector*

```
% Import the Data
[~, ~, raw] = xlsread('R:\MQP\Strain Data.xlsx','Sheet1','B1:D1000');

% Create output variable
StrainData = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw;

Strain = StrainData; % Strain Vector Exx,Eyy,Gammaxy

for i= 1:1000
Stress0(i,,:) = Qbar0*Strain(i,,:);
% Stress for 0 ply in xy plane
end
for i = 1:1000
Stress012(i,,:) = stresstransformation(0)*Stress0(i,,:);
% Stress for 0 Ply in 12 Plane
end
for i = 1:1000
Stress90(i,,:) = Qbar90*Strain(i,,:);
% Stress for 90 ply
end
```

```

for i = 1:1000
Stress9012(i, :, :) = stresstransformation(90)*Stress90(i, :,:);
% Stress for 90 Ply in 12 Plane
end

```

### *Calculating the Strains for 12*

```

for i = 1:1000
    Strain012(i, :, :) = straintransformation(0)*Strain(i, :,:); % Strain for 0 Ply in 12 plane
end
for i = 1:1000
    Strain9012(i, :, :) = straintransformation(90)*Strain(i, :,:); % Strain for 90 Ply in 12 plane
end

```

### *Material Properties from Laminate Theory*

```

E_x = 1/(height*Ap11); % Young's Modulus in the x-direction
E_y = 1/(height*Ap22); % Young's Modulus in the y-direction
Poissonxy = -Ap12/Ap11; % Poisson's Ratio in the xy
Poissonyx = -Ap12/Ap22; % Poisson's Ratio in the yx
Shearxy = 1/(height*Ap66); % Shear Modulus in xy

```

### *Laminate Theory for a Given Load*

#### Import the data

```
[~, ~, raw] = xlsread('R:\MQP\Strain Data.xlsx', 'Sheet1', 'A1:A1000');
```

#### Create output variable

```
columnA = reshape([raw{:}], size(raw));
```

#### Clear temporary variables

```
clearvars raw;
```

### *Calculating the Stresses in XY and 12 for LT*



```

Load = [columnA zeros(size(columnA,1),1) zeros(size(columnA,1),1)];% Load from Instron Data

for i = 1:1000
    StrainLT(i, :, :) = Ap*Load(i, :,:)' ;
    % Strain from Laminate Theory
end
for i = 1:1000
    Stress0LT(i, :, :) = Qbar0*StrainLT(i, :,:)' ;
    % Stress for 0 Ply from LT
end
for i = 1:1000
    Stress0LT12(i, :, :) = stresstransformation(0)*Stress0LT(i, :,:)' ;
    % Stress for 0 Ply from LT in 12 Plane
end
for i = 1:1000
    Stress90LT(i, :, :) = Qbar90*StrainLT(i, :,:)' ;
    % Stress for 90 ply from LT
end
for i = 1:1000
    Stress90LT12(i, :, :) = stresstransformation(90)*Stress90LT(i, :,:)' ;
    % Stress for 90 Ply from LT in 12 Plane
end

```

### *Calculating the Strains for LT 12*

```

for i = 1:1000
    Strain0LT12(i, :, :) = straintransformation(0)*StrainLT(i, :,:)' ;
    % Strain for 0 Ply from LT in 12 Plane
end
for i = 1:1000
    Strain90LT12(i, :, :) = straintransformation(90)*StrainLT(i, :,:)' ;
    % Strain for 90 Ply from LT in 12 Plane
end

```

### *Plots for XY Coordinate System*

#### Stress in 0 Ply from Strain Data

```

X1 = Strain(:,1);
Y1 = Stress0(:,1);
X2 = Strain(:,2);
Y2 = Stress0(:,2);

```

```
x3 = Strain(:,3);
Y3 = Stress0(:,3);

createfigure(x1,y1,x2,y2,x3,y3);
title('Stress in 0 ply v. Strain');
```

### Stress in 0 Ply from Laminate Theory using Load

```
x4 = StrainLT(:,1);
Y4 = Stress0LT(:,1);
x5 = StrainLT(:,2);
Y5 = Stress0LT(:,2);
x6 = StrainLT(:,3);
Y6 = Stress0LT(:,3);

createfigure(x4,y4,x5,y5,x6,y6);
title('Stress in 0 ply from Laminate Theory v. Strain');
```

### Stress in 90 Ply from Strain Data

```
x19 = Strain(:,1);
Y19 = Stress90(:,1);
x20 = Strain(:,2);
Y20 = Stress90(:,2);
x21 = Strain(:,3);
Y21 = Stress90(:,3);

createfigure(x19,y19,x20,y20,x21,y21);
title('Stress in 90 ply v. Strain');
```

### Stress in 90 Ply from Laminate Theory using Load

```
x22 = StrainLT(:,1);
Y22 = Stress90LT(:,1);
x23 = StrainLT(:,2);
Y23 = Stress90LT(:,2);
x24 = StrainLT(:,3);
Y24 = Stress90LT(:,3);

createfigure(x22,y22,x23,y23,x24,y24);
title('Stress in 90 ply from Laminate Theory v. Strain');
```

## *Plots for 12 coordinate system*

### Stress in 0 Ply from Strain Data

```
x25 = Strain012(:,1);
Y25 = Stress012(:,1);
x26 = Strain012(:,2);
Y26 = Stress012(:,2);
x27 = Strain012(:,3);
Y27 = Stress012(:,3);

createfigure12(x25,Y25,x26,Y26,x27,Y27);
title('Stress in 0 ply in the 12 plane v. Strain');
```

### Stress in 0 Ply from Laminate Theory using Load

```
x28 = Strain0LT12(:,1);
Y28 = Stress0LT12(:,1);
x29 = Strain0LT12(:,2);
Y29 = Stress0LT12(:,2);
x30 = Strain0LT12(:,3);
Y30 = Stress0LT12(:,3);

createfigure12(x28,Y28,x29,Y29,x30,Y30);
title('Stress in 0 ply from Laminate Theory in the 12 Plane v. Strain');
```

### Stress in 90 Ply from Strain Data

```
x43 = Strain9012(:,1);
Y43 = Stress9012(:,1);
x44 = Strain9012(:,2);
Y44 = Stress9012(:,2);
x45 = Strain9012(:,3);
Y45 = Stress9012(:,3);

createfigure12(x43,Y43,x44,Y44,x45,Y45);
title('Stress in 90 ply in 12 Plane v. Strain');
```

### Stress in 90 Ply from Laminate Theory using Load

```
x46 = Strain90LT12(:,1);
Y46 = Stress90LT12(:,1);
x47 = Strain90LT12(:,2);
Y47 = Stress90LT12(:,2);
```

```

x48 = Strain90LT12(:,3);
y48 = Stress90LT12(:,3);

createfigure12(x46,y46,x47,y47,x48,y48);
title('Stress in 90 ply from Laminate Theory in 12 Plane v. Strain');

```

### *Bland Altman Plots for Strain*

The purpose of these plots is to compare the correlation between the two data sets from Laminate Theory and Data Acquisition for Stresses and Strains experienced

```

strainBA((x4+x1)/2, x4-x1, (x5+x2)/2, x5-x2, (x6+x3)/2, x6-x3)
title('Strain Bland-Altman Plot for 0 Degree Raster')

strainBA((x22+x19)/2, x22-x19, (x23+x20)/2, x23-x20, (x24+x21)/2, x24-x21)
title('Strain Bland-Altman Plot for 90 Degree Raster')

strainBA12((x28+x25)/2, x28-x25, (x29+x26)/2, x29-x26, (x30+x27)/2, x30-x27)
title('Strain Bland-Altman Plot for 0 Degree Raster 12 Coordinate')

strainBA12((x46+x43)/2, x46-x43, (x47+x44)/2, x47-x44, (x48+x45)/2, x48-x45)
title('Strain Bland-Altman Plot for 90 Degree Raster 12 Coordinate')

```

### *Bland-Altman Plots for Stress*

```

stressBA((y4+y1)/2, y4-y1, (y5+y2)/2, y5-x2, (y6+y3)/2, y6-y3)
title('Stress Bland-Altman Plot for 0 Degree Raster')

stressBA((y22+y19)/2, y22-y19, (y23+y20)/2, y23-y20, (y24+y21)/2, y24-y21)
title('Stress Bland-Altman Plot for 90 Degree Raster')

stressBA12((y28+y25)/2, y28-y25, (y29+y26)/2, y29-y26, (y30+y27)/2, y30-y27)
title('Stress Bland-Altman Plot for 0 Degree Raster 12 Coordinate')

stressBA12((y46+y43)/2, y46-y43, (y47+y44)/2, y47-y44, (y48+y45)/2, y48-y45)
title('Stress Bland-Altman Plot for 90 Degree Raster 12 Coordinate')

```

## 6.4.17 Determining the stress and strain relationships in a 45/-45 Laminate

### *Constants*

```

E_1 = 1904e6; % Young's Modulus in the 1-direction (Pa)
E_2 = 2228e6; % Young's Modulus in the 2-direction (Pa)
G_12 = 896.9e6; % Shear Modulus in the 1-2 (Pa)

```

```
Poisson12 = 0.157; % Poisson's Ratio in the 1-2
t = 0.0001778; % Ply Thickness (m)
Poisson21 = Poisson12*E_2/E_1; % Poisson's Ratio in the 2-1
height = 0.0033782 ; % total height in (m)
```

### *Calculating the Stiffness Matrix Q*

```
Q11 = E_1/(1-Poisson12*Poisson21); % Stiffness in 11
Q12 = Poisson12*E_2/(1-Poisson12*Poisson21); % Stiffness in 12 and 21
Q22 = E_2/(1-Poisson12*Poisson21); % Stiffness in 22
Q66 = G_12; % Stiffness in 66
```

### *Calculating Directional Stiffness matrix Qbar*

```
Qbar45 = Qbar(Q11, Q12, Q22, Q66, 45);
Qbar_45 = Qbar(Q11, Q12, Q22, Q66, -45);
```

### *Calculating Extensional Stiffness Aij*

```
plies = height/(t*8);% number of plies for the material

A11 = plies*(Qbar_45(1,1)+Qbar45(1,1))*t;
A12 = plies*(Qbar_45(1,2)+Qbar45(1,2))*t;
A16 = plies*(Qbar_45(1,3)+Qbar45(1,3))*t;
A21 = plies*(Qbar_45(2,1)+Qbar45(2,1))*t;
A22 = plies*(Qbar_45(2,2)+Qbar45(2,2))*t;
A26 = plies*(Qbar_45(2,3)+Qbar45(2,3))*t;
A61 = plies*(Qbar_45(3,1)+Qbar45(3,1))*t;
A62 = plies*(Qbar_45(3,2)+Qbar45(3,2))*t;
A66 = plies*(Qbar_45(3,3)+Qbar45(3,3))*t;

A = [A11 A12 A16;A21 A22 A26;A61 A62 A66];
```

### *Calculating the Inverse Extensional Stiffness [A]<sup>-1</sup>*

```
Ap11 = (A22*A66-A26^2)/det(A);
Ap12 = (A16*A26-A12*A66)/det(A);
Ap22 = (A11*A66-A16^2)/det(A);
Ap16 = (A12*A26-A22*A16)/det(A);
Ap26 = (A12*A16-A11*A26)/det(A);
Ap66 = (A11*A22-A12^2)/det(A);

Ap = [Ap11 Ap12 Ap16;Ap12 Ap22 Ap26;Ap16 Ap26 Ap66];
```

### Calculating the Stress Vector

```

% Import the data
[~, ~, raw] = xlsread('R:\MQP\Strain Data.xlsx','sheet1','B1:D1000');

% Create output variable
StrainData = reshape([raw{:}],size(raw));

% Clear temporary variables
clearvars raw;

Strain = StrainData; % Strain Vector Exx,Eyy,Gammaxy

for i = 1:1000
Stress45(i, :, :) = Qbar45*Strain(i, :,:);
% Stress for 45 Ply
end
for i = 1:1000
Stress4512(i, :, :) = stresstransformation(45)*Stress45(i, :,:);
% Stress for 45 Ply in 12 Plane
end
for i = 1:1000
Stress_45(i, :, :) = Qbar_45*Strain(i, :,:);
% Stress for -45 ply
end
for i= 1:1000
Stress_4512(i, :, :) = stresstransformation(-45)*Stress_45(i, :,:);
% Stress for -45 Ply in 12 Plane
end

```

### Calculating the Strains for 12

```

for i = 1:1000
Strain4512(i, :, :) = straintransformation(45)*Strain(i, :,:);
% Strain for 45 Ply in the 12 Plane
end
for i = 1:1000
Strain_4512(i, :, :) = straintransformation(-45)*Strain(i, :,:);
% Strain for -45 Ply in the 12 Plane
end

```

### Material Properties from Laminate Theory

```

E_x = 1/(height*Ap11); % Young's Modulus in the x-direction
E_y = 1/(height*Ap22); % Young's Modulus in the y-direction
Poissonxy = -Ap12/Ap11; % Poisson's Ratio in the xy

```

```
Poissonyx = -Ap12/Ap22; % Poisson's Ratio in the yx
Shearxy = 1/(height*Ap66);% Shear Modulus in xy
```

### *Laminate Theory for a Given Load*

#### Import the data

```
[~, ~, raw] = xlsread('R:\MQP\Strain Data.xlsx', 'Sheet1', 'A1:A1000');
```

#### Create output variable

```
columnA = reshape([raw{:}],size(raw));
```

#### Clear temporary variables

```
clearvars raw;
```

### *Calculating the Stresses in XY and 12 for LT*

```
Load = [columnA zeros(size(columnA,1),1) zeros(size(columnA,1),1)];% Load from Instron Data

for i = 1:1000
Stress45LT(i, :, :) = Qbar45*StrainLT(i, :,:)' ;
% Stress for 45 ply from LT
end
for i = 1:1000
Stress45LT12(i, :, :) = stresstransformation(45)*Stress45LT(i, :,:)' ;
% Stress for 45 Ply from LT in 12 Plane
end
for i = 1:1000
Stress_45LT(i, :, :) = Qbar_45*StrainLT(i, :,:)' ;
% Stress for -45 ply from LT
end
for i = 1:1000
Stress_45LT12(i, :, :) = stresstransformation(-45)*Stress_45LT(i, :,:)' ;
% Stress for -45 Ply from LT in 12 Plane
end
```

### *Calculating the Strains for LT 12*

```
for i = 1:1000
Strain45LT12(i, :, :) = straintransformation(45)*StrainLT(i, :,:)' ;
```

```

    % Strain for 45 Ply from LT in 12 Plane
end
for i = 1:1000
    Strain_45LT12(i, :, :) = straintransformation(-45)*StrainLT(i, :, :);
    % Strain for -45 Ply from LT in 12 Plane
end

```

### *Plots for XY Coordinate System*

#### Stress in 45 Ply from Strain Data

```

x7 = Strain(:,1);
y7 = Stress45(:,1);
x8 = Strain(:,2);
y8 = Stress45(:,2);
x9 = Strain(:,3);
y9 = Stress45(:,3);

createfigure(x7,y7,x8,y8,x9,y9);
title('Stress in 45 ply v. Strain');

```

#### Stress in 45 Ply from Laminate Theory using Load

```

x10 = StrainLT(:,1);
y10 = Stress45LT(:,1);
x11 = StrainLT(:,2);
y11 = Stress45LT(:,2);
x12 = StrainLT(:,3);
y12 = Stress45LT(:,3);

createfigure(x10,y10,x11,y11,x12,y12);
title('Stress in 45 ply from Laminate Theory v. Strain');

```

#### Stress in -45 Ply from Strain Data

```

x13 = Strain(:,1);
y13 = Stress_45(:,1);
x14 = Strain(:,2);
y14 = Stress_45(:,2);
x15 = Strain(:,3);
y15 = Stress_45(:,3);

createfigure(x13,y13,x14,y14,x15,y15);
title('Stress in -45 ply v. Strain');

```



Stress in -45 Ply from Laminate Theory using Load

```

X16 = StrainLT(:,1);
Y16 = Stress_45LT(:,1);
X17 = StrainLT(:,2);
Y17 = Stress_45LT(:,2);
X18 = StrainLT(:,3);
Y18 = Stress_45LT(:,3);

createfigure(X16,Y16,X17,Y17,X18,Y18);
title('Stress in -45 ply from Laminate Theory v. Strain');

```

*Plots for 12 coordinate system*Stress in 45 Ply from Strain Data

```

X31 = Strain4512(:,1);
Y31 = Stress4512(:,1);
X32 = Strain4512(:,2);
Y32 = Stress4512(:,2);
X33 = Strain4512(:,3);
Y33 = Stress4512(:,3);

createfigure12(X31,Y31,X32,Y32,X33,Y33);
title('Stress in 45 ply in the 12 Plane v. Strain');

```

Stress in 45 Ply from Laminate Theory using Load

```

X34 = Strain45LT12(:,1);
Y34 = Stress45LT12(:,1);
X35 = Strain45LT12(:,2);
Y35 = Stress45LT12(:,2);
X36 = Strain45LT12(:,3);
Y36 = Stress45LT12(:,3);

createfigure12(X34,Y34,X35,Y35,X36,Y36);
title('Stress in 45 ply from Laminate Theory in the 12 Plane v. Strain');

```

Stress in -45 Ply from Strain Data

```

X37 = Strain_4512(:,1);
Y37 = Stress_4512(:,1);
X38 = Strain_4512(:,2);
Y38 = Stress_4512(:,2);

```

```
x39 = Strain_4512(:,3);
Y39 = Stress_4512(:,3);

createfigure12(x37,y37,x38,y38,x39,y39);
title('Stress in -45 ply in the 12 plane v. Strain');
```

### Stress in -45 Ply from Laminate Theory using Load

```
x40 = Strain_45LT12(:,1);
Y40 = Stress_45LT12(:,1);
x41 = Strain_45LT12(:,2);
Y41 = Stress_45LT12(:,2);
x42 = Strain_45LT12(:,3);
Y42 = Stress_45LT12(:,3);

createfigure12(x40,y40,x41,y41,x42,y42);
title('Stress in -45 ply from Laminate Theory in the 12 plane v. Strain');
```

### *Bland Altman Plots for Strain*

The purpose of these plots is to compare the correlation between the two data sets from Laminate Theory and Data Acquisition for Stresses and Strains experienced

```
strainBA((x10+x7)/2, x10-x7, (x11+x8)/2, x11-x8, (x12+x9)/2, x12-x9)
title('Strain Bland-Altman Plot for 45 Degree Raster')

strainBA((x16+x13)/2, x16-x13, (x17+x14)/2, x17-x14, (x18+x15)/2, x18-x15)
title('Strain Bland-Altman Plot for -45 Degree Raster')

strainBA12((x34+x31)/2, x34-x31, (x35+x32)/2, x35-x32, (x36+x33)/2, x36-x33)
title('Strain Bland-Altman Plot for 45 Degree Raster 12 Coordinate')

strainBA12((x40+x37)/2, x40-x37, (x41+x38)/2, x41-x38, (x42+x39)/2, x42-x39)
title('Strain Bland-Altman Plot for -45 Degree Raster 12 Coordinate')
```

### *Bland-Altman Plots for Stress*

```
stressBA((Y10+Y7)/2, Y10-Y7, (Y11+Y8)/2, Y11-Y8, (Y12+Y9)/2, Y12-Y9)
title('Stress Bland-Altman Plot for 45 Degree Raster')

stressBA((Y16+Y13)/2, Y16-Y13, (Y17+Y14)/2, Y17-Y14, (Y18+Y15)/2, Y18-Y15)
title('Stress Bland-Altman Plot for -45 Degree Raster')

stressBA12((Y34+Y31)/2, Y34-Y31, (Y35+Y32)/2, Y35-Y32, (Y36+Y33)/2, Y36-Y33)
title('Stress Bland-Altman Plot for 45 Degree Raster 12 Coordinate')
```

```
stressBA12((Y40+Y37)/2, Y40-Y37, (Y41+Y38)/2, Y41-Y38, (Y42+Y39)/2, Y42-Y39)
title('Stress Bland-Altman Plot for -45 Degree Raster 12 Coordinate')
```

## 6.4.18 Sub-functions for Stress and Strain Analysis

### *Reduced Stiffness Matrix with Transformation*

```
function y = Qbar(Q11, Q12, Q22, Q66, theta)
% This function calculates Qbar
% Q11 is the stiffness in 11
% Q12 is the stiffness in 12
% Q22 is the stiffness in 22
% Q66 is the stiffness in 66
% Theta is the Raster angle in degrees
Qbar11 = Q11*cosd(theta).^4+2*(Q12+2*Q66)*sind(theta).^2*cosd(theta).^2+Q22*sind(theta).^4;
Qbar12 = (Q11+Q22-4*Q66)*sind(theta).^2*cosd(theta).^2+Q12*(sind(theta).^4+cosd(theta).^4);
Qbar21 = Qbar12;
Qbar22 = Q11*sind(theta).^4 +2*(Q12+2*Q66)*sind(theta).^2*cosd(theta).^2+Q22*cosd(theta).^4;
Qbar16 = (Q11-Q12-2*Q66)*sind(theta)*cosd(theta).^3+(Q12-Q22+2*Q66)*sind(theta).^3*cosd(theta);
Qbar61 = Qbar16;
Qbar26 = (Q11-Q12-2*Q66)*sind(theta).^3*cosd(theta)+(Q12-Q22+2*Q66)*sind(theta)*cosd(theta).^3;
Qbar62 = Qbar26;
Qbar66 = (Q11+Q22-2*Q12-2*Q66)*sind(theta).^2*cosd(theta).^2+Q66*(sind(theta).^4+cosd(theta).^4);

y = [Qbar11 Qbar12 Qbar16; Qbar21 Qbar22 Qbar26; Qbar61 Qbar62 Qbar66];
```

### *Strain Transformation Matrix*

```
function y = straintransformation(theta)
% This function transforms the strain data from xy to 12
% theta is the angle of the ply in degrees

Tstrain = [cosd(theta).^2 sind(theta).^2 sind(theta)*cosd(theta);...
           sind(theta).^2 cosd(theta).^2 -sind(theta)*cosd(theta); ...
           -2*sind(theta)*cosd(theta) 2*sind(theta)*cosd(theta) cosd(theta).^2-sind(theta).^2];

y = Tstrain;
```

### *Stress Transformation Matrix*

```
function y = stresstransformation(theta)
% This function transforms the stress vector from xy to 12
% theta is the raster angle in degrees

Tstress = [cosd(theta).^2 sind(theta).^2 2*sind(theta)*cosd(theta);...
           sind(theta).^2 cosd(theta).^2 -2*sind(theta)*cosd(theta);...
           -sind(theta)*cosd(theta) sind(theta)*cosd(theta) cosd(theta).^2-sind(theta).^2];

y = Tstress;
```

## 6.4.19 Plotting Functions used for Stress and Strain Analysis

### *Stress Strain Plot in XY Plane*

```
function createfigure(X1, Y1, X2, Y2, X3, Y3)
%CREATEFIGURE(X1,Y1,X2,Y2,X3,Y3)
% X1: Strain xx
% Y1: Stress xx
% X2: Strain yy
```

```

% Y2: Stress yy
% X3: Strain xy
% Y3: Stress xy

% Auto-generated by MATLAB on 20-Sep-2013 11:04:31
% Purpose is for Stress Strain plot in xy plane

% Create figure
figure1 = figure;

% Create axes
axes1 = axes('Parent',figure1);
box(axes1,'on');
hold(axes1,'all');

% Create plot
plot(X1,Y1,'Parent',axes1,'DisplayName','xx');

% Create plot
plot(X2,Y2,'Parent',axes1,'DisplayName','yy');

% Create plot
plot(X3,Y3,'Parent',axes1,'DisplayName','xy');

% Create xlabel
xlabel('Strain(\mu strain)');

% Create ylabel
ylabel('Stress (Pa)');

% Create legend
Legend(axes1,'show');

```

### *Stress Strain Plot in 12 Plane*

```

function createfigure12(X1, Y1, X2, Y2, X3, Y3)
%CREATEFIGURE12(X1, Y1, X2, Y2, X3, Y3)
% X1: Strain 11
% Y1: Stress 11
% X2: Strain 22
% Y2: Stress 22
% X3: Strain 12
% Y3: Stress 12

% Auto-generated by MATLAB on 22-Sep-2013 22:29:36
% Purpose is for Stress Strain plot for 12 Plane

% Create figure
figure1 = figure;

% Create axes
axes1 = axes('Parent',figure1);
box(axes1,'on');
hold(axes1,'all');

% Create plot
plot(X1,Y1,'Parent',axes1,'DisplayName','11');

% Create plot
plot(X2,Y2,'Parent',axes1,'DisplayName','22');

% Create plot
plot(X3,Y3,'Parent',axes1,'DisplayName','12');

% Create xlabel
xlabel('Strain(\mu strain)');

% Create ylabel
ylabel('Stress (Pa)');

% Create legend
Legend(axes1,'show');

```

*Strain Bland-Altman Plot for XY Plane*

```

function strainBA(X1, Y1, X2, Y2, X3, Y3)
%strainBA(X1, Y1, X2, Y2, X3, Y3)
% X1: Strain Average xx
% Y1: Strain Difference xx
% X2: Strain Average yy
% Y2: Strain Difference yy
% X3: Strain Average xy
% Y3: Strain Difference xy

% Auto-generated by MATLAB on 22-Sep-2013 21:58:25
% The purpose of this function is to create a Bland Altman Plot for Strain
% in the xy plane

% Create figure
figure1 = figure;

% Create axes
axes1 = axes('Parent',figure1);
box(axes1,'on');
hold(axes1,'all');

% Create plot
plot(X1,Y1,'Parent',axes1,'DisplayName','xx');

% Create plot
plot(X2,Y2,'Parent',axes1,'DisplayName','yy');

% Create plot
plot(X3,Y3,'Parent',axes1,'DisplayName','xy');

% Create xlabel
xlabel('Average Strain (\mu strain)');

% Create ylabel
ylabel('Difference between Laminate Theory and Data (\mu strain)');

% Create legend
legend(axes1,'show');

```

*Strain Bland-Altman Plot for 12 Plane*

```

function strainBA12(X1, Y1, X2, Y2, X3, Y3)
%strainBA12(X1, Y1, X2, Y2, X3, Y3)
% X1: Strain Average 11
% Y1: Strain Difference 11
% X2: Strain Average 22
% Y2: Strain Difference 22
% X3: Strain Average 12
% Y3: Strain Difference 12

% Auto-generated by MATLAB on 22-Sep-2013 22:06:21
% The purpose of this function is to create a Bland Altman Plot for Strain
% in the 12 plane

% Create figure
figure1 = figure;

% Create axes
axes1 = axes('Parent',figure1);
box(axes1,'on');
hold(axes1,'all');

% Create plot
plot(X1,Y1,'Parent',axes1,'DisplayName','11');

% Create plot
plot(X2,Y2,'Parent',axes1,'DisplayName','22');

```

```

% Create plot
plot(X3,Y3,'Parent',axes1,'DisplayName','12');

% Create xlabel
xlabel('Average Strain (\mu strain)');

% Create ylabel
ylabel('Difference between Laminate Theory and Data (\mu strain)');

% Create legend
legend(axes1,'show');

```

### *Stress Bland-Altman Plot for XY Plane*

```

function stressBA(X1, Y1, X2, Y2, X3, Y3)
%stressBA(X1, Y1, X2, Y2, X3, Y3)
% X1: Stress Average xx
% Y1: Stress Difference xx
% X2: Stress Average yy
% Y2: Stress Difference yy
% X3: Stress Average xy
% Y3: Stress Difference xy

% Auto-generated by MATLAB on 22-Sep-2013 22:15:24
% The purpose of this function is to create a Bland Altman Plot for Stress
% in the xy plane

% Create figure
figure1 = figure;

% Create axes
axes1 = axes('Parent',figure1);
box(axes1,'on');
hold(axes1,'all');

% Create plot
plot(X1,Y1,'Parent',axes1,'DisplayName','xx');

% Create plot
plot(X2,Y2,'Parent',axes1,'DisplayName','yy');

% Create plot
plot(X3,Y3,'Parent',axes1,'DisplayName','xy');

% Create xlabel
xlabel('Average Stress(Pa)');

% Create ylabel
ylabel('Difference between Laminate Theory and Data (Pa)');

% Create legend
legend(axes1,'show');

```

### *Stress Bland Altman Plot for 12 Plane*

```

function stressBA12(X1, Y1, X2, Y2, X3, Y3)
%stressBA12(X1, Y1, X2, Y2, X3, Y3)
% X1: Stress Average 11
% Y1: Stress Difference 11
% X2: Stress Average 22
% Y2: Stress Difference 22

```

```

% X3: Stress Average 12
% Y3: Stress Difference 12

% Auto-generated by MATLAB on 22-Sep-2013 22:19:50
% The purpose of this function is to create a Bland Altman Plot for Stress
% in the 12 plane

% Create figure
figure1 = figure;

% Create axes
axes1 = axes('Parent',figure1);
box(axes1,'on');
hold(axes1,'all');

% Create plot
plot(X1,Y1,'Parent',axes1,'DisplayName','11');

% Create plot
plot(X2,Y2,'Parent',axes1,'DisplayName','22');

% Create plot
plot(X3,Y3,'Parent',axes1,'DisplayName','12');

% Create xlabel
xlabel('Average Stress(Pa)');

% Create ylabel
ylabel('Difference between Laminate Theory and Data (Pa)');

% Create legend
Legend(axes1,'show');

```

## 6.4.20 To determine the Coefficient of Thermal Expansion for Orthotropic Laminates, it uses Mathematics presented in di Scalea

### *Constants*

```

Kt = .5;% Transverse Sensitivity
Poisson = 0.28;% Poisson's Ratio for material used by gage manufacturer for calibration
CTEref = 23.5;% Coefficient of Thermal Expansion for reference Material(micrometers/m*C)

```

### *Importing Data from Spreadsheet*

#### Import the data (Temperature)

```
[~, ~, raw] = xlsread('R:\MQP\ThermalExpansionData.xlsx','Sheet1','A2:A142');
```

#### Create output variable

```
Temperature = reshape([raw{:}],size(raw));
```

### Clear temporary variables

```
clearvars raw;
```

### Import the data (Strain xx)

```
[~, ~, raw] = xlsread('R:\MQP\ThermalExpansionData.xlsx', 'Sheet1', 'B2:B142');
```

### Create output variable

```
strainx = reshape([raw{:}], size(raw));
```

### Clear temporary variables

```
clearvars raw;
```

### Import the data (Strain yy)

```
[~, ~, raw] = xlsread('R:\MQP\ThermalExpansionData.xlsx', 'Sheet1', 'C2:C142');
```

### Create output variable

```
strainy = reshape([raw{:}], size(raw));
```

### Clear temporary variables

```
clearvars raw;
```

### Import the data (Strain ref)

```
[~, ~, raw] = xlsread('R:\MQP\ThermalExpansionData.xlsx', 'Sheet1', 'D2:D142');
```



Create output variable

```
Strainref = reshape([raw{:}],size(raw));
```

Clear temporary variables

```
clearvars raw;
```

*Calculating the CTE*

```
for i = 1:140
    CTEx(i) = CTEref + (1/(Temperature(i+1)-Temperature(i)))*((1-Poisson*Kt)/...
        (1-Kt^2))*((Strainx(i)-Strainy(i)*Kt)-(1-Kt)*Strainref(i));
end

for i = 1:140
    CTEy(i) = CTEref+(1/(Temperature(i+1)-Temperature(i)))*((1-Poisson*Kt)/...
        (1-Kt^2))*((Strainy(i)-Strainx(i)*Kt)-(1-Kt)*Strainref(i));
end
```

*Plots for expansion*

```
Plotthermalstrain(Temperature, Strainx)
title('Strain in the x-direction from Thermal Loading')

Plotthermalstrain(Temperature, Strainy)
title('Strain in the y-direction from Thermal Loading')

Plotthermalstrain(Temperature, Strainref)
title('Strain for the Reference Material from Thermal Loading')

Plotthermalexpansion(Temperature(1:140), [CTEx; CTEy])
```

## 6.4.21 Plotting Functions used for Heat Expansion

### *Thermal Strain Plot*

```
function Plotthermalstrain(X1, Y1)
%Plotthermalstrain(X1, Y1)
% X1: Temperature
% Y1: Vector of Strain

% Auto-generated by MATLAB on 23-Sep-2013 08:59:50

% Create figure
figure1 = figure;

% Create axes
axes1 = axes('Parent',figure1);
box(axes1,'on');
hold(axes1,'all');

% Create plot
plot(X1,Y1);

% Create xlabel
xlabel('Temperature (K)');

% Create ylabel
ylabel('Strain(\mu strain)');
```

### *Thermal Expansion Coefficient Plot*

```
function Plotthermalexpansion(X1, YMatrix1)
%Plotthermalexpansion(X1, YMATRIX1)
% X1: Temperature
% YMATRIX1: matrix of Thermal Expansion Coefficients

% Auto-generated by MATLAB on 23-Sep-2013 09:09:02

% Create figure
figure1 = figure;

% Create axes
axes1 = axes('Parent',figure1);
box(axes1,'on');
hold(axes1,'all');

% Create multiple lines using matrix input to plot
plot1 = plot(X1,YMatrix1,'Parent',axes1);
set(plot1(1),'DisplayName','CTEx');
set(plot1(2),'DisplayName','CTEy');

% Create xlabel
xlabel('Temperature(K)');

% Create ylabel
ylabel('Coefficient of thermal expansion');

% Create title
title('Thermal Expansion Coefficients over Temperature Range');

% Create legend
legend(axes1,'show');
```