

4-2019

Finite Element Model Validation and Testing of an Off-Road Vehicle under Dynamic Loading Conditions

MacKenzie Susan Cunningham

Follow this and additional works at: <https://commons.erau.edu/edt>



Part of the [Mechanical Engineering Commons](#)

Scholarly Commons Citation

Cunningham, MacKenzie Susan, "Finite Element Model Validation and Testing of an Off-Road Vehicle under Dynamic Loading Conditions" (2019). *Dissertations and Theses*. 444.
<https://commons.erau.edu/edt/444>

This Thesis - Open Access is brought to you for free and open access by Scholarly Commons. It has been accepted for inclusion in Dissertations and Theses by an authorized administrator of Scholarly Commons. For more information, please contact commons@erau.edu.

FINITE ELEMENT MODEL VALIDATION AND TESTING OF AN OFF-ROAD
VEHICLE SUSPENSION UNDER DYNAMIC LOADING CONDITIONS

by

MacKenzie Susan Cunningham

A Thesis Submitted to the College of Engineering Department of Mechanical
Engineering in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Mechanical Engineering

Embry-Riddle Aeronautical University
Daytona Beach, Florida
April 2019

FINITE ELEMENT MODEL VALIDATION AND TESTING OF AN OFF-ROAD
VEHICLE SUSPENSION UNDER DYNAMIC LOADING CONDITIONS

by

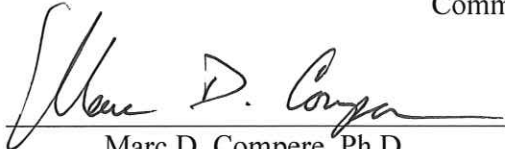
MacKenzie Susan Cunningham

This thesis was prepared under the direction of the candidate's Thesis Committee Chair, Dr. Darris L. White, Professor, Daytona Beach Campus, and Thesis Committee Members Dr. Marc D. Compere, Professor, Daytona Beach Campus, and Dr. Patrick N. Currier, Professor, Daytona Beach Campus, and has been approved by the Thesis Committee. It was submitted to the Department of Mechanical Engineering in partial fulfillment of the requirements for the degree of Master of Science in Mechanical Engineering

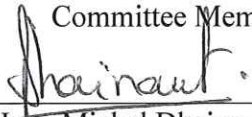
Thesis Review Committee:



Darris L. White, Ph.D.
Committee Chair



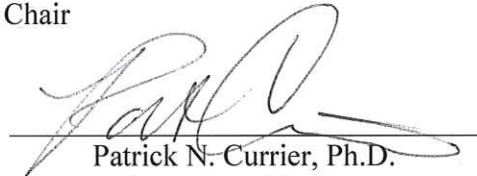
Marc D. Compere, Ph.D.
Committee Member



Jean-Michel Dhainaut, Ph.D.
Graduate Program Chair,
Mechanical Engineering



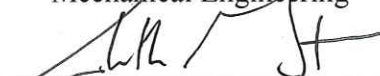
Maj Mirmirani, Ph.D.
Dean, College of Engineering



Patrick N. Currier, Ph.D.
Committee Member



Eduardo A. Divo, Ph.D.
Department Chair,
Mechanical Engineering



Christopher Grant, Ph.D.
Associate Vice President of Academics

4/10/19
Date

Acknowledgements

I am deeply grateful to my committee for all their help and guidance during my research. Dr. White has been immensely helpful in keeping me on track. Dr. Compere helped me through some difficult data acquisition troubleshooting with late nights of data collection and re-programming, for this I am deeply grateful.

I could not be where I am today without my loving and supportive family. I would particularly thank my grandmother for instilling a sense of intense determination. To my mother and father for providing me with every opportunity possible to achieve whatever I set out to accomplish. And to my loving husband for his support through the last year and constant nudging to make progress on my thesis.

Abstract

Researcher: MacKenzie Susan Cunningham

Title: Finite Element Model Validation and Testing of an Off-Road Vehicle Suspension Under Dynamic Loading Conditions

Institution: Embry-Riddle Aeronautical University

Degree: Master of Science in Mechanical Engineering

Year: 2019

In the vehicle design life cycle, validation tests consume a significant portion of the available development time. With a short, one-year design cycle, the Embry-Riddle Baja Team leverages computer-aided engineering (CAE) tools to simulate critical test cases whose loading conditions can be accurately represented by a series of static loads and constraints. Using these conditions, a Finite Element Model (FEM) can be employed to accurately predict the effects of the loading conditions in the components. During the initial design of the front suspension, one major load case was determined to be the main failure load. However, after validation testing the suspension exhibited a failure indicative of a load path not predicted. To obtain a more complete understanding of the dynamic loading conditions on the affected component, instrumentation was implemented to measure strain in the critical member. A dynamic vehicle test case was performed to measure a high-frequency, high-load case representative of an event during the vehicle service life. The measurement was then utilized to validate a Finite Element Model, in turn used to re-design the member. This component withstands the loading condition for infinite fatigue life without increasing the overall weight of the design, although the failure was not reproduced during testing.

Table of Contents

Acknowledgements.....	ii
Abstract.....	iii
Chapter I.....	1
Introduction.....	1
Chapter II.....	7
Review of the Relevant Literature.....	7
Design of Experiment.....	11
Strain Gauge.....	13
Data Analysis.....	19
Linear Regression Modeling for Calibration.....	19
Digital Filtering.....	24
Confidence Interval.....	27
Finite Element Modeling.....	29
Topology Optimization.....	32
Chapter III.....	36
Methodology.....	36
Strain Gauge Installation and Instrumentation.....	41
Data Acquisition System.....	48

Amplifier.....	49
Raspberry Pi.....	51
Calibration.....	52
Finite Element Model	57
Chapter IV.....	60
Results.....	60
Data Analysis	60
Data Filtering.	64
Zero-Offset.....	65
Strain Calculation.....	68
FEM Validation	73
Component Re-Design.....	76
Chapter V	80
Discussion, Conclusions, and Recommendations.....	80
References.....	82
Appendix A.....	89
Bibliography	89
Appendix B.....	91
Vishay Micro-Measurements: Strain Gage Installations with M-Bond 200 Adhesive	91

Appendix C	99
Vishay Micro-Measurements Application Note TT-609: Strain Gage Soldering	
Techniques	99
Appendix D.....	103
DAQ Scripts.....	103
Appendix E	111
Plotted Results	111
GPS Waypoint and Parking Chalk Location.	111
FFT Digital Filtered Results.	116
Event Window Plots.	133

List of Tables

	Page
Table 1 Strain Gauge Specification	41
Table 2 Calculated and Measured Strain Due to Load Applied to Cantilevered Beam Creating a Moment of Bending about the x-axis, M_{b_x} , at the Strain Gauge as Seen in Figure 35. Mean and Standard Deviation of 30 Measurements Presented for Each Weight.	56
Table 3 Calculated and Measured Strain Due to Load Applied to Cantilevered Beam Creating a Moment of Bending about the z-axis at the Strain Gauge as Seen in Figure 37.	56
Table 4 Calculated and Measured Strain Due to Axial Load, F_y , Applied to Component as Seen in Figure 38	56
Table 5 Descriptive Statistics of Linear Regression Model for Strain in Each Direction.	57
Table 6 Missed Data During Experimental Data Collection, Illustrated as a Percentage of the Total Data Collection with the Effect on the Event.....	61
Table 7 Identification of Missing Datapoints and Location of Event Endpoint.....	61
Table 8 GPS and Arduino (ARD) Resolution of the "time.time" Function Recorded on the Raspberry Pi.....	64
Table 9 ..ARD Vm_analog Reading Zero-Offset Value Identification Using Mean of First 100 Datapoints, Accuracy Verified with Standard Deviation	67
Table 10 Velocity of Vehicle at Event Contact Location in Meters per Second and Miles per Hour.....	72
Table 11 Maximum Strain Due to Event at each Strain Gauge Location, Type of Strain Specified, with Mean, Standard Deviation, and 95% Confidence Interval of all 9 Data Collection Events.....	73
Table 12 FEM Results, Maximum Stress in the Member and von Mises FOS, from Event Loading Condition, Scaled Loading Condition for Maximum Velocity, v_{max} , and Scaled Loading Condition for Maximum Mass and Velocity, v_{max} and W_{max}	75
Table 13 FEM Results Re-Designed Member, Maximum Stress in the Member and von Mises FOS, from Event Loading Condition, Scaled Loading Condition for Maximum Velocity, v_{max} , and Scaled Loading Condition for Maximum Mass and Velocity, v_{max} and W_{max}	78

Table 14 Resolved Force, Based on Corresponding Strain Measurement Under
Cantilevered Beam Constraints..... 79

List of Figures

	Page
Figure 1: SAE J670 Standard Axis Orientation [4].	2
Figure 2: Stress Calculation using Simple Loads (Huizinga, 2002).	8
Figure 3: Parameters of load spectrum [6].	9
Figure 4: FEM of Leaf Spring (left) and Free Body Diagram Representation of Boundary Conditions in FEM (right) [7].	10
Figure 5: Test Rig Calibration of Leaf Spring [7].	10
Figure 6: Static and Fatigue Analysis Based on FEM [7].	11
Figure 7: Definition of Strain [18].	13
Figure 8: Metallic Strain Gauge [18].	14
Figure 9: Wheatstone Bridge Circuit for Measurements Using Strain Gauges [17].	15
Figure 10: Visual representation of the effects of temperature on the reading of a strain gauge [17].	17
Figure 11: Data from Calibration of Trailer Axle [30].	22
Figure 12: Track Rod Strain-Load Calibration [31].	23
Figure 13: Linear Regression and Error Bars for Strain Output due to Torque about Y [33].	24
Figure 14: Two Continuous Sinusoidal Signals Sampled at 0.2 Sec [40].	27
Figure 15: Force Matrix (Left), Structural Matrix (center), Displacement Matrix (Right) [44].	29
Figure 16: First-Order Tetrahedral Element, Left, Second-Order Tetrahedral Element, Right [47].	31
Figure 17: Stress Concentration Test for Different Elements [48].	32
Figure 18: Hollow Tube, Inner and Outer Radii.	34
Figure 19: Square Tube, Inner and Outer Edge Length.	34
Figure 20: Right Front Lower Fore Control Arm, Deformation.	36

Figure 21: Control Arm Deformation Area, Enhanced	36
Figure 22: Concrete Block, Approximately 3.5in High.....	37
Figure 23: Logs, Approximately 15in in Diameter.....	38
Figure 24: Muddy Log, Approximately 8-12in Protruding	38
Figure 25: Large Concrete Chunk, Approximately 1 ft in Height.....	39
Figure 26: Parking Calk Width Measurement	39
Figure 27: Parking Calk Height Measurement	40
Figure 28: Strain Gauge Longitudinal Strain Measurement of a Beam in about the SAE defined z-direction in a Half Bridge	42
Figure 29: Strain Gauge Longitudinal Strain Measurement of a Beam in Bending about the SAE defined z-direction in a Half Bridge.....	44
Figure 30: Strain Gauge Set 3, Longitudinal and Transverse Strain Measurement of a Beam Subject to Bending and Axial Force.....	44
Figure 31: Component Strain Due to Initial Loading Condition Estimation.....	46
Figure 32: Lead-Wire Connected to Wire Harness.....	48
Figure 33: Wire Harness Connected to DAQ Box.....	49
Figure 34: Samples within Event.....	50
Figure 35: Strain Gauge Calibration Setup for Strain Measurement Due to Bending about the x-axis in the SAE Coordinate Frame, Component Perpendicular to the Ground	53
Figure 36: Strain Gauge Calibration Setup for Strain Measurement Due to Bending about the x-axis in the SAE Coordinate Frame, Component Perpendicular to the Ground and Non-Gauged Linkage Not Constrained.	53
Figure 37: Strain Gauge Calibration Setup for Strain Measurement Due to Bending about the z-axis in the SAE Coordinate Frame, Component Perpendicular with Calibrated Weights Applied.	54
Figure 38: Strain Gauge Calibration Setup for Axial Loading, Along Positive y-axis. ...	54
Figure 39: Weight Balance for Strain Gauge Axial Calibration Utilizing Two Turnbuckles.....	55
Figure 40: Strain Calibration in Bending about the z-axis FEM.	58

Figure 41: GPS Waypoints of Data Collection 11, Identifying the GPS Location of the Parking Chalk.....	62
Figure 42: GPS Waypoints, Data Collection 4, Overlay with Google Maps	62
Figure 43: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 1, SG1.....	65
Figure 44: Simply-Supported Beam Under Uniform Loading [56].....	69
Figure 45: Strain Measured by Each Gauge During Event Run 10 with Parking Chalk Location Estimation and Velocity.	71
Figure 46: Simply Supported Beam Constraints with one Unit Load (N) Applied in the Axial Direction (+ F_y FEM Coordinates, - F_y SAE Coordinates).	74
Figure 47: Re-Designed Suspension Lower Fore Link.....	77
Figure 48: FEM Stress Results in the Re-Designed Component Under Tested Loading Condition.....	77
Figure 49: Component FEM Under Cantilevered Beam Constraints and Loads.....	79
Figure 50: Ribbon Leads Aligned on Solder Pad with Protective Coating Removed. ...	100
Figure 51: Tinned Lead-Wire	101
Figure 52: Lead-Wires Installed on Solder Pads After Tinning	101
Figure 53: Arduino Strain Gauge and Voltage Reading Script, Part 1 of 3.....	104
Figure 54: Arduino Strain Gauge and Voltage Reading Script, Part 2 of 3.....	105
Figure 55: Arduino Strain Gauge and Voltage Reading Script, Part 3 of 3.....	106
Figure 56: Arduino Python Script, Serial Read and Write	107
Figure 57: GPS Python Script, Serial Read and Write, Part 1 of 3.....	108
Figure 58: GPS Python Script, Serial Read and Write, Part 2 of 3.....	109
Figure 59: GPS Python Script, Serial Read and Write, Part 3 of 3.....	110
Figure 60: GPS Waypoints of Data Collection 1	111
Figure 61: GPS Waypoints of Data Collection 2.....	112
Figure 62: GPS Waypoints of Data Collection 3.....	112

Figure 63: GPS Waypoints of Data Collection 4.....	113
Figure 64: GPS Waypoints of Data Collection 5.....	113
Figure 65: GPS Waypoints of Data Collection 6.....	114
Figure 66: GPS Waypoints of Data Collection 7.....	114
Figure 67: GPS Waypoints of Data Collection 8.....	115
Figure 68: GPS Waypoints of Data Collection 9.....	115
Figure 69: GPS Waypoints of Data Collection 10.....	116
Figure 70: GPS Waypoints of Data Collection 11.....	116
Figure 71: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 1, SG1.....	117
Figure 72: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 1, SG2.....	117
Figure 73: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 1, SG3.....	118
Figure 74: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 2, SG1.....	118
Figure 75: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 2, SG2.....	119
Figure 76: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 2, SG3.....	119
Figure 77: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 3, SG1.....	120
Figure 78: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 3, SG2.....	120
Figure 79: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 3, SG3.....	121
Figure 80: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 4, SG1.....	121

Figure 81: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 4, SG2.....	122
Figure 82: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 4, SG3.....	122
Figure 83: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 5, SG1.....	123
Figure 84: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 5, SG2.....	123
Figure 85: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 5, SG3.....	124
Figure 86: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 6, SG1.....	124
Figure 87: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 6, SG2.....	125
Figure 88: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 6, SG3.....	125
Figure 89: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 7, SG1.....	126
Figure 90: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 7, SG2.....	126
Figure 91: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 7, SG3.....	127
Figure 92: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 8, SG1.....	127
Figure 93: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 8, SG2.....	128
Figure 94: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 8, SG3.....	128
Figure 95: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 9, SG1.....	129
Figure 96: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 9, SG2.....	129

Figure 97: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 9, SG3.....	130
Figure 98: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 10, SG1.....	130
Figure 99: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 10, SG2.....	131
Figure 100: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 10, SG3.....	131
Figure 101: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 11, SG1.....	132
Figure 102: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 11, SG2.....	132
Figure 103: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 11, SG3.....	133
Figure 104: Strain Measured by Each Gauge During Event Run 1 with Parking Chalk Location Estimation and Velocity.	134
Figure 105: Strain Measured by Each Gauge During Event Run 2 with Parking Chalk Location Estimation and Velocity.	134
Figure 106: Strain Measured by Each Gauge During Event Run 3 with Parking Chalk Location Estimation and Velocity.	135
Figure 107: Strain Measured by Each Gauge During Event Run 4 with Parking Chalk Location Estimation and Velocity.	135
Figure 108: Strain Measured by Each Gauge During Event Run 5 with Parking Chalk Location Estimation and Velocity.	136
Figure 109: Strain Measured by Each Gauge During Event Run 6 with Parking Chalk Location Estimation and Velocity.	136
Figure 110: Strain Measured by Each Gauge During Event Run 7 with Parking Chalk Location Estimation and Velocity.	137
Figure 111: Strain Measured by Each Gauge During Event Run 8 with Parking Chalk Location Estimation and Velocity.	137
Figure 112: Strain Measured by Each Gauge During Event Run 9 with Parking Chalk Location Estimation and Velocity.	138

Figure 113: Strain Measured by Each Gauge During Event Run 10 with Parking Chalk
Location Estimation and Velocity. 138

Figure 114: Strain Measured by Each Gauge During Event Run 11 with Parking Chalk
Location Estimation and Velocity. 139

Chapter I

Introduction

In the vehicle design life cycle, validation tests consume a significant portion of the available development time [1]. Traditionally, endurance tests are performed to verify the fatigue life and structural strength of a vehicle and/or component. These tests are time intensive and require a full vehicle or physical parts; requiring testing to take place at a late stage of design. With a short, one-year design cycle, the Embry-Riddle Aeronautical University (ERAU) Women's Baja Team leverages CAE tools to simulate critical fatigue-life and structural strength related test cases. Using these conditions, an FEM can be employed to accurately predict the fatigue in the part.

Baja is a competition organized by the Society of Automotive Engineers (SAE) in which engineering students design and build an all-terrain single-seat sport utility vehicle. The vehicle must be designed to survive the server off-road terrain of the competition including specifically designed events to test different aspects of the vehicle's performance; acceleration, sled pull, land maneuverability, and suspension and traction courses [2].

During the initial design of the front suspension, one major load case was determined to be the main fatigue load case; however, after validation testing, the component exhibited a failure indicative of a load path not predicted. To obtain a more complete understanding of the dynamic loading conditions on the affected component, instrumentation was implemented to measure strain in the form of three strain gauges on the critical member. Two strain gauges were used to measure bending about the SAE coordinate system, as seen in Figure 1, in the x (Mbx) and z (Mbz) directions and a third

to measure axial strain. These three measurement devices allowed the loads in each primary direction to be understood, thus resolving the strain in the member [3].

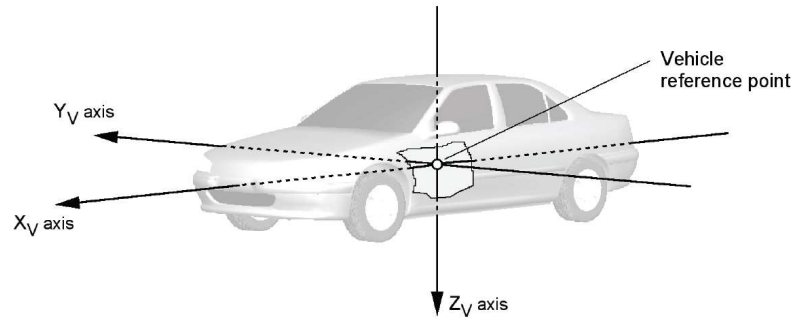


Figure 1: SAE J670 Standard Axis Orientation [4].

A dynamic vehicle test case was performed to measure a high-frequency, high-load case representative of what would be experienced during the vehicle life cycle. After statistical data analysis is performed, the repeatedly measured load case is utilized in the FEM to simulate the fatigue life of the component and the part is re-designed accordingly.

In addition to the load case studied here, the installation of the defined data acquisition system enables the ERAU Women's Baja team to collect strain data in this component during competition, gathering a strain-life history in the member. The strain-life history can then be utilized for the development of component and vehicle specific fatigue life calculation, the final major input to accelerated test development [1]. If expanded to more components, the strain-life history from competition could be utilized to perform an initial fatigue validation, through CAE, of the whole vehicle once the design phase is completed. This measured load history is an extensive undertaking and is saved for future work. In this thesis, only the defined load case was covered.

A validated FEM will aid redesign for an unexpected failure in the front suspension lower control arm. These tasks are accomplished by defining the test condition, measuring the load case under this condition, and re-designing the component to withstand the measured loading condition with in the space and weight constraints currently sustained.

Definitions of Variables

A	Cross-Sectional Area
a	Outer Width of a Square Tube
b	Inner Width of a Square Tube
E	Modulus of Elasticity
F_y	Force in the y-direction
F_x	Force in the x-direction
F_z	Force in the z-direction
f	Frequency
$gain_{ard}$	Analog Gain of Arduino Analog to Digital Converter
$gain_{amp}$	Gain of Amplifier
I	Second Area Moment of Inertia
J	Polar Moment of Inertia
k	Gauge Factor
L_{beam}	Length of Beam
M	Moment
n	Number of Samples in a Population

P	Axial Load
R^2	Coefficient of Determination
r	Radius
r_{in}	Inner Radius of a Tube
r_{out}	Outer Radius of a Tube
s	Sample Standard Deviation
s^2	Sample Variance
T	Torque Applied
t	Time
V_{ard}	Nominal Arduino Supply Voltage
V_M	Voltage Measured
V_B	Voltage Supplied
v	Velocity
v_{max}	Maximum Vehicle Velocity
W_{max}	Maximum Driver Weight
X	Independent Variable
\bar{X}	Sample Mean
x	Distance
x_i	Sample i of Population
x_1	Distance from SG1 to the Spherical Bearing Pin Location
x_3	Distance from SG3 to the Spherical Bearing Pin Location
Y	Dependent Variable
y	Distance from Center of Axis to Applied Moment or Torque

ε_{Mb_x}	Strain Due to Bending about the x-axis
ε_{Mb_z}	Strain Due to Bending about the z-axis
ε_{F_y}	Strain Due to Axial Deformation
ε	Strain
σ_{max}	Yield Stress
σ	Normal Stress
σ^2	Population Variance
τ	Shear Stress
ν	Poisson's Ratio

List of Acronyms

ADC	Analog to Digital Conversion
AE	Aerospace Engineering
ARD	Arduino
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CPU	Central Processing Unit
csv	Comma Separated Variables
DAQ	Data Acquisition
DOE	Design of Experiments
DOF	Degrees of Freedom
DOT	Department of Transportation
ERAU	Embry-Riddle Aeronautical University
FE	Finite Element

FEA	Finite Element Analysis
FEM	Finite Element Method
FFT	Fast Fourier Transform
FOS	Factor of Safety
GPS	Global Positioning System
MSE	Mean Squared Error
OEM	Original Equipment Manufacturer
RMSE	Root Mean Squared Error
SAE	Society of Automotive Engineers
SG1	Strain Gauge Set One
SG2	Strain Gauge Set Two
SG3	Strain Gauge Set Three
SSE	Sum of Squares of Errors
SST	Sum of Squares Total
TTI	Texas Transportation Institute

Chapter II

Review of the Relevant Literature

With a short, one-year design cycle, the Embry-Riddle Baja Team leverages CAE tools to simulate critical structural loading conditions and fatigue-life related test cases. With respect to fatigue, traditionally, endurance tests are performed to verify the fatigue life of a vehicle and/or component. These tests are time intensive and require a full vehicle or physical parts, requiring testing to take place at a late stage of design. To reduce this time, tests are performed under worst case conditions. Due to the load severity, the loading cycles can be reduced to achieve the same damage. Even this method takes far too long to be solely relied upon. The design engineer needs to know immediately whether the design has fatigue life problems and how to solve them. However, due to the nature of loads and mechanical behavior in a structure being complex, neither of these criteria are easily predicted [1].

However, some loading conditions can be described by representative static loads and boundary condition in a linear FEM [1]. The first phase in CAE analysis for fatigue life is a FEM where loads and constraints are applied to a CAD part to predict material stresses. For this model, loads are derived from previous measurements or calculated based on highest expected loading condition, while constraints and material properties are known. Often, test engineers will collect strain measurements under pre-defined loading conditions, loading spectrum. These results are utilized for validation of the created model. This model output, stress distribution over the component, is utilized to identify critical locations in the design and defining alternative designs. This approach is shown in Figure 2.

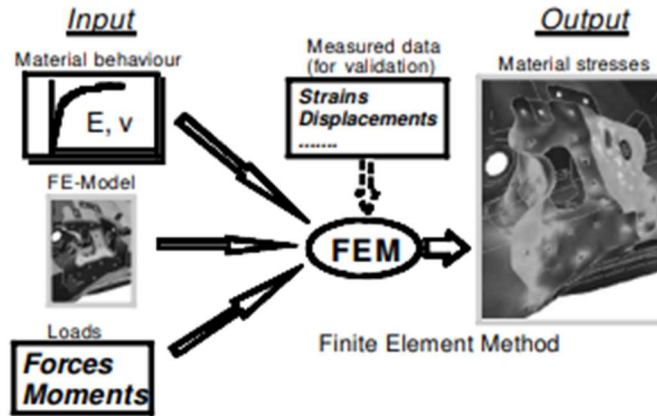


Figure 2: Stress Calculation using Simple Loads (Huizinga, 2002).

An initial design, CAD model, material properties, and constraints, is analyzed with estimated loading conditions to obtain critical stress locations in the component. These critical locations are then instrumented to obtain actual strain displacement measurements, under the specified loading spectra. The measured data is then utilized to validate the FEM, assuming constraints, CAD, and the material properties are modeled correctly the loads can be adjusted to validate the model. This validated model can then be utilized to iterate re-designs of the component.

This process is also demonstrated in Putra [5] where a static linear FEM of a coil spring with static loads and constraints was utilized to find the critical areas to locate strain gauges for the purpose of strain history data collection. First an initial FEM is conducted to locate the critical stress regions. Second the part is instrumented as close to the critical locations as possible. Finally, the vehicle underwent a defined testing case for which the strain readings were recorded and utilized to validate a CAE model of the vehicle loading conditions and boundary conditions.

Utilizing this methodology, knowledge of the loading spectrum for the component it is of critical importance [6]. The load spectrum parameters, Figure 3, depend on each other and must be critically considered to determine a representative design spectrum [7]. If usage and operating conditions are known, only the determination of vehicle related loads related are necessary for determining load spectrum.

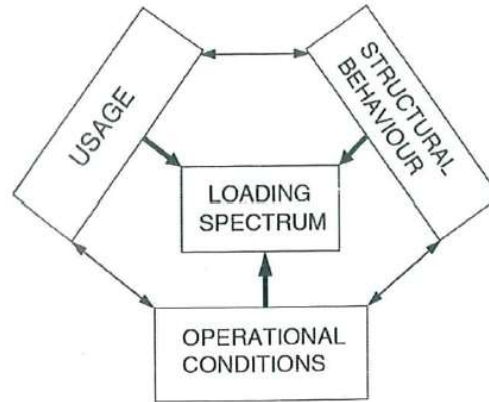


Figure 3: Parameters of load spectrum [6].

Utilizing this load spectrum Sener [7] verified the computer analysis results on a leaf spring by physically testing the part in the laboratory. First by calibrating the strain readings from the physical part in the laboratory to the FEM. The setup of FEM and laboratory test rig are illustrated in Figures Figure 4 and Figure 5 respectively.

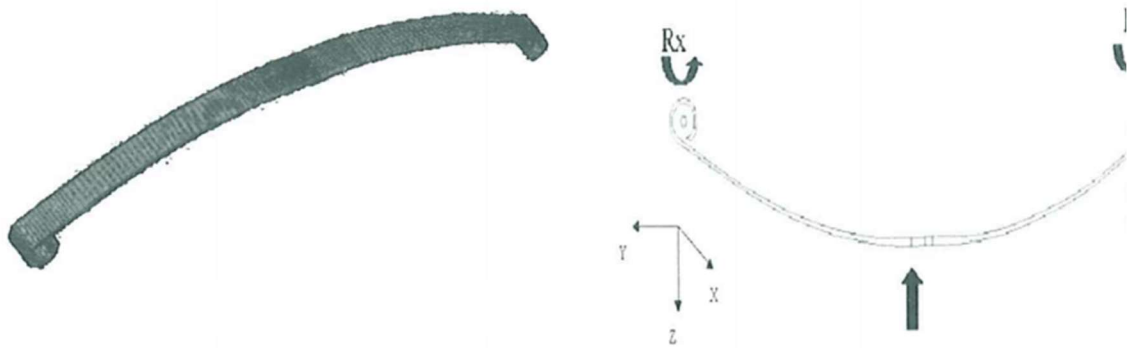


Figure 4: FEM of Leaf Spring (left) and Free Body Diagram Representation of Boundary Conditions in FEM (right) [7].

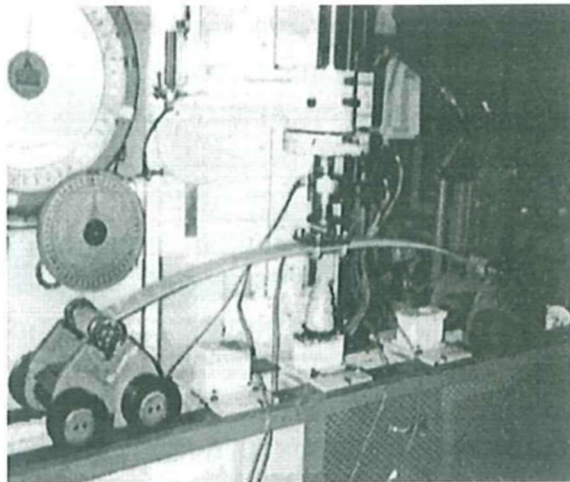


Figure 5: Test Rig Calibration of Leaf Spring [7].

Static and fatigue analysis was performed using the linear FEM. The basic parameters of fatigue analysis according to finite element theory were utilized in the analysis; linear analysis results, dynamic load data, and material specification, illustrated

in Figure 6 [7]. The most critical load acquired during road testing was utilized to execute the linear static analysis. The results of CAE analysis were similar to the laboratory tests.

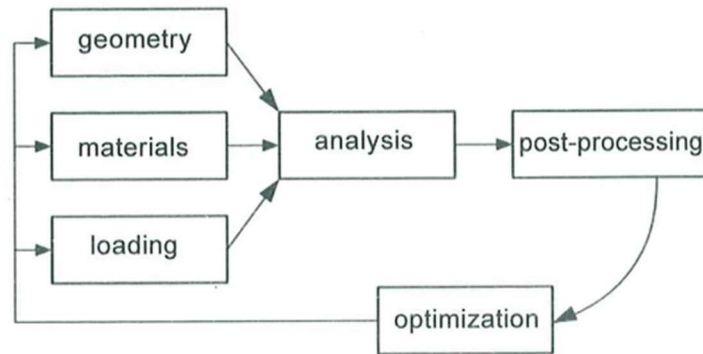


Figure 6: Static and Fatigue Analysis Based on FEM [7].

In industry, most load spectrum are already known, as many automotive manufacturers have conducted vehicle usage studies and have statistical models of customer usage as well as data from these loading conditions. Even in this case sometimes parts fail to perform as intended, in which case these components become candidates for instrumentation, allowing the strain at the suspected critical location to be recorded under vehicle loading conditions [8].

Design of Experiment

In engineering, experimentation is used in new product design as well as design iteration to identify the reasons for changes to an output variable given a change of an input variable or a process or system [9]. The planning and conduction of an experiment, design of experiment (DOE), is crucial in obtaining valid data so that objective conclusion can be determined through data analysis. For example, DOE can be designed to determine key product design characteristics though the specified DOE process. Dean and Voss state the outline of a DOE:

- a) Define the objectives of the experiment
- b) Identify all sources of variation, including:
 - i. Treatment factors and their levels,
 - ii. Experimental units,
 - iii. Blocking factors, noise factors, and covariates
- c) Choose a rule for assigning the experimental units to the treatments
- d) Specify the measurements to be made, the experimental procedure, and the anticipated difficulties
- e) Run a pilot experiment
- f) Specify the model
- g) Outline the analysis
- h) Calculate the number of observations that need to be taken
- i) Review the above decisions. Revisit, if necessary. [10, p. 8]

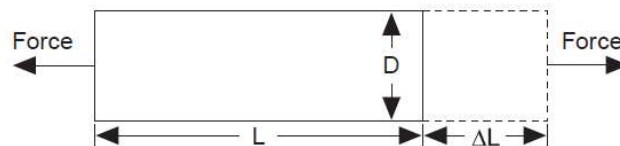
Where treatment factors are any item whose effect on the data is to be studied.

Following this process many automotive manufacturers have gathered service loading data from instrumentation under specified conditions [11, 12, 13]. The key determination of the DOE for this application is the applicable service loads, which can be difficult to determine with an acceptable confidence level [14]. The durability of a component is determined by its estimated service loads, therefore the accuracy of the service loads are of paramount importance to the design for durability of any component [15]. In the automotive industry, some service loading conditions are known through history and experience of each automotive manufacturer. On occasion engineering departments will take a survey of customers, through dealerships, technical societies, or

university reports to gather customer usage data, utilizing this data to determine the vehicle service loads [15]. There is a spectrum of customer usage for each vehicle and geographic location the in which the vehicle is sold depending on the type of driving performed by each customer and the severity of the roads [11]. Many studies have been conducted with accelerometers, though strain gauge data are collected at structural areas of concern [11, 12, 13].

Strain Gauge

Strain gauge technology has been in existence since 1936, invented by two different men at the same time, Simmons and Ruge [16]. Throughout this history, the production methods have changed significantly but the founding principle of measure has not change, nor has the basic electrical implementation [17]. Strain gauges are simple mechanisms comprised of a wire on a carrier. This carrier is mounted, through bonding, on the specimen to be measured. If bonded properly, “the strain gauge behaves as an integrated measuring element” and strain gradients can be measured along the active grid length, if they occur along the axis of the strain gauge measuring grid [17, p. 20]. The basic principle of measuring strain with a strain gauge is derived explicitly from the definition of strain itself. Strain (ε) is the quantity of deformation of a body due to an applied load, defined by change in length over original length, as illustrated in Figure 7 [18].



$$\varepsilon = \frac{\Delta L}{L}$$

Figure 7: Definition of Strain [18].

This definition yields Equation 1, the longitudinal strain.

$$\varepsilon = \frac{\Delta L}{L} \quad (1)$$

Utilizing this equation and the principle that copper and iron wires change in resistance when subjected to strain, a principle discovered by Lord Kelvin in 1856, the strain in a metallic strain gauge can be measured by applying a voltage to measure the change in resistance across the gauge [19].

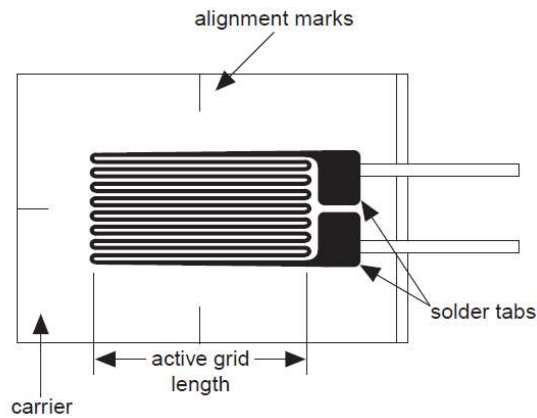


Figure 8: Metallic Strain Gauge [18].

As seen in Figure 8, the wire of a strain gauge is snaked back and forth along the active grid length achieving a compounding resistance or elongation as the specimen is subject to loading. This winding develops a sensitivity defined as the gauge factor, k [17]. The gauge factor directly relates the change in strain to the change in resistance as given in Equation 2.

$$k = \frac{\Delta R/R}{\Delta L/L} = \frac{\Delta R/R}{\varepsilon} \quad (2)$$

While the gauge factor can be derived by principles of wire geometry and Poisson's ratio, the gauge factor given by the manufacturer is recommended for measurements because the real gauge factor will vary from the calculated value due to manufacturing [17].

Magnitude of measured strain is small, in the order of microns or 10^{-6} . Due to this fact, the electrical measurement of strain must be very sensitive. The Wheatstone bridge circuit is the most suitable circuit for measuring sensitive resistance changes due to the ability of the circuit to measure resistances even with unstable voltage sources [20]. The circuit diagram, as shown in Figure 9, displays the circuit in a common format where the supplied voltage is defined as V_B and the measured voltage is V_M .

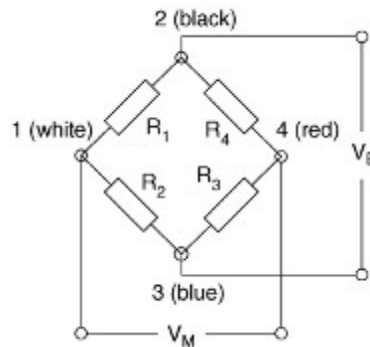


Figure 9: Wheatstone Bridge Circuit for Measurements Using Strain Gauges [17].

Through Kirchhoff's Laws the Wheatstone bridge yields the electrical relationship:

$$\frac{V_M}{V_B} = \frac{R_1}{R_1 + R_2} - \frac{R_3}{R_3 + R_4} \quad (3)$$

As can be deduced from Equation 3, the change in voltage across the circuit is dependent on the ratios of the resistances on each arm of the bridge. If the resistances are identical (i.e. $R_1=R_3$ and $R_2=R_4$) the voltage measured is equal to zero [17]. If any, or all, of the resistors are replaced with strain gauges, the strain can be measured through the change in voltage across the bridge, resolved with Equation 4.

$$\frac{V_M}{V_B} = \frac{(\Delta R_1 - \Delta R_2 + \Delta R_3 - \Delta R_4)}{2(2R_0 + \Delta R_1 + \Delta R_2 + \Delta R_3 + \Delta R_4)} \quad (4)$$

Assuming small angle theorem, as the relative change of each strain gauge to the initial strain value is very small:

$$\frac{V_M}{V_B} = \frac{(\Delta R_1 - \Delta R_2 + \Delta R_3 - \Delta R_4)}{2(2R_0)} \quad (5)$$

While strain gauges are simple mechanisms, their implementation and usage must be executed with great care and forethought. Some important considerations for the implementation and utilization of strain gauges include transverse sensitivity, temperature effects, mechanics of the strain gauge, dynamic behavior, installation methods, Wheatstone bridge circuit implementation, cable resistance considerations, signal processing, and calibration.

To mitigate the effects of temperature on the strain reading, the minimum number of strain gauges required to measure strain in any direction is two. The wound wire inside the strain gauge changes in length as the specimen is subjected to elongation or compression. However, the wire is so small it is also sensitive to changes in temperature. To cancel the change in temperature, two strain gauges are required in a half bridge

configuration, as seen in Figure 10 c). Figure 10 illustrates a half bridge with one active strain gauge and a compensation strain gauge; in Figure 10 a) no temperature effects are present, therefore pure strain is measured, b) temperature change effecting only the active strain gauge, measuring change in gauge length due to strain and temperature, c) temperature change effecting both the active gauge and compensation gauge, measuring pure strain [17].

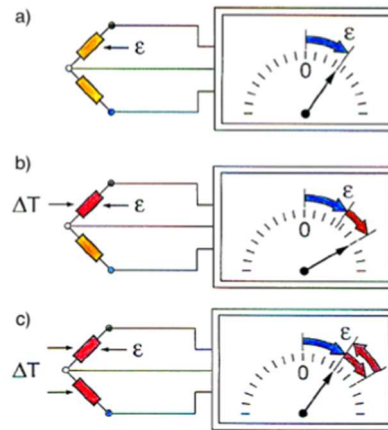


Figure 10: Visual representation of the effects of temperature on the reading of a strain gauge [17].

Another important consideration for the implementation of strain gauges is transverse sensitivity and transverse strain, as mentioned above. Like longitudinal strain, as defined in Equation 1, transverse strain is the measure of the change in length over the original length. However, the length being measured is perpendicular, or transverse, to the strain gauge measurement direction [17]. If present in Figure 7, the transverse strain, ε_2 , would be represented as:

$$\varepsilon_2 = \frac{\Delta D}{D} \quad (6)$$

For any given material, the material property represented by Poisson's ratio, ν , is mathematically defined as the material's transverse strain sensitivity in proportion to its longitudinal strain, ε_1 , sensitivity:

$$\nu = \left| \frac{\varepsilon_2}{\varepsilon_1} \right| \quad (7)$$

Therefore, the transverse strain measurement is equal to Poisson's ratio times the longitudinal strain measurement.

$$\varepsilon_2 = -\nu \varepsilon_1 \quad (8)$$

The change of sign in Equation 8 is due to the mechanics of the material. If the bar in Figure 7 was elongated, the measurement D would get smaller therefore transverse and longitudinal strain must be of opposite sign.

As mentioned previously, strain measurements are of an exceedingly small order of magnitude, on the order of microns. While the sensitivity of the Wheatstone bridge preserves the accuracy of change in resistance signal to a change in voltage, data acquisition (DAQ) systems typically require amplification at these excitation levels [21]. Amplifiers are utilized to achieve this requirement as well as signal conditioning, balancing the bridge, and low-pass filtering [21].

Low-pass filtering enables the removal of some aliasing effects on the measured signal. When the analog filter whose half-power point is set to the Nyquist frequency f_0 , defined $f_0 = 1/2 \Delta t$, is used as the low-pass filter, 50% of the energy is reduced at f_0 . If data is then collected at a sample rate of Δt , "aliasing restores the energy to nearly its true value at f_0 " [22, p. 262].

Bridge balancing can be accomplished with trimming potentiometers to compensate for inexact resistors [23]. Balancing adjustment can also be utilized in the amplifier to adjust the analog signal zero point to give the largest measurement range [24]. If the analog to digital conversion (ADC) range is 0-5 volts, a nominal 2.5 volt reading at zero excitation would give the largest positive and negative range.

ADCs transform the continuous analog signal into “a discrete binary code suitable for digital processing” [21, p. 139]. For application with strain gauges the analogue voltage signal must be converted to a digital signal be recorded. The accuracy of the chosen ADC must be considered, accuracy requirements include gain error, offset error, linearity, no missing codes, and evaluation over the expected temperature range. Conversion speed and system power limitations can also be determining factors in the choice of ADCs.

Data Analysis

Linear Regression Modeling for Calibration. Regression analysis was developed to find the best relationship between a response variable, Y , and one or more explanatory variables, x , as well as quantifying the strength of the relationship [25]. In scientific study, there exists some unknown parameters in the measured relationship between x and Y , the method of least squares regression analysis is often used to estimate the unknown parameters in this relationship through the generation of a regression line, shown in Equation 9 [26].

$$\hat{Y} = a + bX \quad (9)$$

Matlab uses the Least Squares linear regression method [27]. This method calculates the coefficients a and b of the linear regression line so that the sum of the

squares of the errors (SSE) about the regression line is minimum [25]. Where y_i is the known value of the response variable, a and b are defined to minimize SSE as defined in Equation 10.

$$SSE = \sum_{i=1}^n (y_i - a - bx_i)^2 \quad (10)$$

Individual equations for a and b are derived from Equation 10 using partial differential equations to yield Equations 11 and 12.

$$b = \frac{n \sum_{i=1}^n x_i y_i - (\sum_{i=1}^n x_i)(\sum_{i=1}^n y_i)}{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2} \quad (11)$$

$$a = \frac{\sum_{i=1}^n y_i - b \sum_{i=1}^n x_i}{n} \quad (12)$$

Where, x_i is the explanatory variable, y_i is the response variable, and n is the index of each datapoint.

The variance, σ^2 , the squared deviation of the observed response variable, y_i , and the corresponding point to the linear regression equation, \hat{y}_i [25]. Mean squared error (MSE), s^2 , estimates σ^2 in Equation 13.

$$s^2 = \frac{SSE}{n - 2} \quad (13)$$

A good linear fit of the regression has a small mean squared error. The square root of MSE, root mean squared error (RMSE), is the difference between the predicted response variable and observed response variable [28]. RMSE value is of the same

magnitude as the data therefore, RMSE is only a good predictor of model accuracy as a percentage of the error induced for each individual datapoint ($RMSE/y_i * 100$) [29].

The coefficient of determination, R^2 , is the proportion of variability in the model that is unexplained, SSE, to the proportion of variability in the model that is explained, total corrected sum of squares (SST) defined in Equation 14 [25].

$$SST = \sum_{i=0}^n (y_i - \bar{y}_i)^2 \quad (14)$$

Where, \bar{y}_i is the mean of the observed response variable for the data set. SST ideally represents the variation in the observed response variable explained by the model. Therefore, R^2 is defined

$$R^2 = 1 - \frac{SSE}{SST} \quad (15)$$

It is important to note the reliability of R^2 depends on the size of the data set and the type of data set to be analyzed. While an R^2 value of 0.98 might be too low for a chemist to accept an experiment's findings, an R^2 value of 0.7 might be high enough for a psychologist to validate their experiment. R^2 criterion also becomes dangerous when comparing different models of the same data set, if one datapoint is added the R^2 value is artificially increased. For these reasons it is not recommended to use R^2 as the sole descriptor of a model when analyzing its validity [25].

When implemented correctly, a linear regression model provides a relationship between a measured response variable and a known explanatory (input) variable [26]. When dealing with instrumentation, linear regression is used as a

calibration tool in the automotive industry, transportation industry, and at universities. The Texas Department of Transportation (DOT) contracted the Texas Transportation Institute (TTI) at Texas A&M University to conduct a study on the effects of road surface finish on vehicle dynamic loads and pavement life [30]. In this study, TTI instrumented a trailer axle with strain gauges, setup to measure in the vertical loading direction. This instrumentation was calibrated using specialized equipment, MTS loading system, to apply the highly accurate loads, reading the strain through a data acquisition system. The measured load-strain relationship was linear, as expected, therefore, a linear regression line was developed to characterize the load-strain relationship, calibration. Some data points from the calibration can be seen with the regression line in Figure 11.

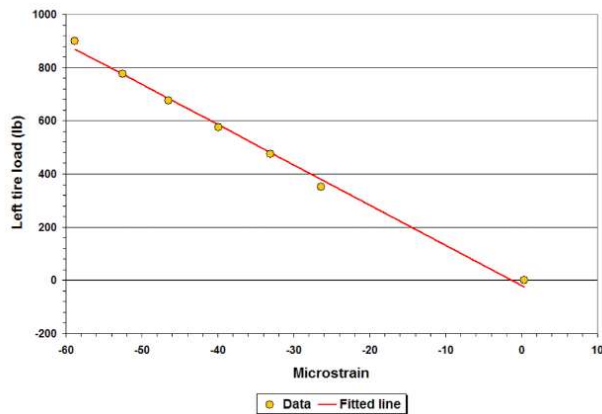


Figure 11: Data from Calibration of Trailer Axle [30].

At Mahindra, an automotive company in India, strain gauges installed on a steering track rod were calibrated known tension and compression loads were applied to the component through a test fixture [31]. Strain data was collected during each loading application and analyzed to establish the calibrated strain-load relationship in Figure 12.

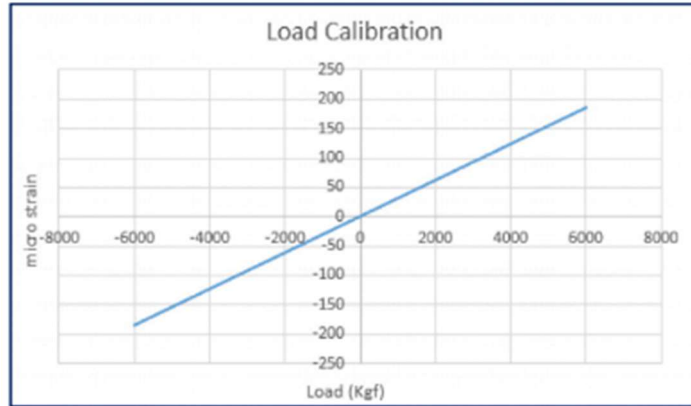


Figure 12: Track Rod Strain-Load Calibration [31].

In another study at Mahindra, the torque on a driveshaft, CV-half shaft, was measured using a strain gauge installation [32]. In this study, the strain-load characterization was calibrated using specialized test equipment to apply a known torque on the instrumented driveshaft.

Finally, at the university level, a study was performed to calibrate a six-degree of freedom (DOF) force and torque sensor [33]. In each direction, the force and moment about that direction were applied and the strain was measured. A linear regression was performed to calibrate the gauge reading to the known force. All six forces and torques were calibrated within 1% error. One linear regression calibration is shown in Figure 13.

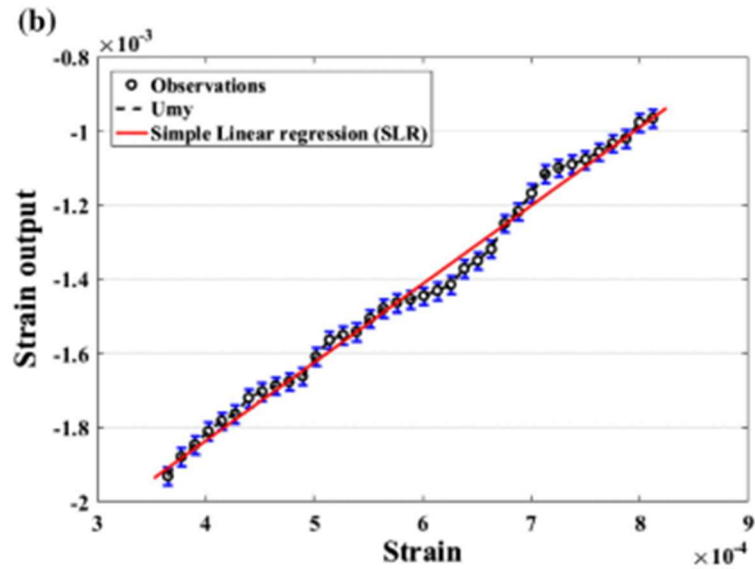


Figure 13: Linear Regression and Error Bars for Strain Output due to Torque about Y [33].

From university to industry, linear regression is a useful and widely used tool for calibrating data. Careful inspection of the regression characteristics, MSE, RMSE, and R^2 , should be conducted to ensure the validity of the model before use. If the model strongly correlates to the data set, linear regression is a useful data analysis tool.

Digital Filtering. The data sampling rate was determined by the designed event length and speed. Within this data collection frequency, noise is introduced to the sensor through the vibration of the system. On the vehicle, there are many frequency components, generated by mechanical sources not relevant to the system's performance [23]. To understand the structural system performance only low-frequencies were of interest, introducing the need for a low-pass digital filter. This filter was specifically designed to remove unwanted frequencies utilizing the fast Fourier transform (FFT) function, `{fft()}`, in Matlab.

Digital filtering is performed in the frequency domain, instead of the time domain, allowing the data to be filtered without adding any phase angle or distortion [34]. The Fourier transform is the mathematical tool used to transform a time domain data series into the frequency domain [35]. Once in the frequency domain, the contribution of different frequencies can be adjusted and the inverse Fourier transform performed to return to the time domain, without any distortion of the original signal [23]. There are two main types of Fourier transforms, continuous- and discrete-time Fourier transforms. Continuous-time Fourier transforms are good for real-time processing of constant signals while discrete-time Fourier transforms are utilized to analyze data with set sampling intervals. The discrete Fourier transform is said to be the “backbone of modern digital signal processing” [35, pp. I-1]. The FFT is a special form of the discrete-time Fourier transform developed by Cooley and Tukey to reduce the computational time of performing the transformation [34]. Computational efficiency of the Cooley-Tukey method is achieved with an added requirement; the data string must be of a power of two in length.

In the automotive industry, Daimler-Chrysler Canada and other original equipment manufacturers (OEMs) utilize digital filter designs based on the FFT to analyze vehicle instrumentation data collected from an array of devices, including wheel force transducers, strain gauges, accelerometers, and rotary variable inductance transducers [36]. Using a band-pass filter to removed unwanted frequency from the data, Daimler-Chrysler Canada were able to create damage and displacement profiles. The output loads and displacements were used to validate a finite element analysis.

The Matlab fast Fourier transform is based off the FFTW Fourier transform software, developed at the Massachusetts Institute of Technology [37]. FFTW is an adaptive, high performance implementation of the Cooley-Tukey [38] FFT algorithm specifically designed to reduce the computational time to perform the discrete-time Fourier transform. FFTW outperforms similar programs in computational speed and result accuracy while maintaining the ability to perform one- and multi-dimensional transforms without restricting input sizes to only powers of two.

Simplified from Cooley-Tukey [38], the main equations from FFTW, presented by Matlab [39] are defined as follows, where X is the dataset in the time domain, Y is the Fourier transformed dataset into the frequency domain of a dataset of length n .

$$Y(k) = \sum_{j=1}^n X(j)W_n^{(j-1)(k-1)} \quad (16)$$

$$X(j) = \frac{1}{n} \sum_{k=1}^n Y(k)W_n^{-(j-1)(k-1)} \quad (17)$$

Where

$$W_n = e^{(-2\pi i)/n} \quad (18)$$

is one of n roots of unity.

Equation 16 is the overall equation executed when the Fourier transform is taken utilizing the `{fft()}` function in Matlab, while Equation 17 is the overall equation

executed when the inverse Fourier transform is taken utilizing the `{ifft()}` function in Matlab [39].

The designed low-pass frequency must be taken with particular attention to the effects of signal aliasing, as described below.

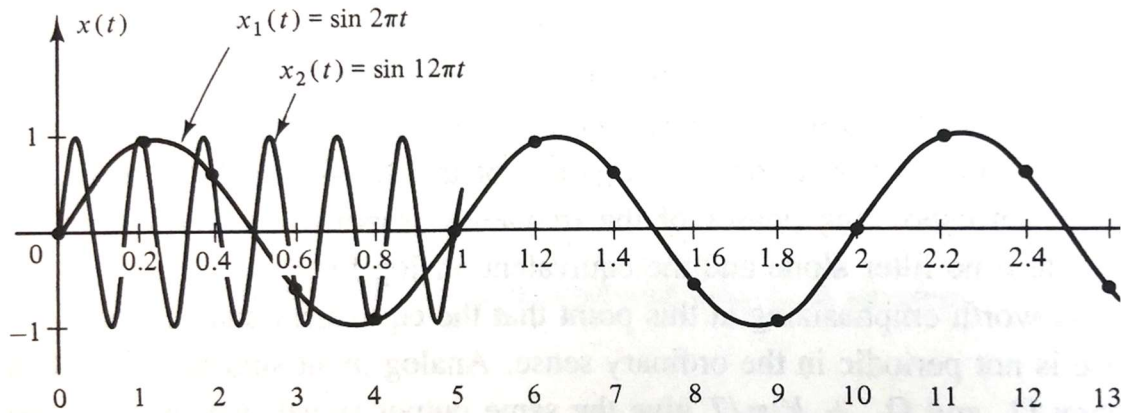


Figure 14: Two Continuous Sinusoidal Signals Sampled at 0.2 Sec [40].

In Figure 14, two continuous sinusoidal signals of different frequencies are sampled every 0.2 seconds. The discrete points generated from sampling each signal every 0.2 seconds are the same values, causing the higher frequency analog signal, when sampled, to appear like the lower frequency signal. This phenomenon is called aliasing [40]. Due to aliasing, the cutoff frequency in low-pass filtering must be significantly larger than the frequency of interest, 10 to 20 times higher than the frequency of interest [22].

Confidence Interval. Of the various types of statistical intervals calculable from sample data, the confidence interval contains an unknown characteristic of the sampled population or sampling process [26]. Therefore, the confidence interval can help account for some of the unknown variability in the sampling process, giving the analyst a higher degree of confidence by providing the probability of selecting a random sample that is

within the specified interval range [25]. For a Gaussian distribution, where the sample mean \bar{X} and sample standard deviations are known, the 95% confidence interval is defined as $\bar{X} \pm 1.96s$ [21]. As a rule-of-thumb, the confidence interval proportion, 1.96, can be estimated to be 2 [41].

\bar{X} is the mean of the sample population and s is the standard deviation of the sample population, as defined in Equations 19 and 20.

The sample mean, \bar{X} , is defined as the average of the samples observed.

$$\bar{X} = \frac{\sum_{i=1}^n x_i}{n} \quad (19)$$

Sample standard deviation, s , is the square root of the average squared deviation of individual observations from the sample mean.

$$s = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{X})^2}{n - 1}} \quad (20)$$

It is important to utilize uncertainty analysis (confidence intervals) to estimate the uncertainty in the measured data [42]. Confidence intervals can give an acceptable level of confidence associated with the measured response variable for a normal distribution. The central limit theorem suggests that with a sufficiently large samples size, the sample mean will be approximately normally distributed [41]. Therefore, if the sample size is large enough, a confidence interval, of x standard deviations from the mean, can be utilized to account for some of the unknown variability in the data collection process. For smaller sample sizes, Wei [43] at Tenneco Automotive, established the accuracy of the confidence interval to provide a conservative design limit on strain data, noting that a sample size of three is too small and would result in an overly-conservative estimate.

Finite Element Modeling

To perform a linear static analysis on the CAD of a component, a finite element (FE) software must be utilized. The FE method is popular because of its speed in performing a large number of computations, enabling the software to analyze problems with complex geometry and loading conditions in a wide variety of engineering problems [44]. FE software execute a system of partial equations in the form of matrix mathematics. There are three matrices used to calculate the displacement in each node due to the boundary conditions; force matrix, structural matrix, and displacement matrix. The structural matrix represents the geometric and material properties of the model. These properties define the structural characteristics of the element to resist deformation under loading, allowing the forces and displacements to be correlated at each node of the elements. There are two types of structural matrices; the stiffness matrix, and the transfer matrix. The stiffness matrix defines the relationship of the displacements at the nodes to the forces while the transfer matrix defines the relationship of the displacements and forces of one node to another [44].

$$\begin{bmatrix} V_a \\ M_a \\ V_b \\ M_b \end{bmatrix} = \begin{bmatrix} \frac{12EJ}{\ell^3} & -\frac{6EJ}{\ell^2} & -\frac{12EJ}{\ell^3} & -\frac{6EJ}{\ell^2} \\ -\frac{6EJ}{\ell^2} & \frac{4EJ}{\ell} & \frac{6EJ}{\ell^2} & \frac{2EJ}{\ell} \\ \frac{12EJ}{\ell^3} & \frac{6EJ}{\ell^2} & \frac{12EJ}{\ell^3} & \frac{6EJ}{\ell^2} \\ -\frac{6EJ}{\ell^2} & \frac{2EJ}{\ell} & \frac{6EJ}{\ell^2} & \frac{4EJ}{\ell} \end{bmatrix} \begin{bmatrix} w_a \\ \theta_a \\ w_b \\ \theta_b \end{bmatrix}$$

Figure 15: Force Matrix (Left), Structural Matrix (center), Displacement Matrix (Right) [44].

FEMs are approximate by nature. Due to the use of a finite number of elements, the displacements not at a nodal location are not solved. FEMs are also only as good as the input characteristics; load case, geometry, constraints, material properties, and mesh [44].

Using the methodology described above an FEA is performed by a series of steps:

- Divide the structure into finite elements, known as meshing.
- Define the properties of the elements, material characteristics as well as the type of FEs.
- Define the constraints on the model.
- Define the loads acting on the model.
- Solve the system of equations.
- Calculate the desired output, stress, strain, displacement, factor of safety, etc.

The precision of each of these steps is critically important to the precision of the model. An error in any of these steps will propagate through the simulation and result in a large error in the result.

Each software has a different user interface for selection of the input variables but the fundamental methods of solving the matrices are based in the same mathematics. Some popularly used software are Nastran, Abaqus, Ansys, and Solidworks Simulation [44]. The main difference in software are the specific theorems implemented to solve a model and the logic in determining what theorem best fits each model. This software logic is not published as it is more beneficial for the corporations who create the software to keep them private. One key aspect of the FEM that information is published about is the mesh.

Ansys and Solidworks both publish what type of mesh elements are utilized in their FE software [45, 46].

Solidworks Simulation “high quality” mesh uses second-order tetrahedral solid elements [45]. As displayed below in Figure 16, second-order tetrahedral elements have a much higher resolution than first-order tetrahedral elements, 30-DOF as compared to 12-DOF [47].

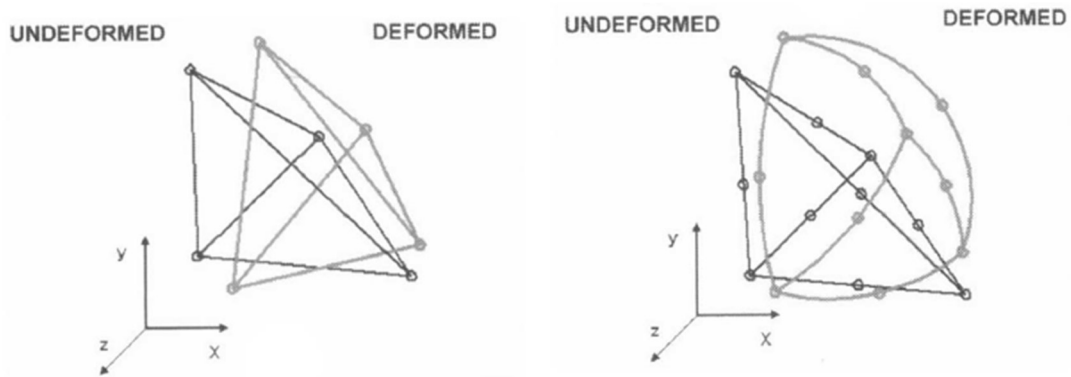


Figure 16: First-Order Tetrahedral Element, Left, Second-Order Tetrahedral Element, Right [47].

The Solidworks Simulation second-order tetrahedral solid element is the same type of element used by ANSYS [48]. ANSYS is a commonly used FE modeling software that uses 3-D quadratic tetrahedral elements as of their newest software release, 19.0 [46]. 3-D quadratic is another way of saying second-order as second order is quadratic.

The accuracy of the elements is important to note, as any variation in calculation of elements can lead to vastly differing results in FE models. Previously the accuracy of tetrahedral and hexahedral mesh elements has been compared to beam theory, with a high degree of accuracy in tension, bending, and torsion, as displayed in Figure 17: Stress

Concentration Test for Different Elements . Where the closed solution is beam theory [48].

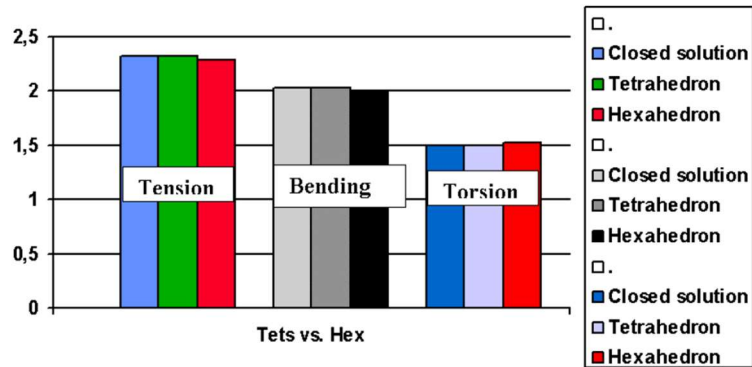


Figure 17: Stress Concentration Test for Different Elements [48].

Solidworks is becoming more widely used at universities and in many industries including the automotive industry, particularly in race vehicle design [49, 50, 51, 52, 53]. Donkervoort Automobielen BV, a Dutch automotive manufacturer utilizes Solidworks to design their high-performance vehicles. Using the integrated design and simulation platform to save time in the design process [50]. University design teams utilize Solidworks CAD and Solidworks Simulation because its ease of use reduces the learning curve for new members while the integration between simulation and design reduce design iteration time [51].

Topology Optimization

Structural design optimization turns design constraints into a minimization problem with the goal of finding the optimal solution of the variable(s), x , to minimize a function, $f(x)$ [54]. Components of a structure that can be used as a design variable include material, cross-sectional shape and dimensions, joint and member assembly, joint location, and types of joints and supports used. For example, for a beam under axial,

torsional, and bending loads with specified material, constraints, and geometric envelope, the cross-sectional shape can be optimized to minimize weight while meeting failure criteria. Axial loading capacity depends on the cross-sectional area of the beam, as determined by the normal stress in the beam under axial loading in Equation (21) [55].

$$\sigma = \frac{P}{A} \quad (21)$$

Where bending capacity depends on the second area moment of inertia of the beam, I , also determined by normal stress in the beam under bending in Equation(22).

$$\sigma = \frac{My}{I} \quad (22)$$

While the torsional loading capability depends on the polar moment of inertia, J , as determined by shear stress in the beam under torsion in Equation(23).

$$\tau = \frac{Ty}{J} \quad (23)$$

The cross-sectional area, second area moment of inertia, and the polar moment of inertia are determined based on the shape of the beam.

For a hollow tube, the cross-sectional area and second moment of inertia are given in [56] while the polar moment of inertia is given in [57], displayed in Equations (24)(25), and (26) respectively.

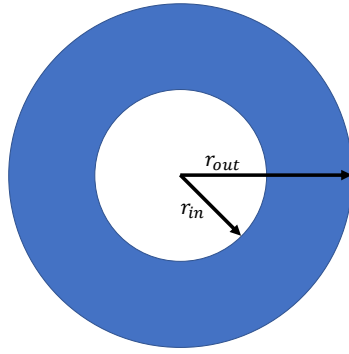


Figure 18: Hollow Tube, Inner and Outer Radii

$$A = \pi(r_{out}^2 - r_{in}^2) \quad (24)$$

$$I = \frac{\pi}{4}(r_{out}^4 - r_{in}^4) \quad (25)$$

$$J = \frac{\pi}{2}(r_{out}^4 - r_{in}^4) \quad (26)$$

For a square tube, the cross-sectional area, second area moment of inertia, and polar moment of inertia are derived in Equations 27, 28, and 29 respectively derived from Rajan (2001) [54].

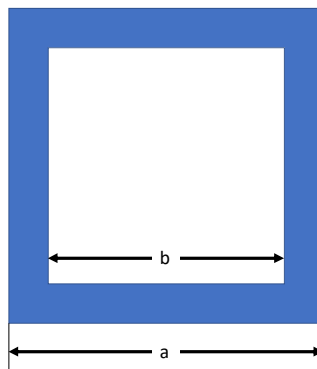


Figure 19: Square Tube, Inner and Outer Edge Length

$$A = a^2 - b^2 \quad (27)$$

$$I = \frac{a^4 - b^4}{12} \quad (28)$$

$$J = \frac{a^4 - b^4}{6} \quad (29)$$

In the automotive industry complex optimization equations are created and utilized to solve more complex geometrical cross-sectional shapes under various constraints; material, space, etc. [58]. Others use complex optimization software, like Altair's Optistruct, to optimize for example the thickness of a beam, to get the minimum volume, given the constraints and loads as well as the optimization parameters of maximum fatigue life and maximum allowable deflection [14]. The topological results from complex simulation can sometimes be used directly for component final design but this is not always the best or most manufacturable solution [59]. Topological results are an efficient way to find the idealized shape and work from this to make the component manufacturable.

Chapter III

Methodology

After testing the 2016 Women's Baja vehicle, Nichole, the front passenger lower fore control link bent, as see in Figures Figure 20 and Figure 21.



Figure 20: Right Front Lower Fore Control Arm, Deformation



Figure 21: Control Arm Deformation Area, Enhanced

The deformation failure present in the component was not predicted with the loading condition used in initial model validation. To understand the most common load case in the component, DOE methods were used to design an experimental test of this load case, measure the strain present in the member, and re-design the member to withstand this loading condition.

To validate this load case, it must first be defined. In the automotive industry, manufacturers have to deal with a wide service loading variation but fortunately for Baja, the vehicle service life has little variation as it is only designed to meet the loading conditions experienced at competition [15]. Competition loading conditions vary according to where the track is setup and who leads this charge. However, there are some loading conditions that consistently occur at each competition. One of these loading conditions is an obstacle that is hard, stationary, fixed and between 3 and 12-in tall; obstacles can include logs, concrete blocks, and rocks.



Figure 22: Concrete Block, Approximately 3.5in High



Figure 23: Logs, Approximately 15in in Diameter



Figure 24: Muddy Log, Approximately 8-12in Protruding



Figure 25: Large Concrete Chunk, Approximately 1 ft in Height

To replicate this common load case on campus, an obstacle similar in shape and height is utilized, a parking chalk. Parking chocks on campus vary in shape and size but the ones near the Mechanical Engineering building are 8-in wide and 5-in tall. This height is close to some of these obstacles and similar in shape.



Figure 26: Parking Chalk Width Measurement



Figure 27: Parking Calk Height Measurement

Obstacles of this magnitude are typically contacted at a middle range speed, around 12 mph. Though they can be contacted at full speed, around 30 mph, if the driver does not see the obstacle. For combined loading, in axial and bending, the worst case is contacting the parking calk straight on therefore this is the chosen loading condition.

The initial FEM used for component design validation, with the approximation of this loading condition, was used as a starting point to identify the area where the strain in the component would be highest in order to instrument the component in that location.

For the DOE the objectives are to:

1. Obtain three statistically calibrated maximum strain gauge measurements from each run of the event. One in the axial direction and two in bending about SAE-coordinate x and z axis respectively during the dynamic loading condition defined above. Along with vehicle speed and position captured by global positioning system (GPS).
2. Statically model the measured event to calculate each strain, ϵ_{Mb_x} , ϵ_{Mb_z} , and ϵ_{F_y} , using a 95% confidence interval.

3. Utilize the linear FEM to calculate the remote load based on the strain in each direction.
4. Re-Design the component to withstand this load case under fatigue and to structurally withstand the load case scaled, by momentum, for the top speed of the vehicle.

Strain Gauge Installation and Instrumentation

To achieve the first objective of the DOE, strain gauges must be specified and sourced, strain gauge layout must be determined, an amplifier must be specified and sourced, and a data acquisition system must be selected, or designed, and programmed.

Strain gauges were then specified and purchased based on industry recommended manufacturers, excitation voltage of DAQ system, and ease of use. Omega manufactures strain gauges for many industrial applications and are widely used in the automotive industry. The strain gauge selected is specified in Table 1.

Table 1

Strain Gauge Specification

Company	Type	Resistance	Grid Length	Grid Width	Tolerance	Gauge Factor	Product Name
Omega	Linear	350 ohms	4.5mm	3.2mm	+/- 0.30%	2.13	SGD-5/350-LY11

The strain gauge resistance was determined by the supply voltage of the DAQ, 5 volts. For stability of the strain gauge zero balance no more than 10 to 15 mA is recommended to go through the strain gauge [17]. This would mean a 350 Ω resistor

could withstand anywhere from 5 to 10 volts where the next common size down, 120 Ω , would only be recommended to work between 2 and 4 volts.

For the same gauge specifications, there are many grid dimensions to choose from, both in width and length. The larger the strain gauge the easier it is to align correctly making installation more accurate. The chosen strain gauge was close to the length of the elements in the FEM and not too wide to fit on the curved component. The gauge also came with ribbon leads, for easier installation to the bondable terminal pads, and was built to be mounted on steel.

A half bridge installation was used to cancel temperature changes and use the smallest number of strain gauges. Utilizing Equation (5), a half-bridge was circuit derived to measure strain of a bar in bending, with strain gauges in the R_1 and R_4 positions.

Therefore:

$$\frac{V_M}{V_B} = \frac{(\Delta R_1 - 0 + 0 - \Delta R_4)}{2(2R_0)} \quad (30)$$

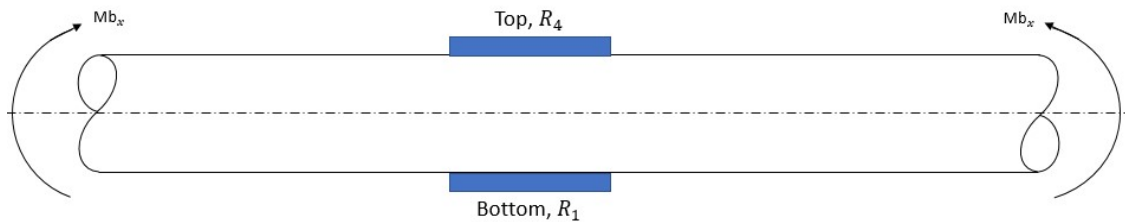


Figure 28: Strain Gauge Longitudinal Strain Measurement of a Beam in about the SAE defined z-direction in a Half Bridge

For strain in bending about the SAE defined x-direction a half-bridge, strain gauge set 1 (SG1), was installed with R_1 is on the bottom of the bar and R_4 is on the top of the bar, from a view point looking at the vehicle from the front, as illustrated above in

Figure 28. The longitudinal strain is the only strain being measured and is defined for each gauge as:

$$R_1 = R_0(1 - k\varepsilon) \quad (31)$$

$$R_4 = R_0(1 + k\varepsilon) \quad (32)$$

With the positive moment defined in Figure 28, the equation of the half bridge can be derived from Equation 30:

$$\frac{V_M}{V_B} = \frac{R_0(1 - k\varepsilon) - R_0(1 + k\varepsilon)}{2(2R_0)} \quad (33)$$

$$\frac{V_M}{V_B} = \frac{R_0(-2k\varepsilon)}{2(2R_0)} \quad (34)$$

$$\frac{V_M}{V_B} = \frac{-k\varepsilon}{2} \quad (35)$$

The sign of the measurement was confirmed with the calibration, when a force is applied to cause a negative moment the circuit measures a positive strain. This is considered in the calibration.

For strain in bending about the SAE defined z-direction a half-bridge, strain gauge set 2 (SG2), was installed with, R₁ is on the front of the bar and R₄ is on the back of the bar, from a view point looking down from above the vehicle, as illustrated below in Figure 29.

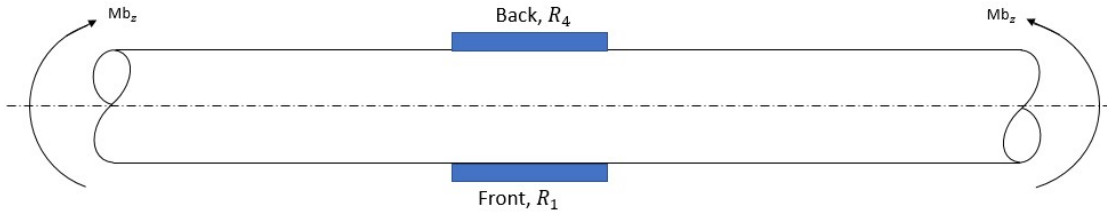


Figure 29: Strain Gauge Longitudinal Strain Measurement of a Beam in Bending about the SAE defined z-direction in a Half Bridge.

The derivation of the strain measured in the half-bridge created above is the same as the half-bridge created by the bending about the x-direction. The sign of the measurement was confirmed with the calibration and accounted for in the calibration.

One method of measuring axial strain utilizing a half-bridge is to measure the transverse strain. For this experiment one strain gauge in the half-bridge, strain gauge set 3 (SG3), was setup to measure both transverse and longitudinal strain illustrated below in Figure 30.

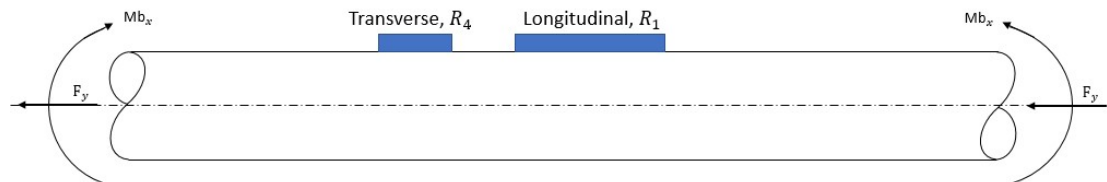


Figure 30: Strain Gauge Set 3, Longitudinal and Transverse Strain Measurement of a Beam Subject to Bending and Axial Force.

As depicted in Figure 30, the strain gauge in bridge position 4 is installed transverse to the y-axis and the strain gauge in bridge position 1 is installed longitudinally along the y-axis. This strain gauge setup measures the axial strain and the bending strain in the bar as derived, starting with Equation 30, below.

$$\varepsilon_y = (\varepsilon_{mb} + \varepsilon_{F_y}) \quad (36)$$

$$\Delta R_4 = -\nu \varepsilon_y \quad (37)$$

$$\Delta R_1 = \varepsilon_y \quad (38)$$

$$\frac{\Delta R}{R} = k\varepsilon \quad (39)$$

$$\frac{V_M}{V_B} = \frac{\Delta R_1}{4R_0} - \frac{\Delta R_4}{4R_0} \quad (40)$$

$$\frac{V_M}{V_B} = \frac{k}{4} \varepsilon_y - \frac{k}{4} (-\nu \varepsilon_y) \quad (41)$$

$$\frac{V_M}{V_B} = \frac{k}{4} (1 + \nu) \varepsilon_y \quad (42)$$

$$\frac{V_M}{V_B} = \frac{k}{4} (1 + \nu) (\varepsilon_{mb} + \varepsilon_{F_y}) \quad (43)$$

To obtain pure axial strain from SG3, the bending strain must be removed after the measurements are taken. This bending strain is the same bending strain measured in SG1. Re-writing Equation 43 in terms of axial strain:

$$\varepsilon_{F_y} = \frac{V_{M_{SG3}}}{V_B} \frac{4}{k(1 + \nu)} - \varepsilon_{mb} \quad (44)$$

From Equation 35:

$$\varepsilon_{mb} = -\frac{V_{M_{SG1}}}{V_B} \frac{2}{k} \quad (45)$$

Combining Equations 44 and 45, resolves the axial strain in the member.

$$\varepsilon_{F_y} = \frac{V_{MSG3}}{V_B} \frac{4}{k(1 + \nu)} + \frac{V_{MSG1}}{V_B} \frac{2}{k} \quad (46)$$

The location of the strain gauges was determined by the initial FEM performed on the component with the approximate loading condition applied, as previously mentioned. The FEM strain result of this approximate loading condition can be seen in Figure 31. The element selected is one of the highest strain locations on the member, no due to small angle error at the mounting point of the spherical bearing attachment to the member. This location is also close to the failure location identified in Figure 20 and Figure 21.

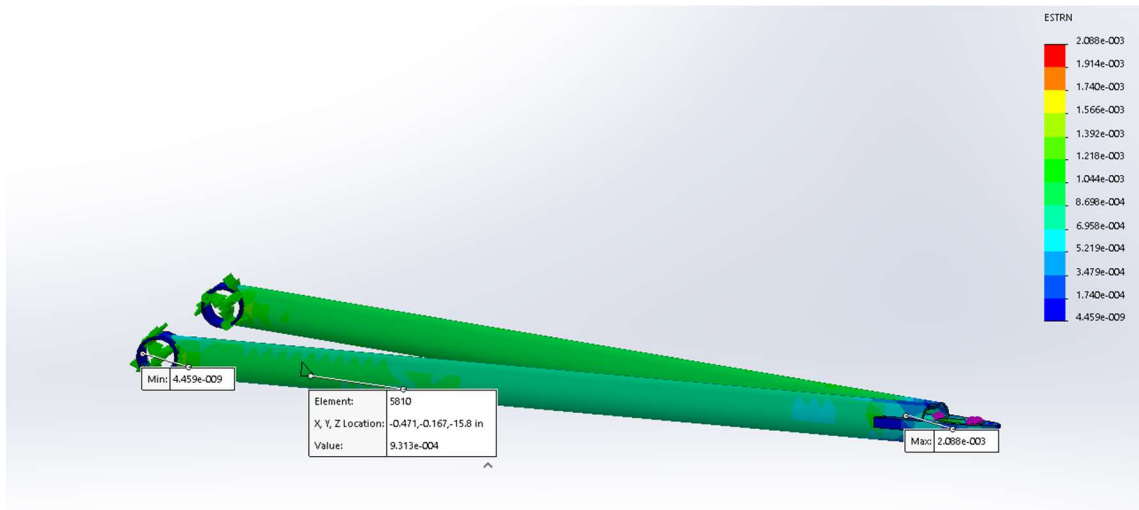


Figure 31: Component Strain Due to Initial Loading Condition Estimation.

The strain gauges were then installed on the left lower fore control link following the Vishay installation guide provided by the ERAU Aerospace Engineering (AE) materials lab instructor [60]. Vishay is a strain gauge manufacturer that also produces strain gauge adhesives. As Omega does not provide a guide or recommendation for the adhesive used for installation. The strain gauges used can be secured with either “cold or hot curing adhesives” [61]. There is no specific adhesive in the strain gauge datasheet, only that the strain gauge and the soldering pads be affixed with the same adhesive. The

chemical composition of strain gauge adhesives are similar between manufacturers therefore the Vishay M-Bond 200 adhesive donated by the ERAU AE materials lab instructor were used out of convenience. M-Bond 200 is a good adhesive to use for this experiment as it is designed to withstand high elongation tests, exceeding 60,000 micro strain ($\mu\epsilon$). Normal operating temperature range of the adhesive is -25° to 150°F , significantly above the range of the experiment 70° to 85°F .

M-Bond 200 is recommended, for best reliability, to be installed at a temperature range of 70° to 85°F and between 30% and 65% relative humidity [60]. This criterion was met by installing the strain gauges in the lab, where the temperature and humidity are within this range. The strain gauge installation guide [60] was followed step by step to secure the strain gauges and the solder pads to the specimen, notes on any deviations taken and the installation guide can be seen in Appendix B. The required products for complete installation of the strain gauges are listed:

- M-Bond 200 Adhesive
- PDT-1 6600 Drafting Tape
- M-Flux AR-2 Activated Rosin soldering Flux
- M-Coat A Air-Drying Polyurethane Coating
- M-Line Rosin Solvent
- M-Prep Neutralizer
- Isopropyl Alcohol
- GSP-1 Gauze Sponges
- CPS-1 Cotton Swabs
- PCT – 2M Gauge Installation Tape

After securing the strain gauges and solder pads to the specimen, via the adhesive process described above, the ribbon leads and lead wires need to be soldered to the solder pads. This process is described in Vishay application note TT-609 [62]. After all soldering was complete, the strain gauge installation needed to be protected, a polyurethane coating aids in this goal as well as protecting the adhesive bond. Polyurethane coatings are essential for the life of the bond as M-Bond 200 bonds are weakened by exposure to high humidity. Due to this degradation, M-Bond 200 is not recommended for installations exceeding one to two years.

Data Acquisition System

After the strain gauges are installed on the specimen, the lead-wires are wired, through connectors, Figure 32 and Figure 33, to the DAQ.

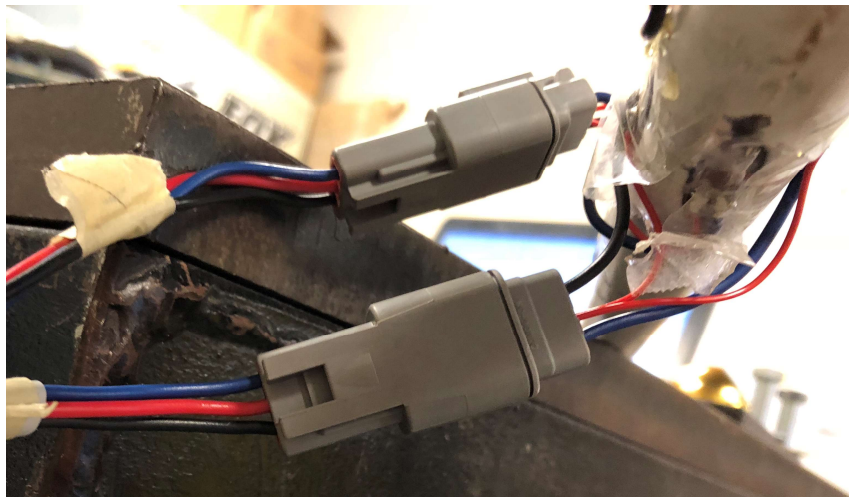


Figure 32: Lead-Wire Connected to Wire Harness



Figure 33: Wire Harness Connected to DAQ Box

Since no access to an industrial DAQ was found, the DAQ was designed around using an Arduino to process and record the strain signal. For this setup to function the strain signal first needed to be amplified.

Amplifier. There were two iterations of amplifier design. The first was an off the shelf amplifier and ADC, HX711, while the second was a custom amplifier build by Mike Potash, an electronics technician with the college of engineering at ERAU. The first amplifier was easy to implement with included example code and wiring guide. With a gain of 128 and a 24-bit ADC, the amplifier had good resolution. The issue with the HX711 amplifier was the sampling frequency allowed. At just 10Hz to 80Hz the data collection could not be made at a high enough frequency to measure the event.

Sampling Frequency. The sampling frequency needed to capture the event is derived from the event time defined in Equation (47) and the number of samples needed during the event, typically 10-20 to prevent aliasing, illustrated in Figure 34.

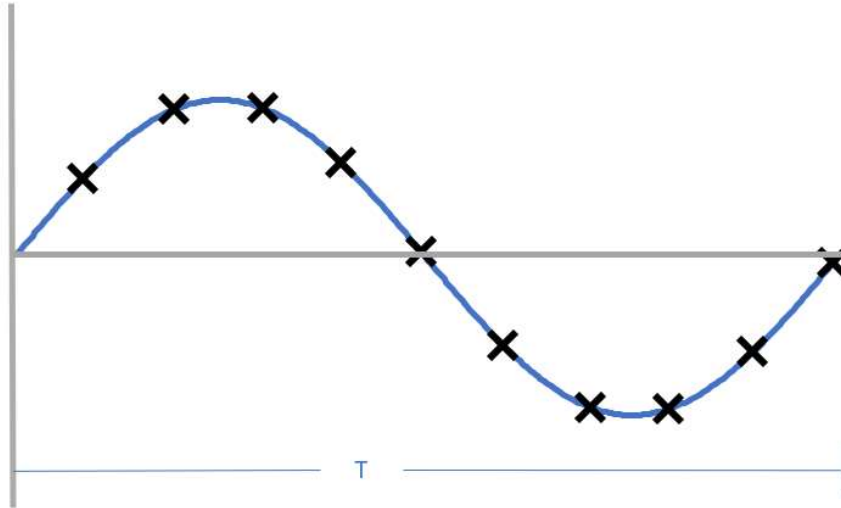


Figure 34: Samples within Event

$$v \cong \frac{\Delta x}{\Delta t} \Rightarrow \Delta t \approx \frac{\Delta x}{v} \quad (47)$$

Where, distance is x , velocity is v , and time is t .

For the defined event the target speed is 12 mph, the parking chalk is 8 inches wide therefore the event time can be calculated using Equation (47).

$$\Delta t = \frac{8in}{1} \left(\frac{1ft}{12in} \right) \left(\frac{1mile}{5280ft} \right) \left(\frac{1hr}{12mph} \right) \left(\frac{3600s}{1hr} \right) = 0.03787s \quad (48)$$

With the event time of 0.03787s, the frequency for one sample over the event would be the $1/0.03787s$ or 26.4Hz. The need for 10 to 20 samples over the event makes the sampling frequency between 264 and 528Hz. This is well above the 10 to 80Hz sampling frequency provided by the HX711 amplifier.

As there is no speedometer on the Baja vehicle a higher sampling frequency was selected, 350Hz, to provide a buffer for inaccuracy in driving speed during the data

collection. The second amplifier was designed to have a maximum sampling frequency of 1kHz, well above the design sample frequency. The amplifier contains the second half of the Wheatstone bridge built into the board as well as an antialiasing filter and zero offset balancing potentiometers for each strain gauge set. Leaving the ADC to the Arduino. The ADC on the Arduino Uno board used has a 10-bit resolution, providing 1023 units at 5 volts or 0.0049 volts per unit [63]. With the 128 gain of the amplifier this provides a resolution of 0.00003828 volts/unit or $11.15\mu\epsilon$ in the axial direction and $7.19\mu\epsilon$ in bending. At the expected strain this resolution within 5% error. The final important factor of the ADC is the reading rate, or frequency at which the board can read the signal. For the Arduino Uno the maximum reading rate is 10kHz, many factors above the selected sampling frequency [63].

With the amplifier and ADC selected, the last part of the hardware components for the DAQ is the data recording component. Many SD shield are compatible with the Arduino, however none of them can record enough channels, 3 strain gauge channels and 3 GPS channels. Thankfully the Raspberry Pi microcomputer can communicate with serial devices through the USB ports.

Raspberry Pi. The Raspberry Pi communicates over serial through python scripting. Two python scripts were created to import the data from the Arduino and the GPS respectively and write the information to two separate comma separated variables (csv) files. This was accomplished by identifying the device location as defined by the Raspberry Pi, creating a file and type, reading the serial input from the device, and writing that information into the file. For the Arduino the baud rate was the most important variable as it is defined as the signal speed in number of bits per second [64].

For the data collection from the Arduino, a count sample count is sent along with the voltage and three strain signals. Each of these five signals are an eight-byte number which makes forty-bytes per sample. With eight bits in a byte, each sample is 320 bits. To get the baud rate of the signal, at 350 samples per second, the sample size in bytes must be multiplied by the sample frequency, obtaining a minimum baud rate of 112,000 bytes per second. The baud rate is less important for the GPS data collection as the GPS device only operates at 1Hz. A baud rate of 115,200 bits per second is a standard baud rate for most micro-controllers. The Arduino scrip and two python scripts written for the DAQ system can be seen in Appendix D. Once the data acquisition system was specified and programmed, an initial data collection test was performed by collecting data from the Arduino and the GPS through the DAQ while driving the vehicle around in the parking lot near the lab.

Calibration

After the initial proof of concept did not present any errors in the DAQ system, the strain gauges were calibrated. Calibration of the three strain gauge sets requires three setups; one to apply a load in the positive z-direction of the SAE coordinate frame to create a negative moment about the x-axis to calibrate SG1 and SG3 for the strain created due to the moment at the gauge, one to apply load in the negative x-direction to create a negative moment about the z-axis to calibrate SG2 for the strain created due to the moment at the gauge, and one to apply the axial force about the positive y-axis to calibrate SG3 for the strain created due to the axial strain in the member. The calibration setup for the moment about the x-axis is shown in Figure 35 and Figure 36. The component is fixed such that the instrumented linkage is pinned at the end closest to the

strain gauges and the non-gauged link is free, creating a cantilevered beam loading application. The weight is applied to the right end of the beam, near the spherical bearing, creating a large moment at the strain gauge location.

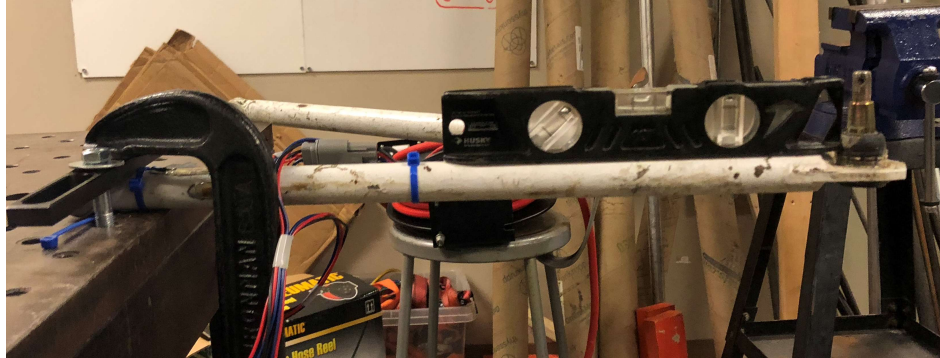


Figure 35: Strain Gauge Calibration Setup for Strain Measurement Due to Bending about the x-axis in the SAE Coordinate Frame, Component Perpendicular to the Ground

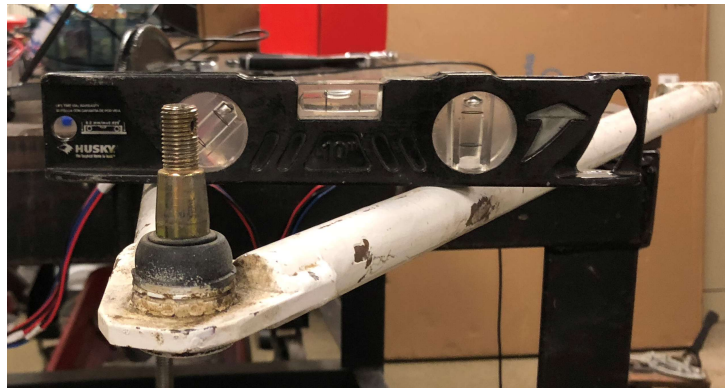


Figure 36: Strain Gauge Calibration Setup for Strain Measurement Due to Bending about the x-axis in the SAE Coordinate Frame, Component Perpendicular to the Ground and Non-Gauged Linkage Not Constrained.

The z-moment calibration setup, as seen in Figure 37, was also fixed as a cantilevered beam. The weight application can be seen in the figure.

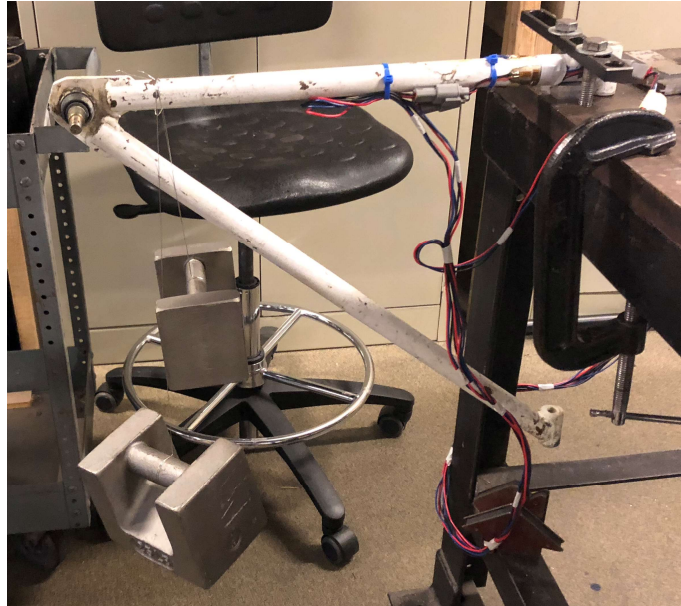


Figure 37: Strain Gauge Calibration Setup for Strain Measurement Due to Bending about the z-axis in the SAE Coordinate Frame, Component Perpendicular with Calibrated Weights Applied.

Finally, the axial load, applied in the positive y-direction, was calibrated with the setup seen in Figure 38 but applying the weight using the setup illustrated in Figure 39.



Figure 38: Strain Gauge Calibration Setup for Axial Loading, Along Positive y-axis.



Figure 39: Weight Balance for Strain Gauge Axial Calibration Utilizing Two Turnbuckles.

The turnbuckles seen in Figure 39 were used to level the weight about the component to minimize the offset of the weight from the center axis of the component. These setups were used along with a serial capture program [65] to write the strain readings from each strain gauge set to a csv file. During the calibration the component was loaded and unloaded with three to five different weights thirty times each. For the bending moment strains, in the x- and z-directions, five different weights were applied while only three weights were applied in the axial direction, due to the need for load variation and a limited number of weights. The mean and standard deviations of the thirty measurements taken during each weight application are presented in Table 2, Table 3, and Table 4.

Table 2

Calculated and Measured Strain Due to Load Applied to Cantilevered Beam Creating a Moment of Bending about the x-axis, Mb_x , at the Strain Gauge as Seen in Figure 35. Mean and Standard Deviation of 30 Measurements Presented for Each Weight.

Weight (lbs)	SG1 measurement ($\mu\epsilon$)		Expected Strain at SG1($\mu\epsilon$)	SG3 measurement ($\mu\epsilon$)		Expected Strain at SG3($\mu\epsilon$)
	\bar{X}	s		\bar{X}	s	
10	96.4	2.6078	-97.7	-110.5962	4.7836	-102.2
20	194.0361	1.7944	-195.4	-236.5254	3.9121	-204.4
25	241.7861	1.9793	-244.2	-296.8024	3.0161	-255.5
35	337.4406	5.6248	-341.9	-414.2892	7.0744	-357.8
45	437.7361	2.5962	-439.5	-540.4107	3.7807	-460.0

Table 3

Calculated and Measured Strain Due to Load Applied to Cantilevered Beam Creating a Moment of Bending about the z-axis at the Strain Gauge as Seen in Figure 37.

Weight (lbs)	SG2 measurement ($\mu\epsilon$)		Expected Strain at SG2($\mu\epsilon$)
	\bar{X}	s	
10	95.4486	1.8117	-97.2
20	192.1500	2.8415	-194.4
25	243.1733	1.7690	-243.1
35	342.642	3.0068	-340.3
45	440.7454	3.2138	-437.5

Table 4

Calculated and Measured Strain Due to Axial Load, F_y , Applied to Component as Seen in Figure 38.

Weight (lbs)	SG3 measurement ($\mu\epsilon$)		Expected Strain at SG3($\mu\epsilon$)
	\bar{X}	s	
100	13.4578	0.8692	34.06
150	21.7400	1.4967	51.09
194	26.7907	1.3206	66.11

All the calibration datapoints are used in a linear regression model to create one linear equation of calibration for each strain gauge. The linear calibration are shown in Equations (49), (50), and (51).

$$\varepsilon_{F_y} = 2.2512 * \varepsilon_{SG3} + 3.8787 \quad (49)$$

$$\varepsilon_{Mb_x} = -1.0039 * \varepsilon_{SG1} - 1.2337 \quad (50)$$

$$\varepsilon_{Mb_z} = -0.9827 * \varepsilon_{SG2} - 4.1858 \quad (51)$$

The regression lines are highly correlated with low MSE, RMSE, and high R^2 values as seen in Table 5.

Table 5

Descriptive Statistics of Linear Regression Model for Strain in Each Direction.

Calibrated Direction	MSE	RMSE	R^2
ε_{F_y}	9.8994	3.1463	0.9442
ε_{Mb_x}	11.5114	3.3928	0.9992
ε_{Mb_z}	6.9273	2.6320	0.9995

With the vehicle instrumented and calibrated the experiment could be run but first the FEM setup was checked with the calibration loads and constraints to ensure similar strain values as the ones in the gauged locations.

Finite Element Model

The similar loads and constraints were used in the FEM as the calibration to verify the material properties, the mesh, and the model software. First the ε_{Mb_x} load case

was tested; a cantilevered beam, modeled with fixed tri-axial rotation at the bushing-chassis attachment joint, with the load applied as a point load at the same location as the calibration, 15.8 inches from the axis of fixed rotation. Next the axial load case, ε_{F_y} , was tested in the FEM with fixed tri-axial rotation at the bushing-chassis attachment joint and the axial load applied at the far end of the bar. Finally, the $\varepsilon_{M_{b_z}}$ load case was tested, similarly to the $\varepsilon_{M_{b_x}}$ conditions but the force is applied at 16.425 inches from the axis of fixed rotation. This last load case under 45 lb condition is modeled in Figure 40, where the strain in the component at the location of SG2 is $327.4 \mu\varepsilon$, expectedly lower than the calibration value due to the fixed location being farther from the strain gauge. These results are similar enough to the calibration to continue with the model.

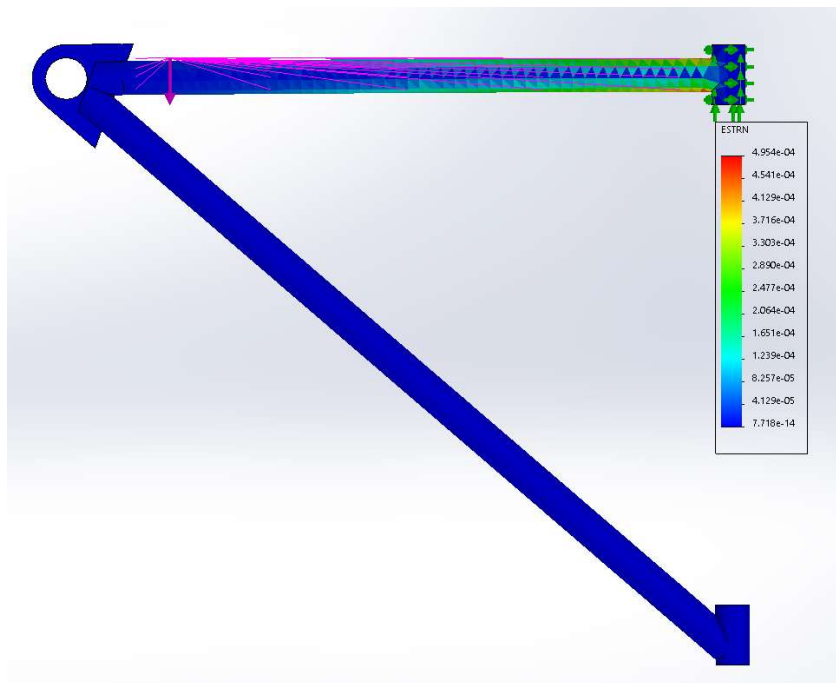


Figure 40: Strain Calibration in Bending about the z-axis FEM.

The FEM was then constructed with the same constraints as the loading condition of the event, a beam pinned at both ends. The constraints at both bushing-chassis

attachment points were pinned using a “on cylindrical faces” constraint where the cylinder was not allowed to translate in either direction only rotate about its axis. While the constraint at the spherical bearing mount was free to translate along the y-axis and to rotate about the z-axis but not to translate along the z-axis using the “on cylindrical faces” constraint.

With the model constraints verified, the strain gauges installed and calibrated, the component re-installed in the vehicle, and the DAQ wired the data collection was performed over the event eleven times. The data produced from those runs are analyzed in the Chapter IV.

Chapter IV

Results

On December 13th, 2018, 11 runs were collected utilizing the hardware and software outlined above. Each run was recorded to a csv file and imported into Matlab utilizing the Thesis_Data_Collection_Code_Setup_v2.m file. First running sub-code Calibration_Data_Collection_Analysis.m which defines the constants, imports the calibration data, and calibrates the data collection readings to the known applied strain, as discussed above.

Data Analysis

The eleven runs of data were then imported from two separate files each, the Arduino files, and the GPS files. As discussed above it was not possible to collect both readings into the same datafile. The GPS records velocity in knots, converted here to m/s.

A check was performed to ensure all values were between 0 and 1023, as any value outside this range would be a bad datapoint due to the utilization of the Arduino analogue pins, which read from this range [63]. All data values were found to be within the acceptable range.

As discussed above, the data collection frequency was set to 350 Hz for the Arduino data collection. This frequency was met however, some intervals were not received by the raspberry pi due to the collection maxing out the central processing unit (CPU). The missed data and its effect on the ability of the run to capture the event is illustrated in Table 6 and Table 7.

Table 6

Missed Data During Experimental Data Collection, Illustrated as a Percentage of the Total Data Collection with the Effect on the Event.

Run	Collection Points Missed	Number of Collection Points Total	Percentage Missed	Effect on Data About Run
1	0	19885	0%	None
2	471	20213	2.33%	None
3	378	17057	2.22%	None
4	205	17219	1.19%	None
5	267	18844	1.42%	None**
6	621	17146	3.62%	None*
7	225	16567	1.36%	None
8	0	19063	0%	None
9	450	16215	2.78%	None
10	0	14259	0%	None
11	0	11295	0%	None

*Missing Data Occurs One Second After the End of the Event.

**Missed Data Occurs Two Seconds After the End of the Event.

Special attention should be paid to the data about the event for these two runs to verify the event was captured and the missing data does introduce error

Table 7

Identification of Missing Datapoints and Location of Event Endpoint

Run	Missing Sections	Event Endpoint	Notes
2	12,722-13,139	1,197	-
3	11,065-11,443	1,260	-
4	13,274-13479	1,374	-
5	1,848-2,115	1,278	less than 2 seconds after event
6	1,502-1,673 & 16,512-16,962	1,251	less than 1 second after event
7	8,280-8,505	1,247	-
9	15,764-16,214	1,107	-

The event endpoint was identified at the end of run 11 when the vehicle was driven to the far side of the parking chalk while the instrumentation was still gathering

data, providing GPS data for the location just after the parking chalk. This data was then utilized to develop a line of the approximate parking chalk location, as seen in Figure 41.



Figure 41: GPS Waypoints of Data Collection 11, Identifying the GPS Location of the Parking Chalk

All the preceding runs were conducted in an oval pattern as seen in Figure 42.



Figure 42: GPS Waypoints, Data Collection 4, Overlay with Google Maps

The Matlab “polyxpoly” function was utilized to return the location of intersection of each GPS dataset with the chalk line and the index of the closest GPS

datapoint from each dataset before the parking chalk location, seen in Figure 60-Figure 70 in Appendix .

The next step is GPS and Arduino data concatenation. Differences in sampling frequency introduced complexity to this step, as the GPS data is collected at 1Hz and the Arduino data is collected at 350Hz. To correlate the datasets, the “time.time” function in the python scripts was recorded in each dataset. This function is supposed to record to the milliseconds however, in the Arduino (ARD) datasets and GPS data collection 1 the function only recorded to the seconds. Therefore, the time offset between the first data point in the Arduino datasets and the GPS datasets must be calculated. To calculate the offset between the two datasets, the GPS first data value was taken as it should also be noted that since GPS data collection 1 only recorded to the second, the concatenation could be off by up to 0.999s. For all other datasets the concatenation resolution relies on the logic used to identify the offset between the two datasets; identifying the first datapoint in the ARD dataset that changed into the next second and adding the known time elapsed since the beginning of the data collection, based on the frequency collected and the number of datapoints to that location, this sum is subtracted from the first GPS time to obtain the offset between the two datasets. This offset could be off by just under the timestep of the ARD data collection which is 0.002857 seconds. These errors should be kept in mind when analyzing the location of the parking chalk contact within the Arduino recorded strain gauge data.

Table 8

GPS and Arduino (ARD) Resolution of the "time.time" Function Recorded on the Raspberry Pi

Run	GPS	ARD
1	seconds	seconds
2	millis	seconds
3	millis	seconds
4	millis	seconds
5	millis	seconds
6	millis	seconds
7	millis	seconds
8	millis	seconds
9	millis	seconds
10	millis	seconds
11	millis	seconds

Data Filtering. The initial data collection of strain, through ARD, was very noisy because it contained frequency energy from other parts of the vehicle system. The structural response of the component of interest in the study and therefore only the frequencies associated with this component and the system it is within, the suspension, are of interest. The natural frequency of the suspension is 1.03Hz due to the system design. To capture only the frequency energy due to the suspension frequency a FFT is performed through the Matlab `fft()` function and all frequency energy above 10Hz is removed. 10 Hz was chosen due to the potential for aliasing of the signal, discussed in the Data Analysis section of the Literature Review. The resulting change in signal can be seen in Figure 43: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT

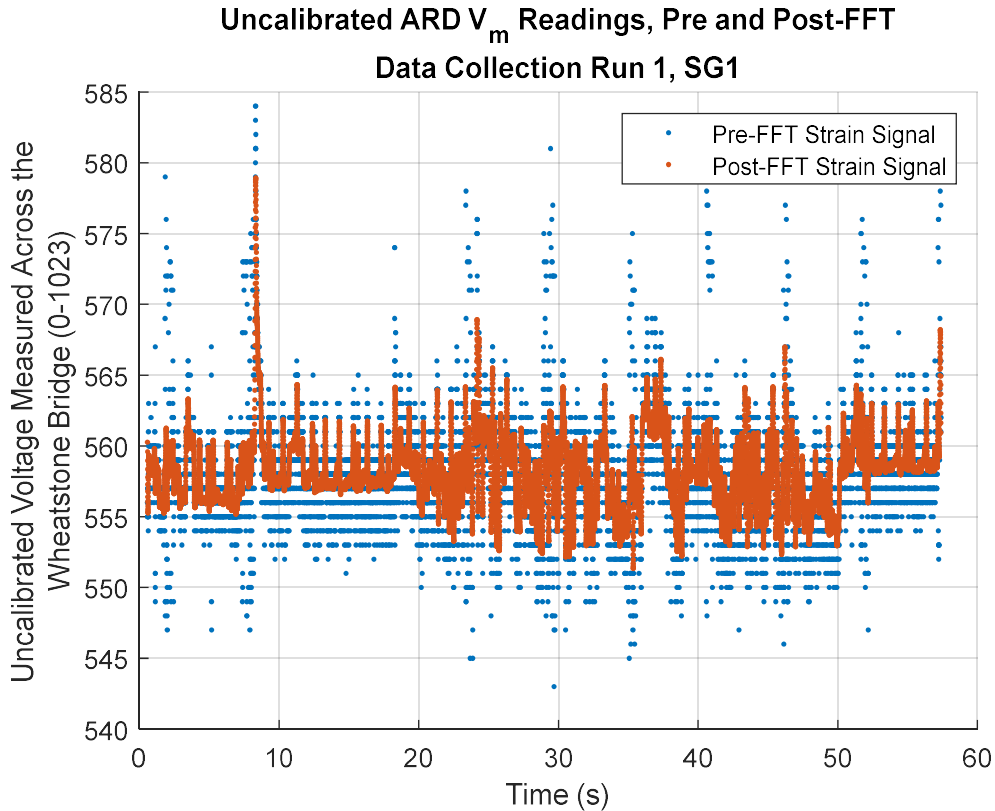


Figure 43: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 1, SG1.

The remaining signals can be seen in Appendix E in Figure 71: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 1, SG1. to Figure 103: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 11, SG3.

Zero-Offset. After the FFT filtering, the GPS and ARD data are concatenated and the end of event location was identified for each dataset as previously discussed. Finally, a section, 1050 datapoints before and after the event, was taken for each dataset. This allows the event to be focused on and the plots of the data to more clearly illustrate where the event is occurring. After which point, calibration steps were implemented. Starting

with finding the zero-offset for the ARD V_{m_analog} for each dataset. This identification was done statistically using the first 100 datapoints of each run at each reading, accuracy verified with standard deviation, as seen in Table 9.

Table 9

ARD V_{m_analog} Reading Zero-Offset Value Identification Using Mean of First 100 Datapoints, Accuracy Verified with Standard Deviation.

Run	Strain Gauge Set	$\bar{X}_{1 \rightarrow 100}$	$S_{1 \rightarrow 100}$	Noted Deviations
1	SG1	558.1404	1.1378	-
1	SG2	412.4751	1.0939	-
1	SG3	539.7049	1.1647	-
2	SG1	559.1575	0.2879	-
2	SG2	412.7594	0.2432	-
2	SG3	541.1236	0.7337	-
3	SG1	560.8382	0.3352	-
3	SG2	414.1805	0.4618	-
3	SG3	548.6908	0.4651	-
4	SG1	561.7106	0.6236	*Datapoints: 160 to 300 Used
4	SG2	415.1533	0.5500	
4	SG3	552.0968	0.5321	
5	SG1	562.2870	0.6743	-
5	SG2	414.3316	0.7062	-
5	SG3	553.7315	0.4661	-
6	SG1	563.0007	0.8728	-
6	SG2	414.7958	0.8228	-
6	SG3	557.2933	0.7151	-
7	SG1	558.8851	3.7108	-
7	SG2	410.6911	3.5775	-
7	SG3	553.2896	3.6791	-
8	SG1	557.0510	0.2403	-
8	SG2	409.0112	0.2258	-
8	SG3	555.0790	0.2666	-
9	SG1	560.7682	0.7084	*Datapoints: 240 to 440 Used
9	SG2	411.8738	0.6024	
9	SG3	557.4614	0.9760	
10	SG1	557.0309	0.2539	-
10	SG2	408.2965	0.3190	-
10	SG3	553.8175	0.2696	-
11	SG1	556.7511	0.2697	-
11	SG2	409.7545	0.3351	-
11	SG3	553.7113	0.2595	-

*Beginning ARD V_{m_analog} readings contained excitations, datapoint range used after signal flattened out again.

Strain Calculation. The mean zero-offset values for each strain gauge measurement, V_{m_analog} , were subtracted from the corresponding array of strain gauge measurements, leaving the change in amplified voltage, V_{m_amp} . This strain delta reading was then used to find the strain value, ε_{SG} , for each array. First by calculating the voltage measured based on the DAQ setup, using Equation 52.

$$V_m = \frac{\left(\frac{V_{m_amp}}{gain_ard}\right) V_{ard}}{gain_amp} \quad (52)$$

Where, V_{ard} is the Arduino maximum voltage range of 5 volt, $gain_ard$ is the value 1023, due to the 10-bit analog to digital conversion, and $gain_amp$ is the gain of the amplifier, 128. The reading from each strain gauge set, V_{m_analog} , is some number between 0 and 1023 due to the ADC conversion in the Arduino where this range is a representation of 0 to 5 volts. By dividing by the range and multiplying by the voltage the measurement was converted from unitless to in units of volts. Finally, the voltage reading needs to be divided by the gain from the amplifier to return it to the initial measurement scale, before amplification, at the Wheatstone bridge of the strain gauge set. The strain in each gauge set can be calculated through this bridge voltage measurement using Equation 53 for SG1 and SG2 and Equation 54 for SG3.

$$\varepsilon = \frac{V_m}{V_b} * \frac{2}{k} \quad (53)$$

$$\varepsilon = \frac{V_m}{V_b} * \frac{4}{k(1 + \nu)} \quad (54)$$

Recall from Equation (43), the strain in Equation (54) is the combined axial and bending strain at the location of SG3. The bending strain in SG3 and SG1 are measured

about the same axis however, the strain gauges are not in exactly the same location. Therefore, the strain measurement in SG3 must be projected to the location of SG1. Based on the component constraints of a double pinned beam in bending, the moment about the beam is parabolic, similar to a simply-supported beam under uniform load [56].

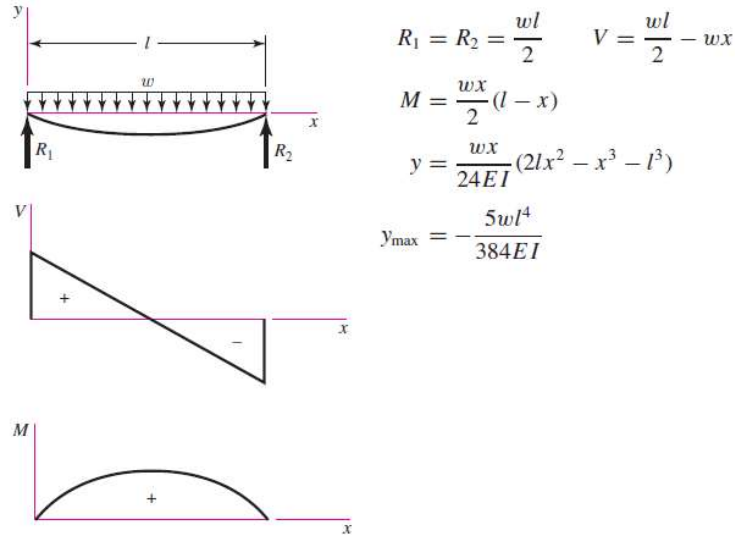


Figure 44: Simply-Supported Beam Under Uniform Loading [56].

Therefore, the strain calculated from Equation (54) must be multiplied by a factor accommodating the difference in positions of SG3 to SG1, in order to subtract the correct amount of bending strain from the SG3 measurement. This factor is defined in Equation 55.

$$factor = \frac{(L_{beam} * x_1 - x_1^2)}{(L_{beam} * x_3 - x_3^2)} \quad (55)$$

Where:

L_{beam} = Total length of the beam from pin to pin.

x_1 = Distance from SG1 to the spherical bearing pin location.

x_3 = Distance from SG3 to the spherical bearing pin location.

Making the combined strain equation for SG3:

$$\varepsilon = \frac{V_m}{V_b} * \frac{4}{k(1 + \nu)} * factor \quad (56)$$

To calculate the axial strain along the calculated strain measured from SG1 is subtracted from the combined calculated strain measured from SG3, since the signs of these strains are opposite, as expressed in Equations (35) and (43), the strains are added. Each calculated strain reading is then multiplied by the corresponding linear regression calibration function, displayed in Equations (49), (50), and (51), to obtain the calibrated strain value over the event for each strain gauge during each data collection. A smaller window of data was then identified to make the event location clearer. The data collection with the clearest parking chalk contact location is run 10, seen in Figure 45. Due to the inherent noisiness of strain gauge readings, some events are harder to see clearly, all events are shown in Figure 104 to Figure 114.

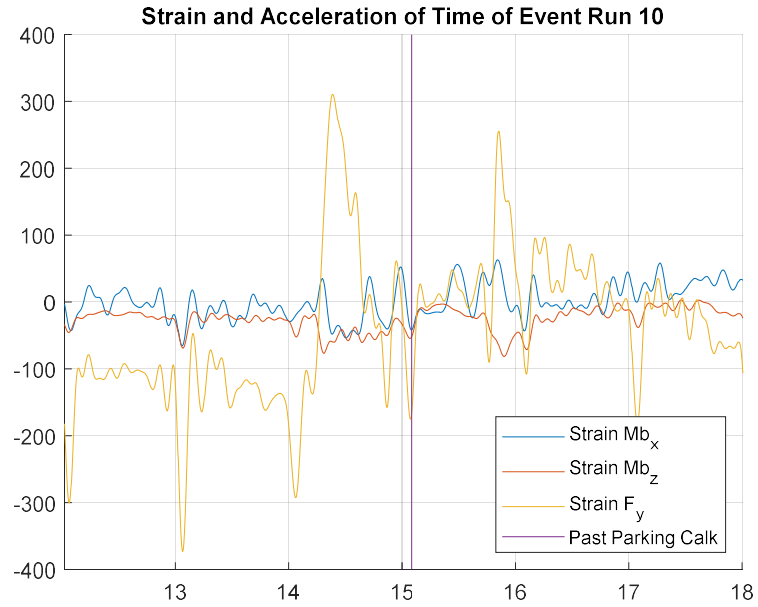


Figure 45: Strain Measured by Each Gauge During Event Run 10 with Parking Chalk Location Estimation and Velocity.

Using Figure 104 to Figure 114 to identify the max strain location of each data collection event, except runs 6 and 7 due to lack of clear contact with the parking chalk. From this datapoint the maximum strain at each gauged location for each run can be calculated by taking the value of maximum strain and subtracting the offset of signal drift. The standard deviation of this maximum strain measurement is large due to the variation in velocities for each run, seen in Table 10.

Table 10

Velocity of Vehicle at Event Contact Location in Meters per Second and Miles per Hour

Run	Velocity at Event (m/s)	Velocity at Event (mph)
1	5.0706	11.3426
2	5.0501	11.2968
3	5.3308	11.9247
4	5.3261	11.9142
5	5.3630	11.9968
8	4.7533	10.6328
9	5.4027	12.0856
10	6.0917	13.6269
11	6.2940	14.0792

Maximum strain values can be normalized to eliminate the error induced by the difference in velocity between events using momentum. Momentum is equal to the mass of an object times its velocity. Since the mass is unchanged between data collections, the maximum strain values of each event can be multiplied by the proportion of the velocity of one singular event over each event respectively. As the target speed for the experiment was 12 mph, the velocity of the event in run 5 is used to normalize maximum strain values of the event in each run. After accounting for this potential error comparing the measurements the mean, standard deviation, 95% confidence interval, and kurtosis were calculated for the 9 events captured during experimentation, see Table 11.

Table 11

Maximum Strain Due to Event at each Strain Gauge Location, Type of Strain Specified, with Mean, Standard Deviation, and 95% Confidence Interval of all 9 Data Collection Events

Strain Type	Strain Location	\bar{X} ($\mu\epsilon$)	s ($\mu\epsilon$)	95% Confidence Interval	Kurtosis
ϵ_{Mb_x}	SG1	25.8381	15.3311	56.5	0.201
ϵ_{Mb_z}	SG2	-25.9104	-18.6336	-63.18	-0.456
ϵ_{F_y}	SG3	266.3615	107.8677	482.10	-0.9773

Using the 95% confidence interval maximum strain value, the loading condition in the linear FEM was validated. Since the beam is simply supported, there should be no strain about the bending about the x-axis, ϵ_{Mb_x} . The strain present in the SG1 measurement can be attributed to the binding of the suspension system, at the shock and in the joints, in jounce. This strain is ignored in the FEM validation because the exact source of binding or resistance in jounce is not known and therefore cannot be modeled accurately.

FEM Validation

To obtain the forces acting on the component from the wheel, a remote unit load was applied in the y- and z-axis, separately, to the attachment point of the component to the spherical bearing mount. The model was simulated twice, once for each unit load, to obtain the resultant strain at the locations of SG2 and SG3. This data was used to develop a four by four matrix of strain readings at the respective gauge locations for the respective unit load. Strain results of constraints and unit load applied in the $-F_y$ direction can be seen in Figure 46: Simply Supported Beam Constraints with one Unit Load (N) Applied in the Axial Direction ($+F_y$ FEM Coordinates, $-F_y$ SAE Coordinates).

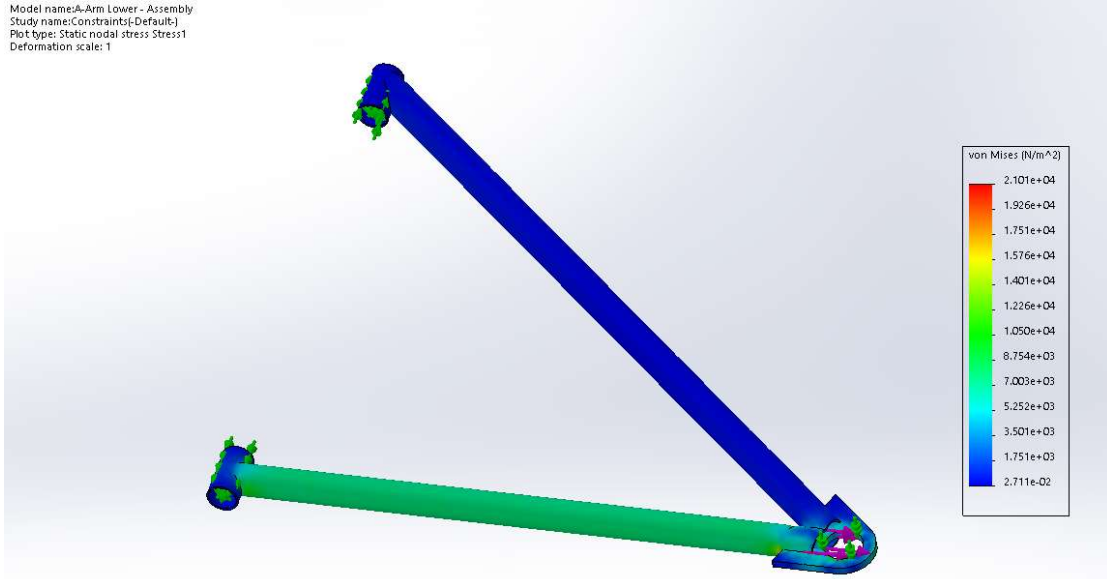


Figure 46: Simply Supported Beam Constraints with one Unit Load (N) Applied in the Axial Direction (+F_y FEM Coordinates, -F_y SAE Coordinates).

The matrix of strains per unit load in the F_y and F_x directions respectively, compliance matrix, can be seen in Equation 57, where the columns are ϵ_{Mb_z} over ϵ_{F_y} due to F_x and F_y respectively.

$$\begin{bmatrix} \epsilon_{Mb_z} \\ \epsilon_{F_y} \end{bmatrix} = \begin{bmatrix} 0.011195 & 0.0021 \\ 0.039495 & 0.03602 \end{bmatrix} \begin{bmatrix} F_x \\ F_y \end{bmatrix} \quad (57)$$

This matrix represents the linear relationship between force in F_x and F_y to the strain about ϵ_{Mb_z} and ϵ_{F_y} for the FEM. Knowing the strain values ϵ_{Mb_z} and ϵ_{F_y} due to the loading condition tested, the forces, F_x and F_y, can be calculated by multiplying a matrix of ϵ_{Mb_z} over ϵ_{F_y} by the inverse of the compliance matrix in Equation 57.

$$\begin{bmatrix} 0.011195 & 0.0021 \\ 0.039495 & 0.03602 \end{bmatrix}^{-1} = \begin{bmatrix} 112.4555 & -6.5563 \\ -123.3046 & 34.9511 \end{bmatrix} \quad (58)$$

ε_{Mb_z} and ε_{F_y} are known from Table 11, making the matrix multiplication:

$$\begin{bmatrix} -112.4555 & 6.5563 \\ 123.3046 & -34.9511 \end{bmatrix} * \begin{bmatrix} -63.18 \\ 482.10 \end{bmatrix} = \begin{bmatrix} -10266N \\ 24640N \end{bmatrix} \quad (59)$$

This loading condition was simulated in the FEM and the strain values for ε_{Mb_z} and ε_{F_y} were $-61.95\mu\varepsilon$ and $468\mu\varepsilon$ respectively, or 2% and 3% error respectively.

With the loading condition fully defined, the component could be evaluated for failure based on maximum yield stress (σ_{max}), 0.709 GPa, and total fatigue non-reversed loading cycles to failure based on Von Mises stress in the member. The total loading cycles are calculated based on ASME elastic modulus S-N curve, with a non-reversing load. The tested event condition was evaluated first, and the test condition scaled for maximum vehicle velocity (v_{max}), approximately 30mph.

Table 12

FEM Results, Maximum Stress in the Member and von Mises FOS, from Event Loading Condition, Scaled Loading Condition for Maximum Velocity, v_{max} , and Scaled Loading Condition for Maximum Mass and Velocity, v_{max} and W_{max} .

Condition	F_x	F_y	σ_{max}	Cycles to Failure
Tested	-10.26 kN	24.64 kN	0.082 GPa	48,470
v_{max}	-25.66 kN	61.6 kN	0.2114 GPa	777

The model does not structurally yield under the test condition, even when amplified for a heavier driver and a higher vehicle speed. Therefore, the tested loading condition did not cause the failure seen on the right fore lower control link, Figure 20. However, the loading condition is a repeated load case and if this component were to complete more than one competition or more than one design season, which is highly likely, the FOS should not be ignored and the member should be re-designed to accommodate for that failure and reduce the likelihood of the failure exhibited in Figure 20.

Component Re-Design. The initial design load case did not predict any axial loading in the member. Since the largest stress in the member was axial, the member was re-designed with this condition as the main focus. From the topology optimization section, it is noted that topology optimization turns design constraints into a minimization problem. For the design in question, the member must fit within the existing envelope of the member. Meaning, the re-designed member must be the same length as the original member, must mount to the spherical bearing mount and the bushing mount without interference, and must accommodate the aft member of the component in assembly. With these constraints, changing the cross-sectional shape of the member is the simplest solution. For axial load, the axial strain in the member is determined in Equation (21) by the load and the cross-sectional area, A . For a hollow tube A is calculated by Equation (24), and for the member in question is equal to 0.1909 square inches. While a square tube with the same wall thickness and outer length would have a cross-sectional area of 0.2431 square inches. Increasing the maximum axial force from 87.3 kN to 111.2 kN. Also increasing the maximum bending moment from 487 N to 11 kN, and the torsional

load from 109.9 kN to 624.2 kN. While only increasing the weight by 0.23lbs. Using this as a starting point, the component was re-designed to be a square tube with a one-inch outer length and 0.065-inch wall thickness. This component failed the FOS test in the full FEM until the wall thickness was increased to 0.08 inches, increasing the weight by 0.53-lbs. Seeing as this weight increase was not ideal, as the unsprung mass of the suspension system directly affect suspension performance, the component was lightened by removing material from the top and bottom sections, as the bending moment about this axis is low.

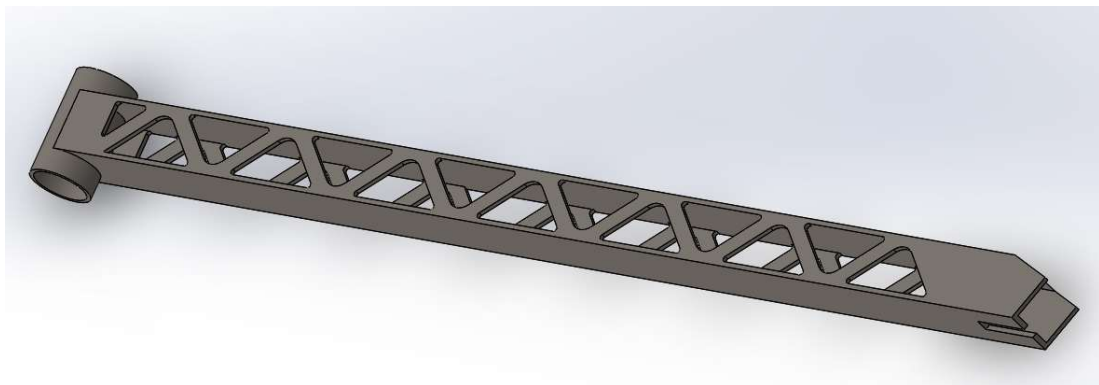


Figure 47: Re-Designed Suspension Lower Fore Link.

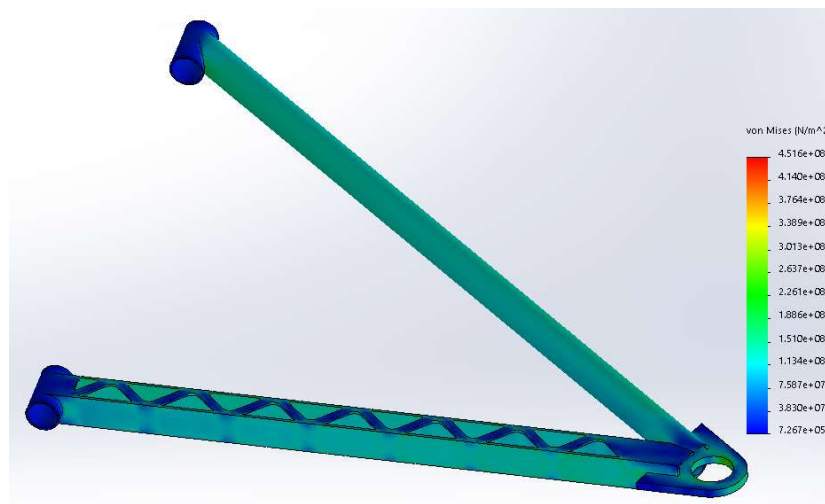


Figure 48: FEM Stress Results in the Re-Designed Component Under Tested Loading Condition.

The re-designed member, seen in Figure 48, was assembled into the lower suspension Solidworks assembly in place of the original member. The FEM was performed under the same conditions as illustrated in Table 12. Results for the re-designed member can be seen in Table 13.

Table 13

FEM Results Re-Designed Member, Maximum Stress in the Member and von Mises FOS, from Event Loading Condition, Scaled Loading Condition for Maximum Velocity, v_{max} , and Scaled Loading Condition for Maximum Mass and Velocity, v_{max} and W_{max} .

Condition	F_x	F_y	σ_{max}	Cycles to Failure
Tested	-10.26 kN	24.64 kN	0.1014 GPa	$1e^6$
v_{max}	-25.66 kN	61.6 kN	0.2523 GPa	333,900

For the tested event the re-designed member can withstand the tested event for infinite life. The second two cases do not come near the yield point of the material, nor are these loading conditions fatigue loading conditions, they are abuse loading conditions.

Since the new member does not fail infinite life testing or yield during abuse cases, the member can be recommended for implementation after one final check. Remembering, there was strain due to bending about the x-axis, ϵ_{Mb_x} , during the experimental data collection. Due to the lightening holes about this strain direction, the member should be checked for this loading condition. If the measured strains are resolved into forces, based on cantilevered beam constraint, all three forces can be resolved, and a

FEM can be run under this loading condition. The resolved force for the corresponding strain measurement is illustrated in Table 14.

Table 14

Resolved Force, Based on Corresponding Strain Measurement Under Cantilevered Beam Constraints.

Force Direction	Measured Strain ($\mu\epsilon$)	Resolved Force (N)
F_z	56.50	-21.9
F_x	-63.18	-23.5
F_y	482.10	6298.5

The resolved FEM of the component was simulated, and the yield FOS was determined to be 2. Stress in the member is below the yield stress by almost a factor of ten, 0.95 GPa. Stress Results from the FEM can be seen in Figure 49.

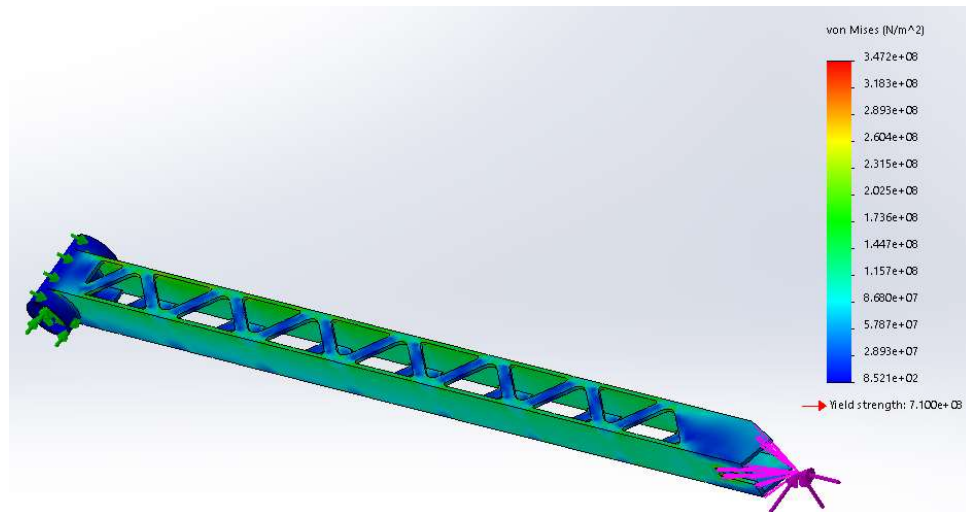


Figure 49: Component FEM Under Cantilevered Beam Constraints and Loads.

The final design was verified through the FEMs, with a final design weight increase of 0.04-lbs.

Chapter V

Discussion, Conclusions, and Recommendations

An experiment to ascertain the loading conditions of the front suspension fore control link was designed and executed. Using Omega strain gauges, a custom amplifier, an Arduino and a Raspberry Pi, the data was reliably and accurately obtained. Strain gauge installation quality was verified through the calibration with standard deviations less than 5% of the measured value and linear regression calibration equations within four microstrain, below the minimum resolution of the DAQ system. Event data has a low kurtosis, meaning the measurements did not produce significant outliers, implying a low variation between the measurements. To account for the potential measurement errors in the system, a 95% confidence interval was used for the FEM validation of strain in the member. Using the linearity of the FEM, the forces acting on the component at the spherical bearing were resolved. After which, the member was re-designed to withstand this loading condition for structural strength and infinite fatigue life only increasing the weight of the assembly by 0.04-lbs.

The design change analyzed here is recommended for implementation on this year's Baja vehicle as it will reduce the likelihood of component failure. However, the event tested in this thesis was discovered not to be the event that caused the structural failure of the right lower fore suspension arm on the Baja vehicle. It should be noted that the failure may still occur, though the design change makes this less likely.

In the future it is recommended to buy a wheel force transducer to measure the exact wheel load and validate not only this component FEM for the event tested but the entire suspension over the full loading conditions of the competition. This was not

feasible for the DOE presented as wheel force transducers are expensive and collecting data at competition requires an expensive data acquisition system to be mounted in the vehicle, made more expensive as it must be waterproofed to survive competition environmental conditions.

Fatigue failure is rare on the Baja vehicle due to the short design life cycle. Therefore, it is not recommended to instrument different components on the vehicle with strain gauges to obtain strain life histories. The only recommended strain gauging would be to resolve specific model validation and load case questions on failing components as completed in this thesis.

The data collected during the course of this thesis work can and should be utilized by the Baja team in the future to aid in any re-designs of the front suspension system. The strain values along with the linearity of the FEM can be utilized, as shown in the FEM validation section of the results, to resolve the remote forces at any location. Resolving the forces at the wheel center is recommended as this load can then be applied to the upper a-arm linkage. It is recommended for any future re-design to follow these steps:

1. Use the original FEM of the front suspension lower control arm assembly to resolve the forces at the wheel; following the steps discussed in the FEM validation section of the results.
2. As long as the tires and the shock remain the same, this force is accurate for the tested loading condition at 12mph. If the tires change (structure or pressure) or the shock changes, which is recommended, the loading condition is a good place to start but should be modified based on the changes in stiffnesses of the suspension and the tires.

References

- [1] F. Huizinga, R. Van Ostaijen and A. Van Oosten Slingeland, "A Practical Approach to Virtual Testing in Automotive Engineering," *Journal of Engineering Design*, pp. 33-47, 2002.
- [2] "About Baja SAE Series," 2019. [Online]. Available: <https://www.sae.org/attend/student-events/baja-sae-tennessee-tech/about>.
- [3] C. C. Chu, "Critical Plane Fatigue Analysis of Various Constant Amplitude Tests for SAE1045 Steels," in *International Congress and Exposition*, Detroit, MI, 1994.
- [4] SAE International, *Surface Vehicle Recommended Practice*, SAE Standard J670, 2008.
- [5] T. E. Putras, S. Abdullah, D. Schramm, M. Z. Nuawi and T. Bruckmann, "Generating Strain Signals Under Consideration of Road Surface Profiles," *Mechanical Systems and Signal Processing*, Vols. 60-61, pp. 485-497, 2015.
- [6] U. V. Grubisic, "Determination of Load Spectra for Design and Testing," *International Journal of Vehicle Design: The Journal of the International Association for Vehicle Design*, vol. 15, no. 1-2, pp. 8-24, 1994.
- [7] A. S. Sener, "Finite Element Based Vehicle Component Fatigue Life Assessment According to a Customer Usage Profile," *Materials Testing*, vol. 56, pp. 198-207, 2014.
- [8] R. A. Lund and K. H. Donaldson Jr., "Approaches to Vehicle Dynamics and Durability Testing," in *SAE Technical Paper Series*, 1982.
- [9] D. C. Montgomery, *Design and Analysis of Experiments*, Fourth ed., New York, NY: John Wiley & Sons, 1997.
- [10] A. Dean and D. Voss, *Design and Analysis of Experiments*, New York, NY: Springer, 1999.
- [11] K. Potter, "Structural Durability and Integrity in Vehicle Design," in *European Conference on Fracture: Fracture of Materials and Structures from Micro to Macro Scale*, Munich, Germany, 2010.
- [12] P. R. Pawar and M. R. Saraf, "Measurement of Road Profile and Study its Effects on Vehicle Durability and Ride," SAE International, Pune, India, 2009.

- [13] M. French and V. F. Mannino, "Alternative Accuracy Measurements for Road Simulation," in *International Congress and Exposition*, Detroit, MI, 1998.
- [14] W. G. Ferriera, F. Martins, S. Kameoka, A. S. Salloum and J. T. Kaeya, "Structural Optimization of Automotive Components Applied to Durability Problems," SAE International, 2003.
- [15] R. C. Rice, Ed., *Fatigue Design Handbook*, Second ed., Warrendale, PA: Society of Automotive Engineers, 1988.
- [16] S. Keil, "On the Strain Gage's 50th Jubilee - A Review of its Evolution and of 33 Years Strain Gage Production at Darmstadt," *RAM*, vol. 4, no. 2, pp. 39-48, 1988.
- [17] S. Keil, *Technology and Practical Use of Strain Gages: With Particular Consideration of Stress Analysis Using Strain Gages*, Berlin: Wilhelm Ernst & Sohn Verlag für Architektur und technische Wissenschaften, 2017.
- [18] National Instruments, *Strain Gauge Measurement - A Tutorial*, 1998.
- [19] M. E. Tuttle, "Fundamental Strain-Gage Technology," in *Manual on Experimental Methods for Mechanical Testing of Composites*, R. L. Pedleton and M. E. Tuttle, Eds., Dordrecht, Springer, 1989, pp. 17-26.
- [20] C. H. Wheatstone, "An Account of Several New Instruments and Processes for Developing the Constants of a Voltaic Circuit," *Philosophical Transactions of the Royal Society, Series B*, vol. 33, pp. 303-327, 1843.
- [21] C. L. Nachtigal, *Instrumentation and Control*, M. D. Martin, Ed., New York, NY: John Wiley & Sons, 1990.
- [22] J. C. Kaimal and J. J. Finnigan, *Atmospheric Boundary Layer Flows: Their Structure and Measurement*, New York, NY: Oxford University Press, 1994.
- [23] J. K. Reichert and J. P. Landolt, "Digital and Analog Filters for Processing Impact Test Data," in *SAE Technical Paper Series*, Dearborn, 1981.
- [24] M. Meyer. US Patent US2631199A, 1950.
- [25] R. E. Walpole, R. H. Myers, S. L. Myers and K. Ye, *Probability & Statistics for Engineers & Scientists*, Eighth ed., Upper Saddle River, NJ: Pearson Prentice Hall, 2007.
- [26] G. J. Hahn and W. Q. Meeker, *Statistical Intervals: A Guide for Practitioners*, New York: John Wiley & Sons, 1991.

- [27] "Linear Regression," [Online]. Available: https://www.mathworks.com/help/matlab/data_analysis/linear-regression.html.
- [28] "Root Mean Squared Error; RMSE," in *3D A-to-Z Encyclopedic Dictionary*, Jefferson, NC, McFarland, 2012.
- [29] M. Taraji, P. R. Haddad, R. I. Amos, M. Talebi, R. Szucs, J. W. Dolan and C. A. Pohl, "Error Measures in Quantitative Structure-Retention Relationships Studies," *Journal of Chromatography A*, vol. 1524, pp. 298-302, 2017.
- [30] E. Fernando, G. Harrison and S. Hilbrich, *Characterizing the Effects of Surface Roughness on Vehicle Dynamic Loads and Pavement Life*, Collage Station, TX: Texas Transportation Institute, 2007.
- [31] M. Senniappan, R. More, S. Bhide and S. Gowda, "Optimization of Commercial Vehicle's Steering Tie Rod Arm Design Based on Strain Life Approach," *SAE Technical Paper 2016-01-0381*, 2016.
- [32] K. K. Darpkar and A. Bidre, "Driveshaft Maximum Torque Estimation via Linear Model, Failure Analysis and Bench Test Simulation, an Alternative Approach," *SAE Technical Paper 2016-28-0171*, 2016.
- [33] T. H. Tran, Y. J. Wang, C. K. Cheng, P. C. Chao and C. C. Wang, "Using Maximum Likelihood to Calibrate a Six-DOF Force/Torque Sensor," *Microsystem Technologies*, vol. 24, pp. 4493-4509, 2018.
- [34] B. D. Grenke, "Digital Filtering for J211 Requirements using a Fast Fourier Transform Based Filter," in *SAE 2002 World Congress*, Detroit, 2002.
- [35] V. K. Madisetti and D. B. Williams, "Signals and Systems," in *The Digital Signal Processing Handbook: Digital Signal Processing Fundamentals*, Second ed., vol. 1, Boca Raton, FL: CRC Press, 2010, pp. (I-1)-(I-2).
- [36] W. B. Ferry, P. R. Frise, G. T. Andrews and M. A. Malik, "Combining Virtual Simulation and Physical Vehicle Test Data to Optimize Durability Testing," *Fatigue and Fracture of Engineering Materials and Structures*, vol. 25, pp. 1127-1134, 12 September 2002.
- [37] M. Frigo and S. G. Johnson, "FFTW: An Adaptive Software Architecture for the FFT," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, 1998.
- [38] J. W. Cooley and J. W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series," *Mathematics of Computation*, vol. 19, pp. 297-307, 1965.

- [39] "fft: Fast Fourier transform," [Online]. Available: <https://www.mathworks.com/help/matlab/ref/fft.html>.
- [40] L. C. Ludeman, *Fundamentals of Digital Signal Processing*, Philadelphia : Harper & Row, 1986.
- [41] S. P. Washington, M. G. Karlaftis and F. L. Mannering, *Statistical and Econometric Methods for Transportation Data Analysis*, Second ed., Boca Raton: CRC Press, 2011.
- [42] M. Pourgol-Mohammad, A. Hejazi, M. Soleimani, P. Ghasemi, A. Ahmadi and D. Jalali-Vahid, "Design for Reliability of Automotive Systems; Case Study of Dry Friction Clutch," *International Journal of Systems Assurance Engineering*, vol. 8, no. 3, pp. 572-583, September 2017.
- [43] Z. Wei, "The Uncertainty of Estimated Lognormal and Weibull Parameters for Test Data with Small Sample Size," SAE International, 2013.
- [44] D. G. Pavlou, *Essentials of the Finite Element Method: For Mechanical and Structural Engineers*, Elsevier Science & Technology, 2015.
- [45] P. M. Kurowski, *Engineering Analysis with SOLIDWORKS Simulation 2016*, Mission, KS: SDC Publications, 2016.
- [46] ANSYS, Inc, "19.0 Release Notes," Canonsburg, PA, 2018.
- [47] P. M. Kurowski, *Finite Element Analysis for Design Engineers*, SAE International, 2004.
- [48] E. Wang, T. Nelson and R. Rauch, "Back to Elements - Tetrahedra vs. Hexahedra," in *International ANSYS Conference*, Munich, Germany, 2004.
- [49] *Chalmers University To Teach Automotive Engineering Students in Sweden Design Fundamentals With SolidWorks Software*, Business Insights: Essentials , 2004.
- [50] *Donkervoort automobielen BV improves design, cuts costs with SolidWorks simulation; world-class sports car manufacturer reduces number of prototypes by half for newly redesigned, hand-crafted D8 GTO*, M2 Presswire, 2012.
- [51] *Ohio state university students design fastest electrical car in the world with SolidWorks education edition software*, 2004.
- [52] *SolidWorks and COSMOS/ Help Engineering Students Compete At SAE Competitions*, Business Insights: Essentials , 2002.

- [53] *University of Michigan Solar Car Team Revs Up SolidWorks to Design World Solar Challenge Contender*, Business Insights: Essentials, 2011.
- [54] S. D. Rajan, *Introduction to Structural Analysis & Design*, New York, NY: John Wiley & Sons, 2001.
- [55] R. C. Hibbeler, *Statics & Mechanics of Materials*, Fourth ed., Indianapolis, IN: Pearson, 2014.
- [56] R. G. Budynas and J. K. Nisbett, *Shigley's Mechanical Engineering Design*, Ninth ed., New York, NY: McGraw-Hill, 2011.
- [57] F. P. Beer, E. R. Johnston Jr, J. T. DeWolf and D. F. Mazurek, *Mechanics of Materials*, Sixth ed., New York, NY: McGraw Hill, 2012.
- [58] K. Izui, S. Nishiwaki and M. Yoshimura, "An Optimal Cross-Sectional Design Method for Automotive Body Frames," SAE International, 2003.
- [59] S. Roper , D. Li, V. Florea, C. Woischwill and I. Y. Kim, "Multi-Material Topology Optimization: A Practical Approach and Application," SAE International, 2018.
- [60] *Strain Gage Installations with M-Bond 200 Adhesive*, Vishay Micro-Measurements, 2017.
- [61] "Precision Strain Gage: Precision Linear Pattern for Static and Dynamic Applications," Omega.
- [62] *Application Note TT-609: Strain Gage Soldering Techniques*, Vishay Micro-Measurements , 2015.
- [63] "analogRead()," 21 February 2019. [Online]. Available: <https://www.arduino.cc/reference/en/language/functions/analog-io/analogread/>.
- [64] G. Held, *Dictionary of Communications Technology: Terms, Definitions and Abbreviations*, 3rd ed., Hoboken, NJ: John Wiley & Sons, 1998.
- [65] Broadcast Equipment Ltd, "Realterm Setup Program".
- [66] P. P. J. Abelkis, *Service Fatigue Loads Monitoring, Simulation, and Analysis: a symposium*, Baltimore, MD: ASTM, 1979.
- [67] P. Duhamel and M. Vetterli, "Fast Fourier Transforms: A Tutorial Review and State of the Art," in *The Digital Signal Processing Handbook: Digital Signal Processing*

- Fundamentals*, Second ed., vol. 1, Boca Raton, FL: CRC Press, 2010, pp. (7-1)-(7-49).
- [68] J. W. Dally, W. F. Riley and K. G. McConnel, *Instrumentation for Engineering Measurement*, New York, NY: John Wiley & Sons, 1984.
- [69] K. BreBler, M. Speckert, R. Muller and C. Weber, "Customer Loads Correlation in Truck Engineering," *Derichte des Faunhofer ITWM*, vol. 151, 2009.
- [70] W. K. Jenkins, "Fourier Methods for Signal Analysis and Processing," in *The Digital Singal Processing Handbook: Digital Signal Processing Fundamentals*, Second ed., vol. 1, Boca Raton, RL: CRC Press, 2010, pp. (1-1)-(1-29).
- [71] A. Javidinejad, *Essentials of Mechanical Stress Analysis*, New York, NY: CRC Press, 2015.
- [72] L. Huang, H. Agrawal and P. Kurudiyara, "Dynamic Durability Analysis of Automotive Structures," in *SAE Technical Paper Series*, Detriot, 1998.
- [73] J. Happian-Smith, Ed., *An Introduction to Modern Vehicle Design*, Boston, MA: Butterworth Heinemann, 2002.
- [74] C. S. Lindquist, *Adaptive & Digital Signal Processing: with Digital Filtering Applications*, vol. 2. International Series in Signal Processing and Filtering, Miami: Steward & Sons, 1989.
- [75] L. J. Karam, J. H. McClellan, I. W. Selesnick and C. S. Burrus, "Digital Filtering," in *The Digital Signal Processing Handbook: Digital Signal Processing Fundamentals*, Second ed., vol. 1, Boca Raton, FL: CRC Press, 2010, pp. (11-1)-(11-89).
- [76] S. Padmanabhan, D. Verma and P. Kumar, "Analysis for Improving Performance of Racing Go-Kart," *International Journal of Applied Engineering Research*, vol. 10, no. 5, pp. 1425-1428, 2015.
- [77] P. R. N. McKeighan, *Fatigue Testing and Analysis under Variable Amplitude Loading Conditions*, West Conshohocken, PA: ASTM, 2005.
- [78] C. N. Madivalar, T. Shay and S. Kolekar, "Fatigue Failure Analysis of Bike Crank Arm Using Solidworks Simulation," *Jounral of Mechanical Engineering Research and Development*, 2018.
- [79] S. Suresh, *Fatigue of Materials*, Second Edition ed., Cambridge: Cambridge University Press, 1998.

- [80] B. Yang, Stress, Strain, and Structural Dynamics: An Interactive Handbook of Formulas, Solutions, and Matlab Toolboxes, Boston, MA: Elsevier Academic Press, 2005.
- [81] C. S. Williams, Designing Digital Filters, T. Kailathe, Ed., Englewood Cliffs, NJ: Prentice-Hall, 1986.

Appendix A

Bibliography

Abelkis, P. P. (1979). *Service Fatigue Loads Monitoring, Simulation, and Analysis: a symposium*. Baltimore, MD, United States: ASTM.

BreBler, K., Speckert, M., Muller, R., & Weber, C. (2009). Customer Loads Correlation in Truck Engineering. *Derichte des Faunhofer ITWM, 151*.

Dally, J. W., Riley, W. F., & McConnell, K. G. (1984). *Instrumentation for Engineering Measurement*. New York, NY: John Wiley & Sons.

Duhamel, P., & Vetterli, M. (2010). Fast Fourier Transforms: A Tutorial Review and State of the Art. In V. K. Madisetti, *The Digital Signal Processing Handbook: Digital Signal Processing Fundamentals* (Second ed., Vol. 1, pp. (7-1)-(7-49)). Boca Raton, FL: CRC Press.

Happian-Smith, J. (Ed.). (2002). *An Introduction to Modern Vehicle Design*. Boston, MA: Butterworth Heinemann.

Huang, L., Agrawal, H., & Kurudiyara, P. (1998). Dynamic Durability Analysis of Automotive Structures. *SAE Technical Paper Series*. Detroit: SAE International.
doi:10.4271/980695

Javidinejad, A. (2015). *Essentials of Mechanical Stress Analysis*. New York, NY, United States: CRC Press.

Jenkins, W. K. (2010). Fourier Methods for Signal Analysis and Processing. In V. K. Madisetti, *The Digital Singal Processing Handbook: Digital Signal Processing Fundamentals* (Second ed., Vol. 1, pp. (1-1)-(1-29)). Boca Raton, RL: CRC Press.

Karam, L. J., McClellan, J. H., Selesnick, I. W., & Burrus, C. S. (2010). Digital Filtering. In V. K. Madisetti, *The Digital Signal Processing Handbook: Digital Signal Processing Fundamentals* (Second ed., Vol. 1, pp. (11-1)-(11-89)). Boca Raton, FL: CRC Press.

Lindquist, C. S. (1989). *Adaptive & Digital Signal Processing: with Digital Filtering Applications* (Vol. 2. International Series in Signal Processing and Filtering). Miami: Steward & Sons.

Madivalar, C. N., Shay, T., & Kolekar, S. (2018). Fatigue Failure Analysis of Bike Crank Arm Using Solidworks Simulation. *Jounral of Mechanical Engineering Research and Development*. doi:10.26480/jmerd.03.2018.09.13

McKeighan, P. R. (2005). *Fatigue Testing and Analysis under Variable Amplitude Loading Conditions*. West Conshohocken, PA, United States: ASTM.

Padmanabhan, S., Verma, D., & Kumar, P. (2015). Analysis for Improving Performance of Racing Go-Kart. *International Journal of Applied Engineering Research*, 10(5), 1425-1428.

Suresh, S. (1998). *Fatigue of Materials* (Second Edition ed.). Cambridge, United Kingdom: Cambridge University Press.

Williams, C. S. (1986). *Designing Digital Filters*. (T. Kailathe, Ed.) Englewood Cliffs, NJ: Prentice-Hall.

Yang, B. (2005). *Stress, Strain, and Structural Dynamics: An Interactive Handbook of Formulas, Solutions, and Matlab Toolboxes*. Boston, MA, United States: Elsevier Academic Press.

Appendix B

Vishay Micro-Measurements: Strain Gage Installations with M-Bond 200 Adhesive

Step 1:

- a) Thoroughly degrease the gauging area with isopropyl alcohol. Recommended to use one-way containers to make sure solvents are uncontaminated. [60]

Action Performed:

- a) Poured isopropyl alcohol into a clean mug, only dipped new gauze pads into mug, wiped gauging area with isopropyl alcohol.

Step 2:

- a) If surface scale or oxide present, preliminary dry abrading with 220- or 320-grit silicon-carbide paper is generally required.
- b) Final abrading is performed with 320-grit silicon-carbide paper on the gauging surface(s) after being thoroughly wetted with M-Prep Conditioner A
- c) Followed by wiping dry with a gauze pad
- d) Repeat steps b and c with 400-grit silicon-carbide paper.
- e) Using ballpoint pen, burnish alignment marks, clean with M-Prep Conditioner A and cotton tipped applicators until there is no discoloration on the tip.
- f) Remove all residue and Conditioner by slowly wiping the surface with a gauze pad. Never allow any solution to dry on the surface because this will leave a contaminating film and interfere with the bond strength.
- g) Apply a liberal amount of M-Prep Neutralizer 5A and scrub with a cotton-tipped applicator.
- h) With a single, slow wiping motion of a gauze pad, carefully dry this surface. [60]

Action Performed:

- a) Paint and rust present on the specimen, therefore preliminary dry abrading performed with 320-grit silicon-carbide paper.
 - b) Final abrading performed with 320-grit silicon-carbide paper, after being thoroughly wetted with M-Prep conditioner A.
 - c) Wiped the surface in one direction completely dry with gauze pads
 - d) Repeated steps b and c with 600-grit silicon-carbide paper
 - e) Used a scribe to burnish alignment marks into the specimen then cleaned surface again with M-Prep Conditioner A and gauze pads
 - f) Thoroughly dried specimen with gauze pads, wiping in a slow fluid unidirectional motion.
 - g) Applied a liberal amount of M-Prep Neutralizer 5A and scrubbed with cotton-tipped applicators.
 - h) Area was dried with a gauze pad, wiping in a slow fluid unidirectional motion.
- Steps g and h were repeated until a new applicator showed no discoloration after wiping the surface.

Step 4:

- a) Using tweezers to remove the gauge from the transparent envelope, place the gauge (bonding side down) on a chemically clean glass plate or gauge box surface. If a solder terminal will be used, position it on the plate adjacent to the gauge as shown. A space of approximately 1/16 in or more where the space allows, or application requires should be left between the gauge backing and the terminal.

- b) Place a 4- to 6-in piece of Vishay Micro-Measurement PCT-2M gauge installation tape over the gauge and terminal. Take care to center the gauge on the tape.
- c) Carefully lift the tape at a shallow angle, bringing the gauge up with the tape as illustrated above. [60]

Action Performed:

- a) Using needle nose pliers to remove the gauge from the transparent envelope the gauge was placed onto the specimen, as it was the only chemically clean surface. Solder terminal was placed approximately 1/16 in below the gauge backing.
- b) 4 in piece of PCT-2M gauge installation tape was placed over the gauge and terminal, taking care to center the tape on the gauge.
- c) Tape was slowly and carefully peeled back at a very shallow angle (less than 45deg).

Step 5:

- a) Position the gauge/tape assembly so that the triangle alignment marks on the gauge are over the layout lines (alignment marks) on the specimen. If the assembly appears to be misaligned, lift one end of the tape at a shallow angle until the assembly is free of the specimen. Realign properly, and firmly anchor at least one end of the tape to the specimen. Vishay PCT-2M gauge installation tape can be realigned without contamination by the tape mastic because this tape will retain its mastic when removed. [60]

Action Performed:

- a) Gauge was positioned along burnished alignment marks, starting with the top triangle alignment mark on the gauge. The tape/gauge assembly was pivoted a

few times to align the triangle alignment marks on the side of the gauge with the corresponding burnish marks on the specimen. Gauge tape firmly pressed to specimen.

Step 6:

- a) Lift the gauge end of the tape assembly at a shallow angle to the specimen surface (about 40 degrees) until the gauge and terminal are free of the specimen surface. Continue lifting the tape until it is free from the specimen approximately $\frac{1}{2}$ in beyond the terminal.
- b) Tuck the loose end of the tape under and press to the specimen surface so that the gauge and terminal lie flat, with the bonding surface exposed. [60]

Action Performed:

- a) Gauge tape was lifted, at a shallow angle (about 40 degrees) until the terminal and strain gauge were fully removed from the specimen.
- b) The gauge tape was then peeled back $\frac{1}{2}$ in more as specified in the Vishay Micro-Measurements strain gauge installation procedure.

Step 7:

- a) M-Bond 200 catalyst can now be applied to the bonding surface of the gauge and terminal. (M-Bond 200 adhesive will harden without the catalyst, but less quickly and reliably.) Very little catalyst is needed, and it should be applied in a thin, uniform coat.
- b) Lift the brush-cap out of the catalyst bottle and wipe the brush approximately 10 strokes against the inside of the neck of the bottle to wring out most of the catalyst.

- c) Set the brush down on the gauge and swab the gauge backing. Do not stroke the brush in a painting style but slide the brush over the entire gauge surface and then the terminal.
- d) Move the brush to the adjacent tape area prior to lifting from the surface.
- e) Allow the catalyst to dry at least one minute under normal ambient conditions of +75 deg F and 30-65% relative humidity before proceeding. [60]

Action Performed:

- a) After raising to room-temperature, the M-Bond 200 catalyst bottle was opened, and the brush was wiped 10 times on the inside of the neck of the bottle to wring out most of the catalyst. Too much catalyst can cause poor bond strength, age-embrittlement of the adhesive and poor glue-line thickness, among other issues.
- b) The brush was then placed on the gauge, at the top of the gauge, and with one fluid stroke the brush was used to swab the gauge and terminal backing and continued onto the adjacent tape area before lifting the brush from the surface.
- c) A timer was set for one minute, after which time the catalyst was checked, visually, for dryness (if the catalyst isn't completely dry there will be areas of no discoloration, this discoloration looks like a chalky texture and is whiter than the deep blue of the applied catalyst.).

Steps 8, 9, and 10 must be completed in the sequence shown and within 3-5 seconds.

Step 8:

- a) Lift the tucked-under tape end of the assembly, and, holding in the same position, apply one or two drops of M-Bond 200 adhesive at the fold formed by the

junction of the tape and specimen surface. This adhesive application should be approximately ½ in outside the actual gauge installation area. This will insure that local polymerization that takes place when the adhesive comes in contact with the specimen surface will not cause unevenness in the gauge glue line. [60]

Action Performed:

- a) After the catalyst was determined to be fully dry, one to one and a half minutes, one drop of M-Bond 200 adhesive was applied at the fold formed by the junction of the tape and the specimen surface, approximately ½ in from the gauge installation area.

Step 9:

- a) Immediately rotate the tape to approximately a 30-degree angle so that the gauge is bridged over the installation area.
- b) While holding the tape slightly taut, slowly and firmly make a single wiping stroke over the gauge/tape assembly with a piece of gauze bringing the gauge back down over the alignment marks on the specimen.
- c) Use a firm pressure with your fingers when wiping over the gauge. A very thin, uniform layer of adhesive is desired for optimum bond performance. [60]

Action Performed:

- a) The tape was immediately rotated to approximately 30 degrees from the specimen.
- b) Holding the tape taut with one hand, a gauze pad was used in the other hand to apply a firm and slow wiping stroke over the assembly.

Step 10:

- a) Immediately upon completion of wipe-out of the adhesive, firm thumb pressure must be applied to the gauge and terminal area.
- b) This pressure should be held for at least one minute. In low-humidity conditions (below 30%), or if the ambient temperature is below +70 deg F, this pressure application time may have to be extended to several minutes. [60]

Action Performed:

- a) Constant firm thumb pressure was applied after the wipe-out of the adhesive.
- b) A timer was set for one minute and a half and constant firm pressure was held for this time.

Step 11:

- a) The gauge and terminal strip are now solidly bonded in place. It is not necessary to remove the tape immediately after gauge installation. The tape will offer mechanical protection for the grid surface and may be left in place until it is removed for gauge wiring.
- b) To remove the tape, pull it back directly over itself, peeling it slowly and steadily off the surface. This technique will prevent possible lifting of the foil on open-faced gauges or other damage to the installation. [60]

Action Performed:

- a) The gauge tape was left on to protect the gauge from the subsequent installations.
- b) After installations were all completed, one strain gauge at a time, the strain gauge tape was removed by pulling it back directly over itself, in a slow and steady

motion, to solder the ribbon leads to the solder terminals. The gauge tape was not removed from remaining gauges.

- c) The protective coating was removed from each ribbon lead with a pencil eraser.
- d) M-Flux AR-2 Activated Resin soldering Flux was applied to the solder tap and ribbon lead

Appendix C

Vishay Micro-Measurements Application Note TT-609: Strain Gage Soldering

Techniques

Soldering must be done with the appropriate tools, supplies, and techniques to ensure accurate strain measurement. Vishay application note TT-609, outlines the recommended procedures and materials to achieve accurate lead-wire attachment to strain gauge solder tabs. Vishay recommends using a temperature-controlled soldering station as an unregulated soldering iron is most likely too hot, causing unwanted oxidation of the solder tip. Elevated temperatures can cause damage to the strain gauge, bonding adhesive and even the test specimen. It is recommended to use a temperature-controlled iron at the recommended temperature for the solder utilized. For routine applications, excluding testing at elevated temperatures or specimens that cannot be exposed to heat, a solder with an intermediate melting temperature is the normal selection. If rosin-core solder is used no soldering flux is necessary, if using solid-wire solder, a liquid activated-rosin flux is recommended to remove oxidation from members being joined and prevent further oxidation during soldering. M-Flux AR-2 activated resin soldering flux (M-Flux AR) was utilized during this installation, following the Vishay application note TT-609 recommendations of applying flux to tin the solder terminals and lead-wires before making soldered connections [60]. Tinning helps ensure good heat transfer during the soldering operation and simplify the lead-wire attachment procedure. During solder installation the strain gauges were protected with gauge tape, to prevent any solder splash from causing damage to the strain gauge. In preparation for soldering to the terminal the ribbon leads were stripped of their protective coating, with a pencil eraser, as instructed

in Omega documentation [61]. The ribbon leads were positioned on the terminal and ready for soldering, Figure 50.



Figure 50: Ribbon Leads Aligned on Solder Pad with Protective Coating Removed.

The terminal tinning procedure from Vishay application note TT-609 was followed exactly;

“The tinning procedure for strain gage tabs and terminals consists of first cleaning and reapplying a small amount of solder to the hot soldering iron tip. Next, apply a drop of M-Flux AR to the tab or terminal. Hold the soldering pencil in a nearly horizontal position (<30 deg), with the flat surface of the tip parallel to the solder tab or terminal. Place the solder wire flat on the gage tab, and press firmly with the tinned hot soldering tip for about one to two seconds, while adding approximately $1/8$ in of fresh solder at the edge of the tip. This procedure ensures that there is sufficient solder and flux for effective tinning. Simultaneously lift both the soldering pencil and solder wire from the tab area. If M-Flux AR... is used in the tinning, it is not necessary to remove the residual soldering flux at this time.” [62, p. 209].

Lead-wires must be prepped next for installation. For preparation the wire must be stripped and tinned. The wire is stripped and twisted for ease of tinning. Tinning is done by applying M-Flux AR to the wire end. Solder is melted on the tip of the soldering iron to form a hemisphere of molten solder, twice the diameter of the wire to be tinned. The wire is then slowly drawn through the molten solder, while adding more solder to the interface as needed.

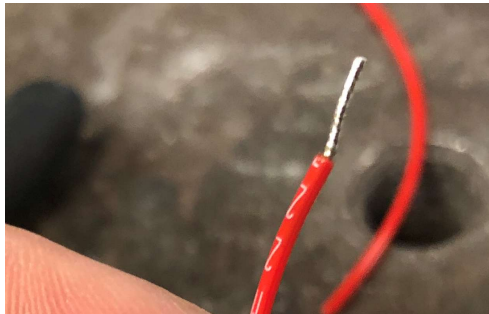


Figure 51: Tinned Lead-Wire

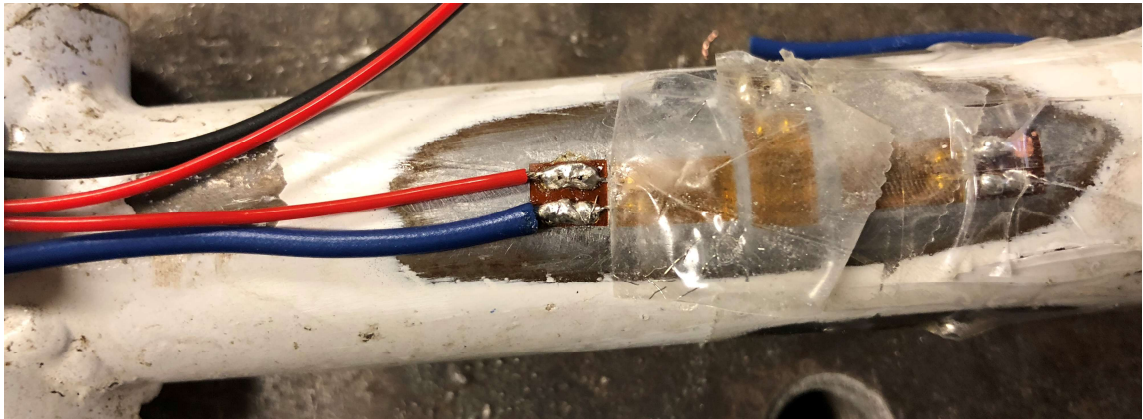


Figure 52: Lead-Wires Installed on Solder Pads After Tinning

The tinned lead-wire can then be positioned on the specimen and trimmed such that the exposed, tinned, wire is only as long as the connection area on the solder terminal and does not protrude off the solder terminal. The wire was affixed in this position with PDT-1 6600 drafting tape before following the Vishay instructions for proper soldering

attachment; “Apply a small amount of M-Flux AR to the joint area and, holding the soldering pencil nearly horizontal, firmly press the flat surface of the tip on the junction for about one second; then lift the tip from the soldered joint. This procedure should result in a smooth, hemispherical solder joint, without any peaks or jagged areas.” [62].

After soldering is complete M-Line Rosin Solvent was used to remove any residual flux from the soldering joints. This step is crucial as any flux residue can cause gauge instability and drift.

After completing the installation and securing the lead-wires to the solder terminals, the circuits were tested to ensure proper strain reading before coating the installation with polyurethane.

Each strain gauge was covered with strain gauge tape to protect it until a polyurethane coating could be applied. One strain gauge at a time, the strain gauge tape was removed and a rectangular area around the strain gauge, including the soldering terminal and the 1/16th in of the encased lead-wires, was taped off with PDT-1 6600 drafting tape, layered to keep the polyurethane contained. M-Coat A air-drying polyurethane coating was applied and allowed to dry for 20 minutes, as recommended on the bottle, repeating to form enough coats to cover the entire gauge, terminals, solder, and lead-wires within the taped off area.

Appendix D

DAQ Scripts

Arduino script written to use ‘analogRead()’ to perform the analog to digital conversion of the strain gauge and voltage signals, seen in Figure 53, Figure 54, and Figure 55, where the commented out portions of the code were used to verify the sample frequency.

```

/*****
  Thesis Data Collection Code
  MacKenzie Cunningham
  */

#include <Wire.h> // Must include Wire library for I2C

float factor1 = 1;
float factor2 = 1;
float factor3 = 1;

unsigned long time_step;

unsigned long cnt=0;
unsigned long tNow, tLast=0;
unsigned long interval =2850; //(us)
int sensorValue;
float voltage;

int strain1_reading;
float strain1_value;
int strain2_reading;
float strain2_value;
int strain3_reading;
float strain3_value;

int tPrev1;
int tPrev2;
int tVolt;
int tPrev3;
int tStrain1;
int tPrev4;
int tStrain2;
int tPrev5;
int tStrain3;
int tPrev6;
int tSerialP1;
int tPrev7;
int tPrev8;
int tSerialP2;
int tSerialP3;

```

Figure 53: Arduino Strain Gauge and Voltage Reading Script, Part 1 of 3

```

void setup() {
  Serial.begin(115200);

  Serial.println("New Run:");
}

void loop() {

tNow = micros();

if( (tNow-tLast)>interval ){
  tLast=tNow;

  tPrev1 = micros();

  sensorValue = analogRead(A0); //read A0 pin to get actual voltage V5
  voltage = float(sensorValue) * (5.0 / 1023.0); //Calculate actual voltage

  tPrev2 = micros();
  tVolt = tPrev2-tPrev1;

  strain1_reading = analogRead(A4);
  strain1_value = float(strain1_reading) / voltage * 1000000;

  tPrev3 = micros();
  tStrain1 = tPrev3-tPrev2;

  strain2_reading = analogRead(A2);
  strain2_value = float(strain2_reading) / voltage * 1000000;

  tPrev4 = micros();
  tStrain2 = tPrev4-tPrev3;

  strain3_reading = analogRead(A1);
  strain3_value = float(strain3_reading) / voltage * 1000000;

  tPrev5 = micros();
  tStrain3 = tPrev5-tPrev4;

  time_step = micros();
  Serial.print(String(cnt) + ",");
  Serial.print(String(voltage, 2) + ",");
  Serial.print(String(strain1_reading) + ",");

```

Figure 54: Arduino Strain Gauge and Voltage Reading Script, Part 2 of 3

```

// Serial.print(voltage, 2);
// Serial.print(",");
// Serial.print(strain1_reading);
// Serial.print(",");

tPrev6 = micros();
tSerialP1 = tPrev6-tPrev4;

// Serial.print(String(strain2_reading) + ",");

Serial.print(strain2_reading);
Serial.print(",");

tPrev7 = micros();
tSerialP2 = tPrev7-tPrev6;

// Serial.println(String(strain3_reading) + ",");

Serial.println(strain3_reading);

tPrev8 = micros();
tSerialP3 = tPrev8-tPrev7;

// Serial.print(String(time_step)+ ",");
// Serial.print(String(tVolt) + "," + String(tStrain1) + "," + String(tStrain2) + ",");
// Serial.print(String(tStrain3) + "," + String(tSerialP1) + "," + String(tSerialP2));
// Serial.println(", " + String(tSerialP3));
//delayMicroseconds(2857/2);

// Serial.print(strain1_reading, 12);
// Serial.print(" ");
// Serial.print(strain2_reading, 12);
// Serial.print(" ");
// Serial.println(strain3_reading, 12);
// Serial.print(" ");

cnt++;
}
}

```

Figure 55:Arduino Strain Gauge and Voltage Reading Script, Part 3 of 3

The Python scripts written to read the serial signals from the Arduino, Figure 56, and the GPS, Figure 57, Figure 58, and Figure 59.


```

#!/usr/bin/python3
#
# simplest serial read and print
#
# Marc Compere, comperem@gmail.com
# created : 27 April 2017
# modified: 06 December 2018

# -----UNTESTED UNTESTED UNTESTED -----
#
# the Enerac M700's require a command string to be sent, so I moved
# to SerialSendAndRead.py
#
# -----UNTESTED UNTESTED UNTESTED -----

import sys
import time
import serial

device = '/dev/ttyACM0'
#device='COM4'

# this makes the string: YY_MM_DD_HH_MM_SS
# https://docs.python.org/3/library/time.html#time.strftime
dateStr = time.strftime("%Y_%m_%d_%H_%M_%S")
fileName = dateStr + ".csv"
print( 'opening fileName for writing .csv data [{0}]...'.format(fileName) , end='') # no newline
myFile = open(fileName,'w')
print('done.')

ser = serial.Serial(device, baudrate=115200, bytesize=8, parity='N', stopbits=1)

ser.reset_output_buffer()
ser.reset_input_buffer() # flush input buffer from prior Arduino strings
time.sleep(0.5)

print('Reading serial device...')
print('\nPress Ctrl-C to quit...')

while 1:
    try:
        myBytes = ser.readline() # read a string from the serial port
        #print(myBytes)
        print("-----")
        myString = myBytes.decode()
        print("myString:",myString)

        myList = myString.split(',')
        print(myList)
        print("one number: {0}".format(myList[1]))

        newString = str( time.time() ) + ", " + myString
        myFile.write(newString)

    except KeyboardInterrupt as err:
        print("caught keyboard ctrl-c:".format(err))
        myFile.close()
        print("closed file; exiting.")
        exit(0)
    except:
        print("Unexpected error:", sys.exc_info()[0])

```

Figure 56: Arduino Python Script, Serial Read and Write


```

#!/usr/bin/python3
#
# example usage:
#   python pynmea_example4.py 9600      (if you have *not* run 'python GPS_Settings.py /dev/ttyUSB0')
#   python pynmea_example4.py 115200   (if 'python GPS_Settings.py /dev/ttyUSB0')
#
# Read and parse NMEA GPS strings directly from the serial port.
#
# This uses python library pynmea2 and does *not* use gpsd or associated gps library.
#
#
# Adafruit Ultimate GPS provides 5 messages by default: GPGGA, GPRMC, GPGSA, GPGSV, GPVTG
#
# GPGGA is the only message needed now: http://www.gpsinformation.org/dale/nmea.htm#GGA
#   time, lat, lon, fix quality, number of satellites, HDOP
#
# -----
# $GPGGA,123519,4807.038,N,01131.000,E,1,08,0.9,545.4,M,46.9,M,,*47
#
# Where:
#   GGA           Global Positioning System Fix Data
#   123519       Fix taken at 12:35:19 UTC
#   4807.038,N   Latitude 48 deg 07.038' N
#   01131.000,E  Longitude 11 deg 31.000' E
#   1           Fix quality: 0 = invalid
#                   1 = GPS fix (SPS)
#                   2 = DGPS fix
#                   3 = PPS fix
#                   4 = Real Time Kinematic
#                   5 = Float RTK
#                   6 = estimated (dead reckoning) (2.3 feature)
#                   7 = Manual input mode
#                   8 = Simulation mode
#   08          Number of satellites being tracked
#   0.9         Horizontal dilution of position
#   545.4,M     Altitude, Meters, above mean sea level
#   46.9,M     Height of geoid (mean sea level) above WGS84
#               ellipsoid
#   (empty field) time in seconds since last DGPS update
#   (empty field) DGPS station ID number
#   *47        the checksum data, always begins with *
# -----
#
# pynmea2 library: https://github.com/Knio/pynmea2
#
# copy-pasted from: https://www.raspberrypi.org/forums/viewtopic.php?f=32&t=119287
#
# Daniel Tellez, tellezd@my.erau.edu
# Joseph Merciez, merciezj@my.erau.edu
# Marc Compere, comperem@erau.edu
# MacKenzie Cunningham, cunninm4@my.erau.edu
# created : 21 Oct 2016
# modified: 06 Dec 2018

# bring in: print(*objects, sep=' ', end='\n', file=sys.stdout)
from __future__ import print_function
from random import random
import serial
import pynmea2
import socket
import struct
import sys
import time
#import datetime
import subprocess
from datetime import datetime, timedelta

# this makes the string: YY_MM_DD_HH_MM_SS
# https://docs.python.org/3/library/time.html#time.strftime
dateStr = time.strftime("%Y_%m_%d_%H_%M_%S")
fileName = dateStr + ".csv"
print( 'opening fileName for writing .csv data [{}]...'.format(fileName) , end='') # no newline
myFile = open(fileName,'w')
print('done.')

```

Figure 57: GPS Python Script, Serial Read and Write, Part 1 of 3

```

def get_rssi ():
    ret = subprocess.run(["cat", "/proc/net/wireless"], stdout = subprocess.PIPE)

    lines = ret.stdout.decode('utf_8').split('\n')

    row_of_numbers = lines[2].split()

    print(row_of_numbers)
    if len(row_of_numbers) == 0:
        while len(row_of_numbers) == 0:

            ret = subprocess.run(["cat", "/proc/net/wireless"], stdout = subprocess.PIPE)

            lines = ret.stdout.decode('utf_8').split('\n')

            row_of_numbers = lines[2].split()

            time.sleep(7)
            return get_rssi()

    else:

        rssi_dbm = row_of_numbers[3]
        return rssi_dbm

if len(sys.argv) < 2:
    print("no serial baud rate specified (9600 or 115200)")
    print("using default 38400")
    #sys.exit()
    #serialBaud=115200
    #serialBaud=38400
    serialBaud=9600 # UBLOX reports at 9600/8/N/1
else:
    serialBaud=int(sys.argv[1])

print('opening serial port with baud [%i]' % serialBaud)

```

Figure 58: GPS Python Script, Serial Read and Write, Part 2 of 3

```

udp_sendto_addr = "10.33.11.114"
#udp_sendto_addr = "192.168.11.100"
udp_port_GPGGA = 5555
udp_port_GPRMC = 5555

sock = socket.socket(socket.AF_INET, # Internet
                     socket.SOCK_DGRAM) # UDP

#-----
# open serial port to read NMEA strings
#-----
#ser = serial.Serial('/dev/ttyUSB0',9600)
#-----
# ---> plug in GPS last to make it /dev/ttyACM0
#ser = serial.Serial('/dev/ttyACM0',serialBaud)
ser = serial.Serial('/dev/ttyACM1',serialBaud) # UBLOX GPS is /dev/ttyACM0 or 1 when steering encoder AMT203
#-----

ser.flushInput()
ser.flushOutput()

print('Reading GPS data from Ultimate GPS Breakout V3.')
print('\nPress Ctrl-C to quit...')

while 1:
    data = ser.readline()

    #-----
    # process GPRMC
    #-----

    if (data.startswith(b"GPRMC")):
        msg = pymea2.parse( data.decode("utf-8") )
        rssi_dbm = get_rssi()

        if (msg.timestamp == None):
            print ("lost signal")

        else:

            # print the important messages: time, lat, lon, status, speed, date
            #print(msg.timestamp, msg.lat, msg.lon, msg.gps_qual, msg.num_sats, msg.horizontal_dil)
            print(epoch, end="")
            print(',lat=%-12.12s' % msg.latitude, end=" ") # note: msg.latitude converts to decimal degrees rather than Deg, min, sec in the raw string
            print(',lon=%-12.12s' % msg.longitude, end="") # msg.longitude is decimal degrees
            print(',stat=%s' % msg.status, end="")
            print(',spd=%5.5s' % msg.spd_over_grnd, end="")
            print(',date=%s,' % msg.datestamp, end="")

            myString = str(str(time.time() + ", " + msg.spd_over_grnd + ", " + msg.latitude + ", " + msg.longitude))
            print(myString)
            myFile.write(myString)

```

Figure 59: GPS Python Script, Serial Read and Write, Part 3 of 3

Appendix E

Plotted Results

GPS Waypoint and Parking Chalk Location. The Matlab “polyxpoly” function was utilized to return the location of intersection of each GPS dataset with the chalk line and the index of the closest GPS datapoint from each dataset before the parking chalk location, seen in Figures Figure 60-Figure 70.

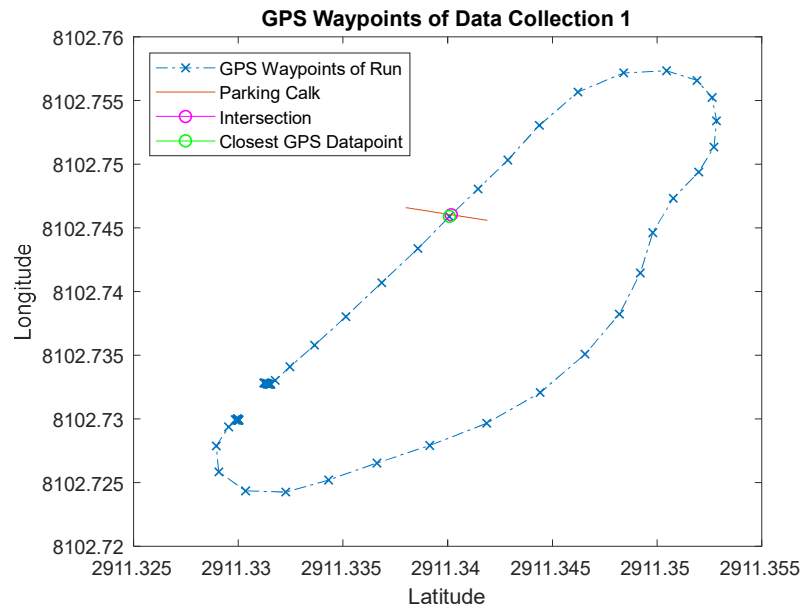


Figure 60: GPS Waypoints of Data Collection 1

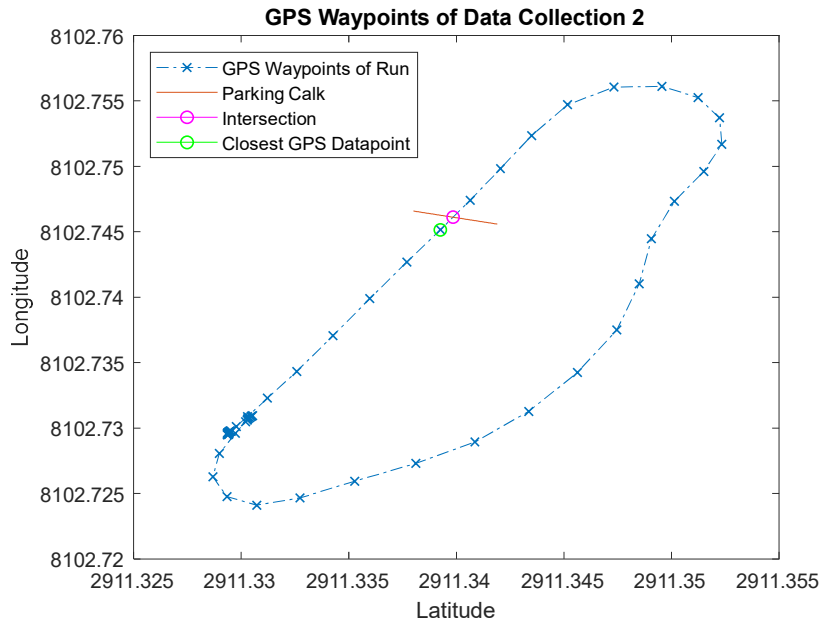


Figure 61: GPS Waypoints of Data Collection 2

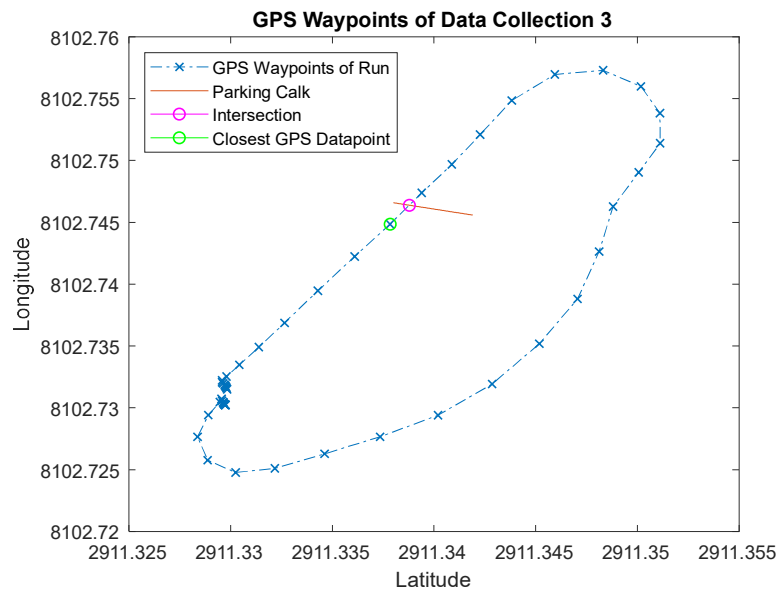


Figure 62: GPS Waypoints of Data Collection 3

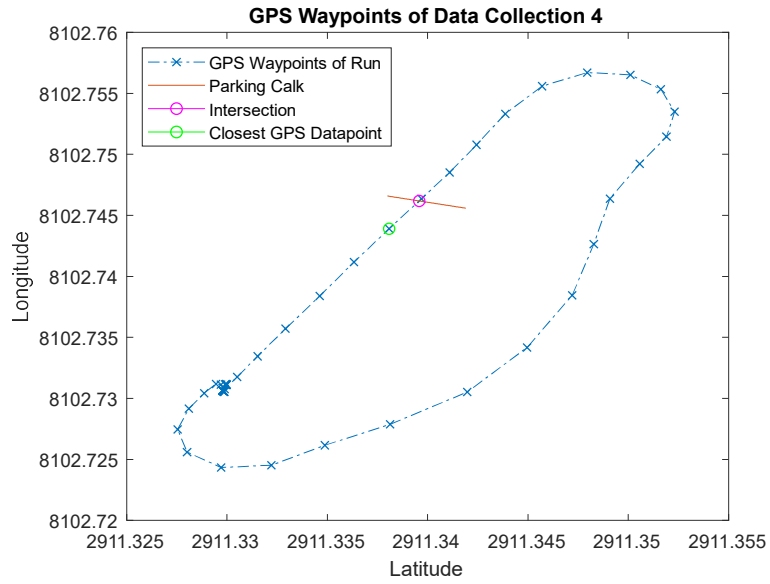


Figure 63: GPS Waypoints of Data Collection 4

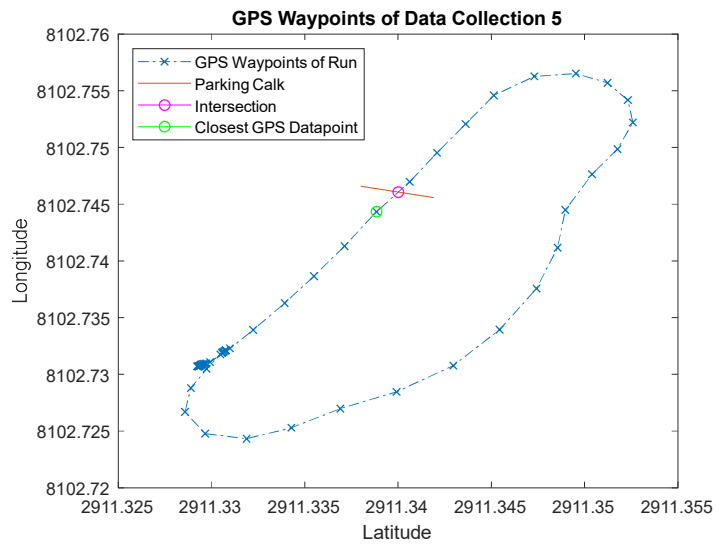


Figure 64: GPS Waypoints of Data Collection 5

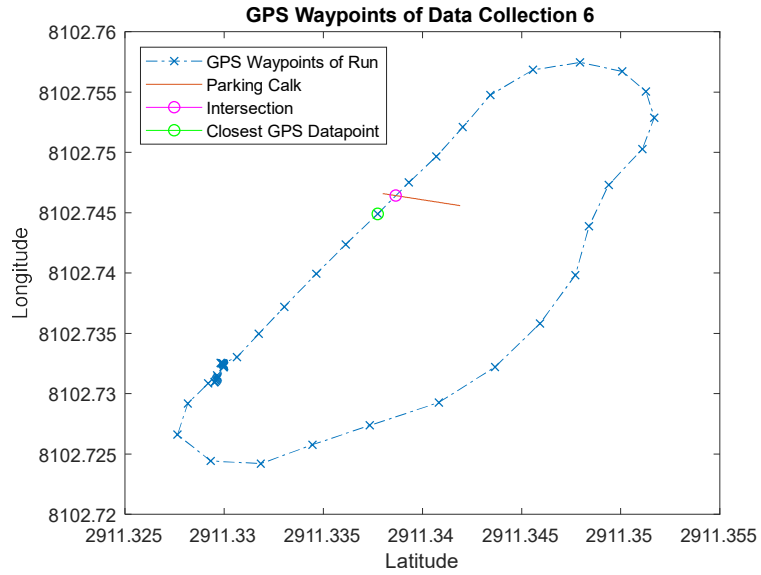


Figure 65: GPS Waypoints of Data Collection 6

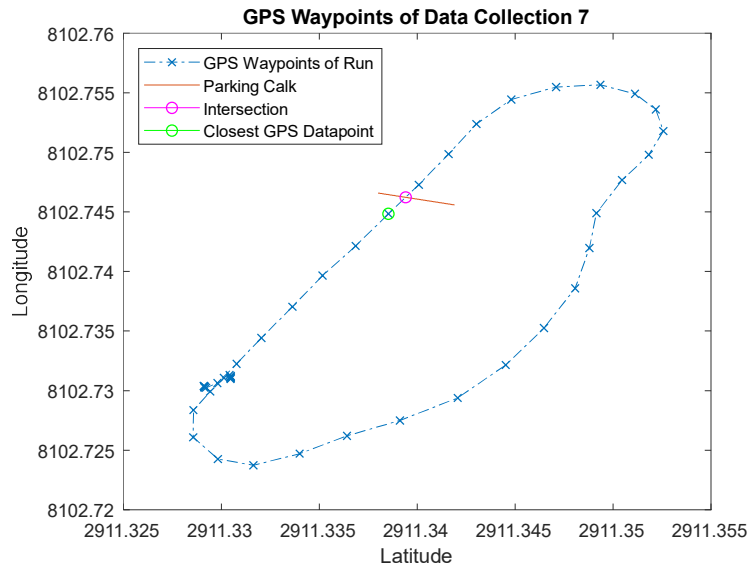


Figure 66: GPS Waypoints of Data Collection 7

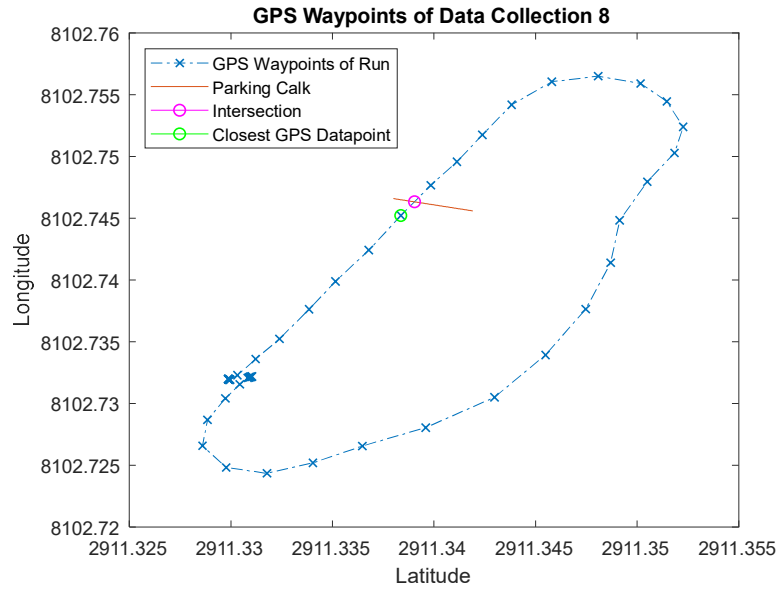


Figure 67: GPS Waypoints of Data Collection 8

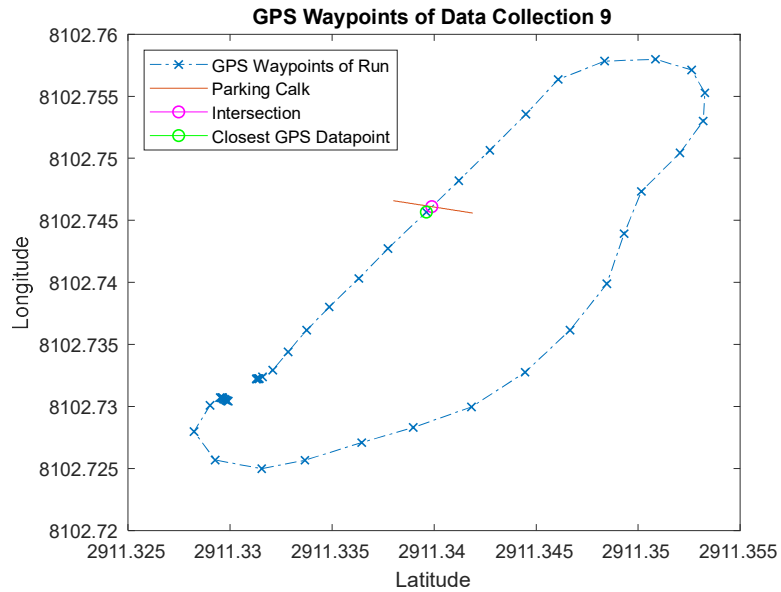


Figure 68: GPS Waypoints of Data Collection 9

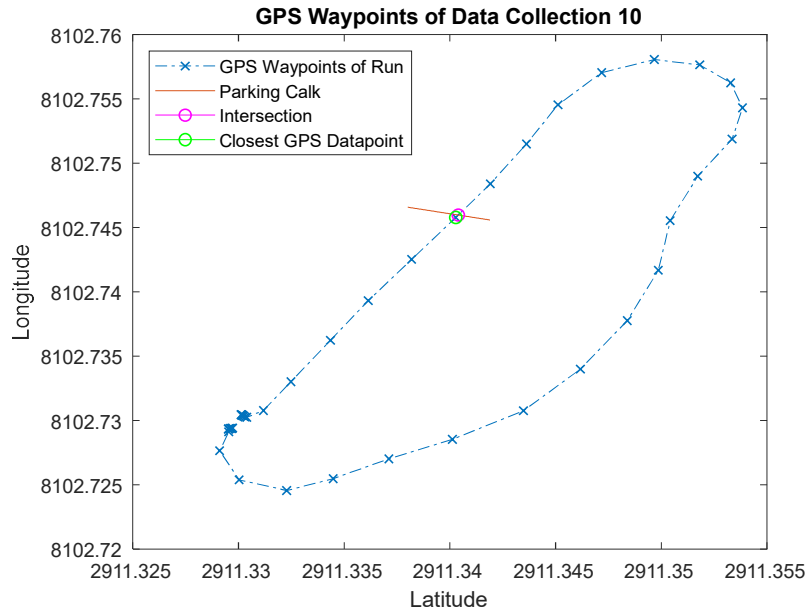


Figure 69: GPS Waypoints of Data Collection 10

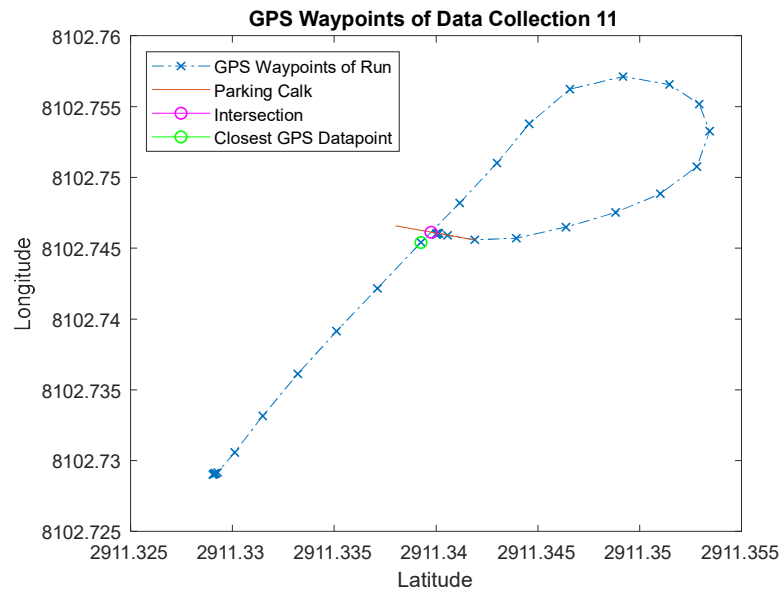


Figure 70: GPS Waypoints of Data Collection 11

FFT Digital Filtered Results. Pre- and Post-FFT resulting ARD V_{m_analog} signal can be seen for each data collection in Figure 71 to Figure 103.

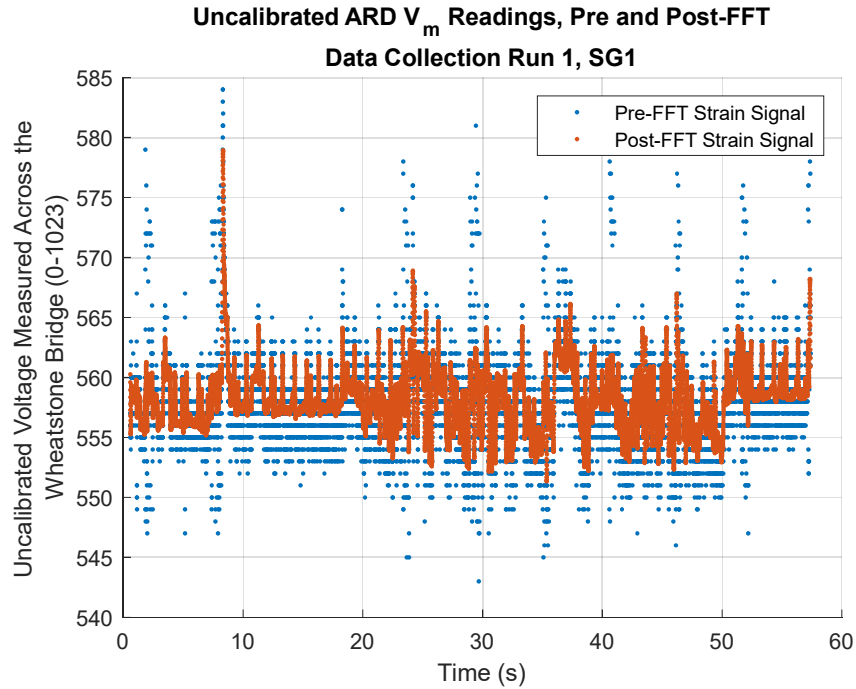


Figure 71: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 1, SG1.

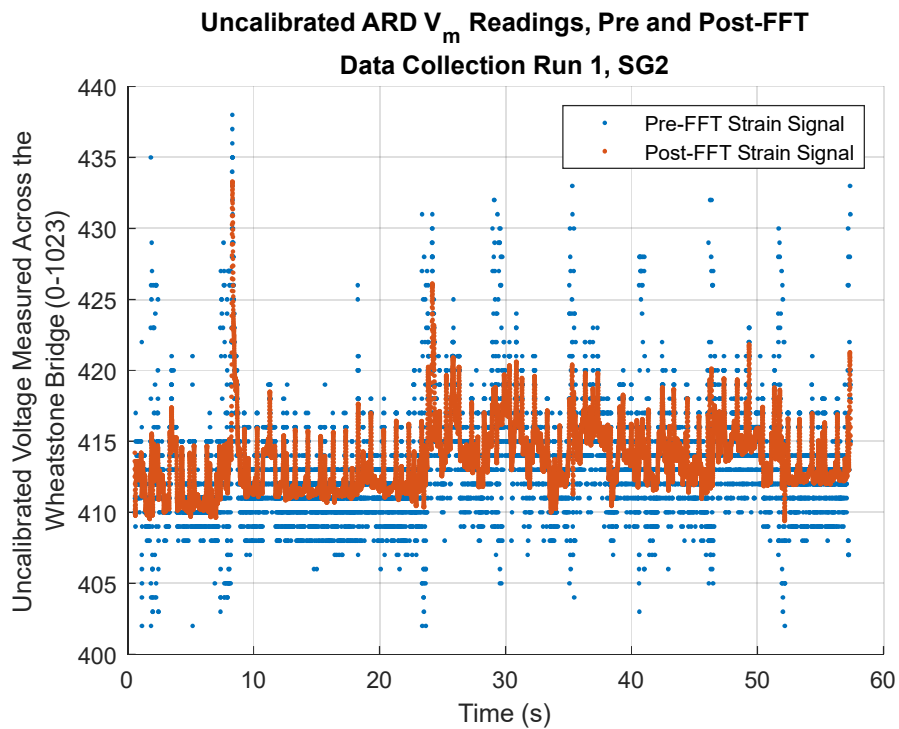


Figure 72: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 1, SG2.

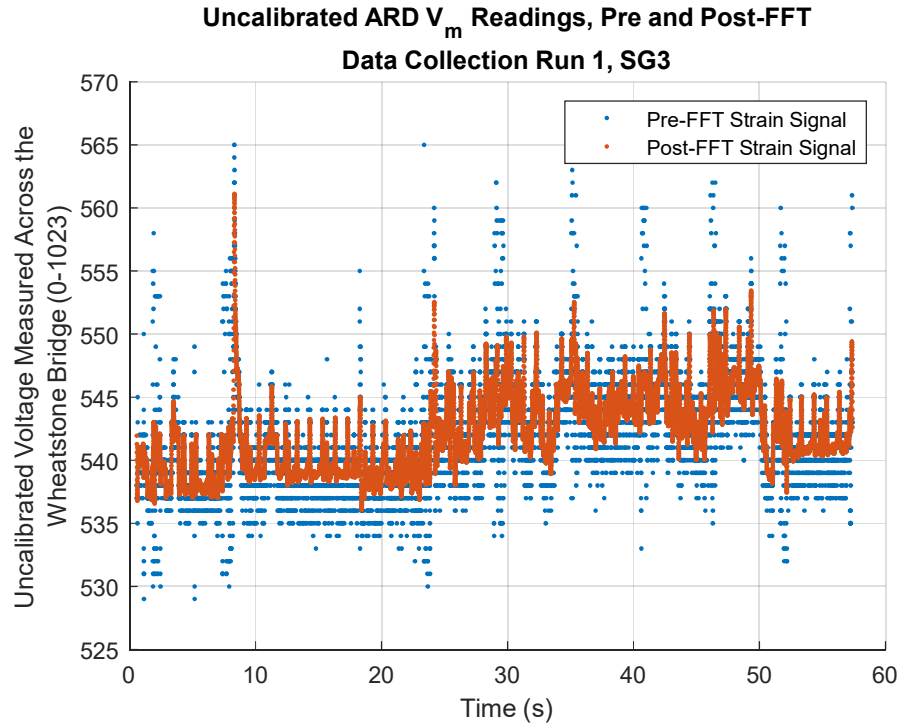


Figure 73: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 1, SG3.

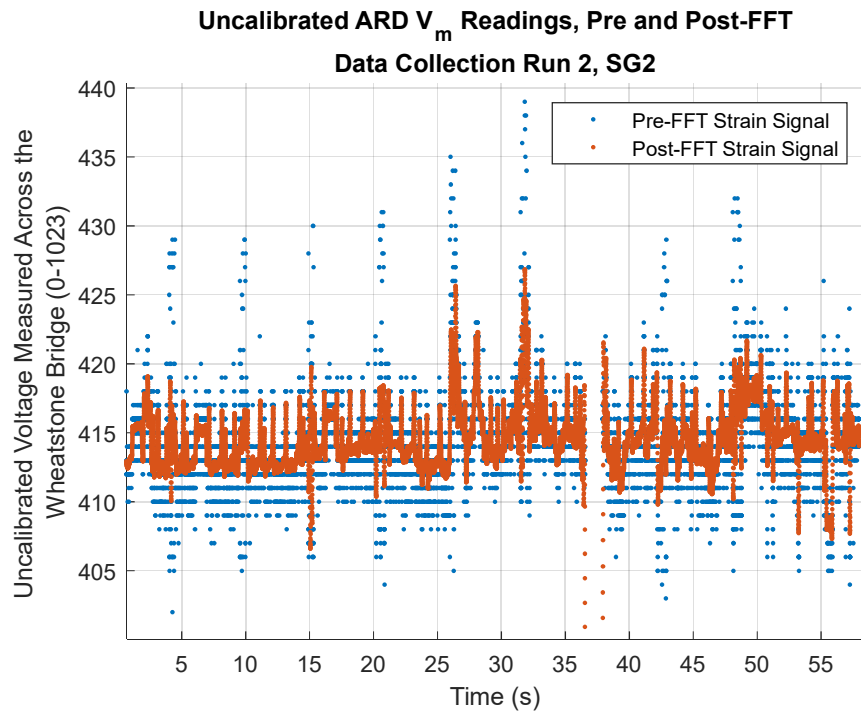


Figure 74: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 2, SG1.

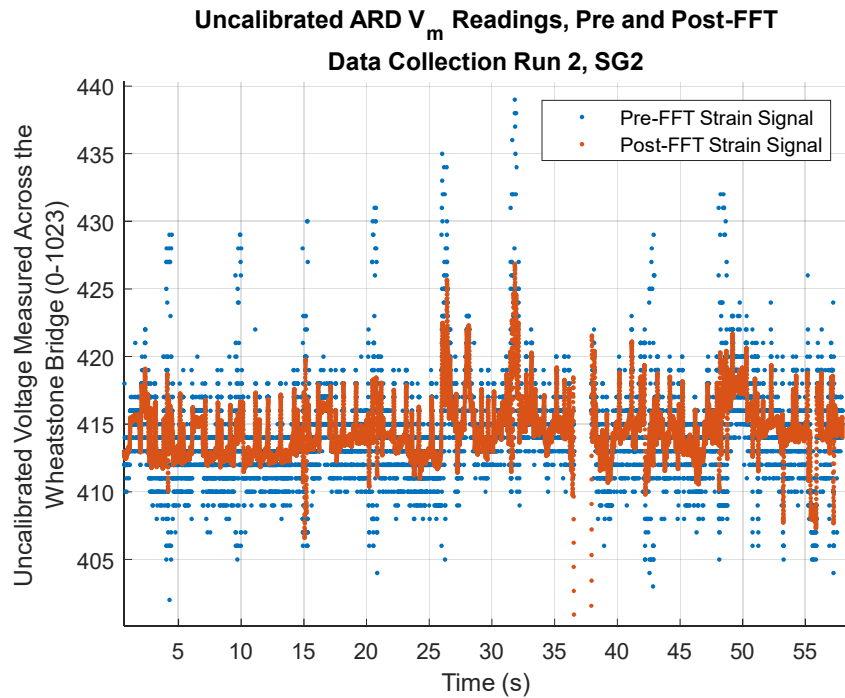


Figure 75: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 2, SG2.

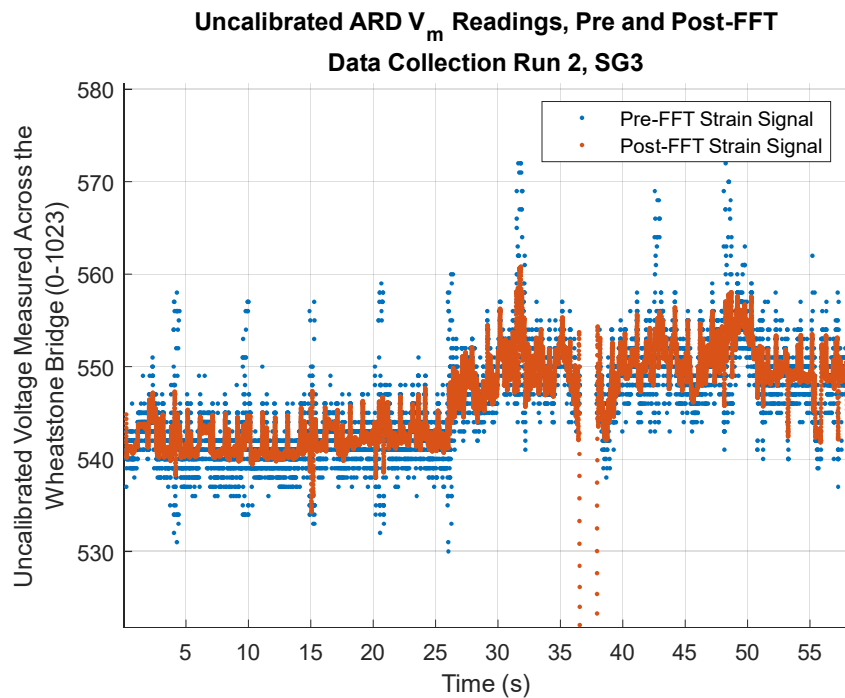


Figure 76: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 2, SG3.

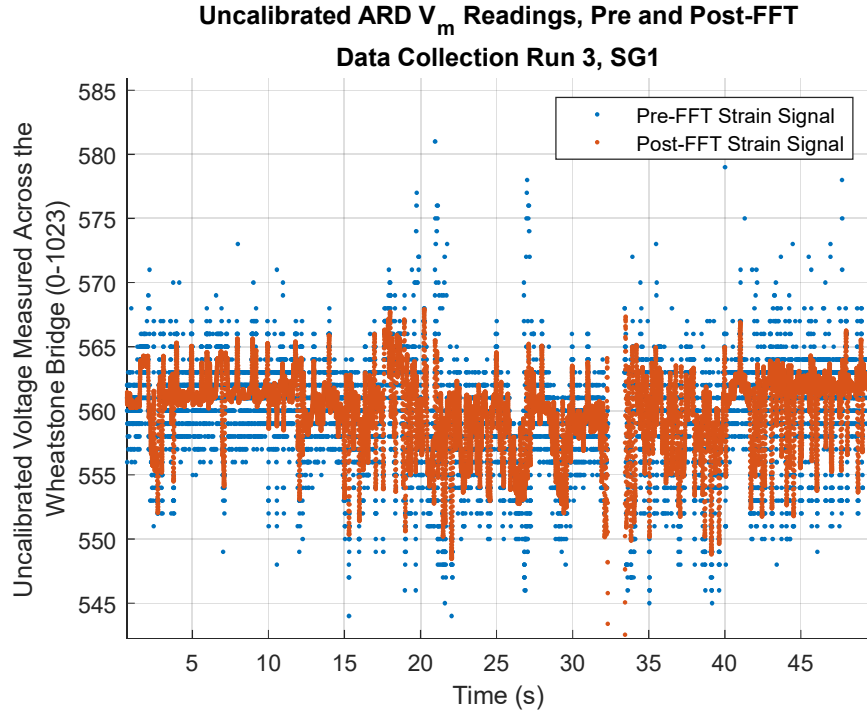


Figure 77: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 3, SG1.

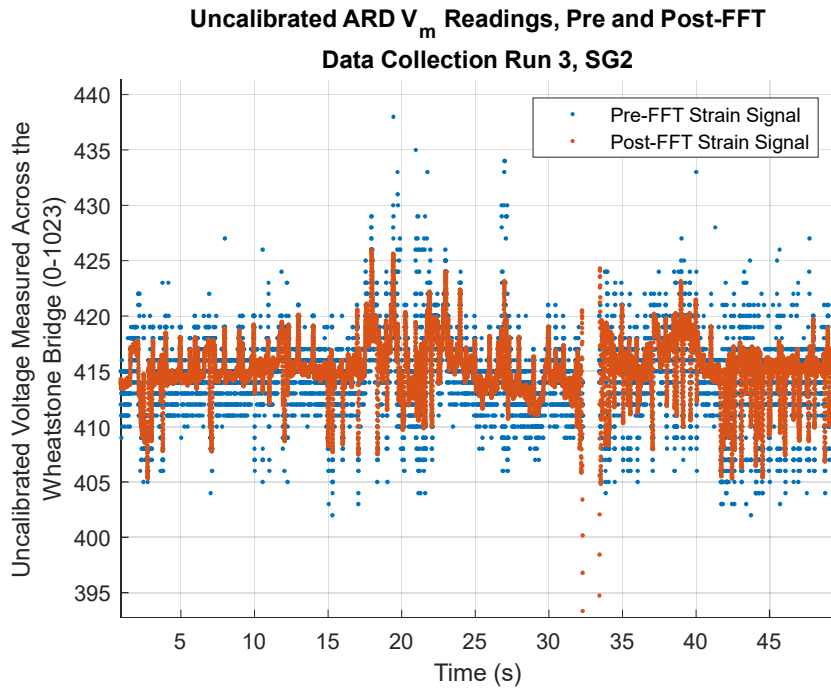


Figure 78: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 3, SG2.

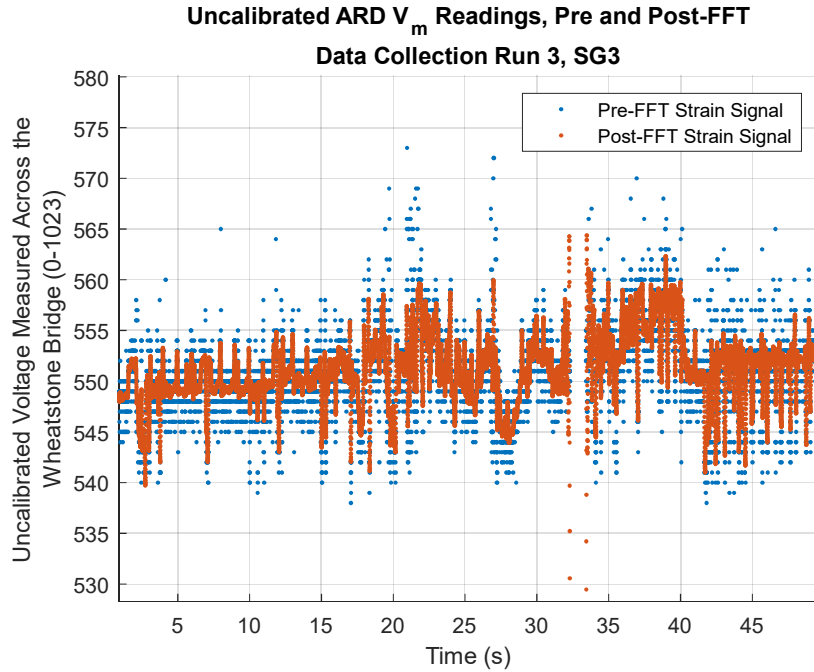


Figure 79: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 3, SG3.

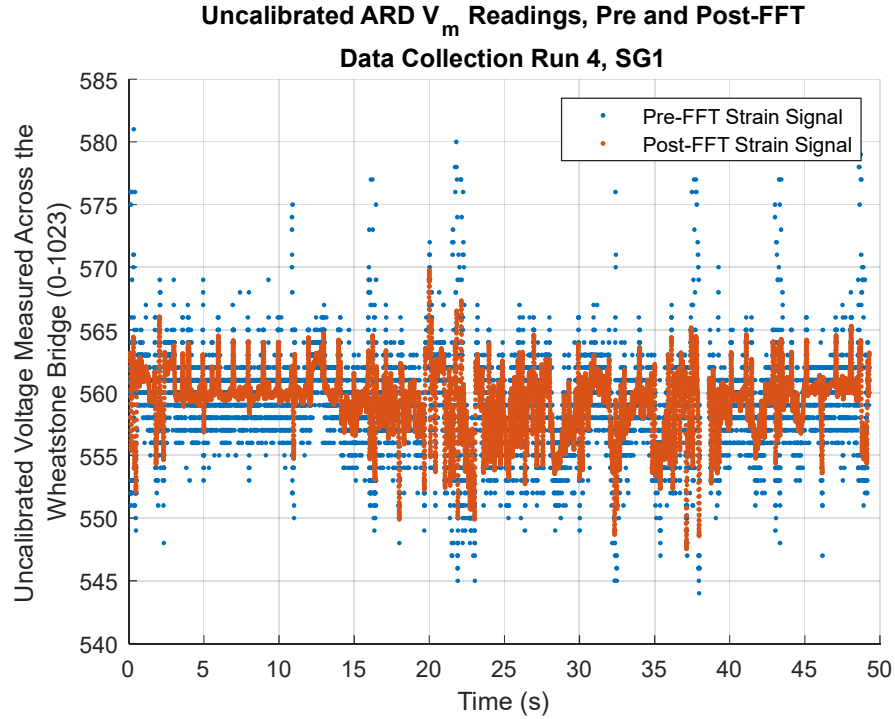


Figure 80: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 4, SG1.

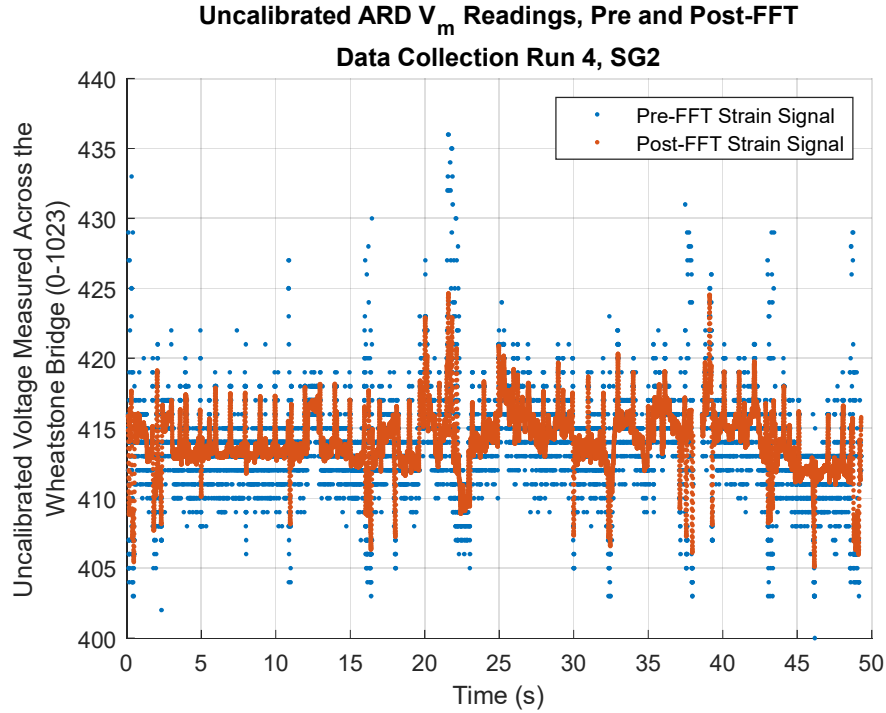


Figure 81: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 4, SG2.

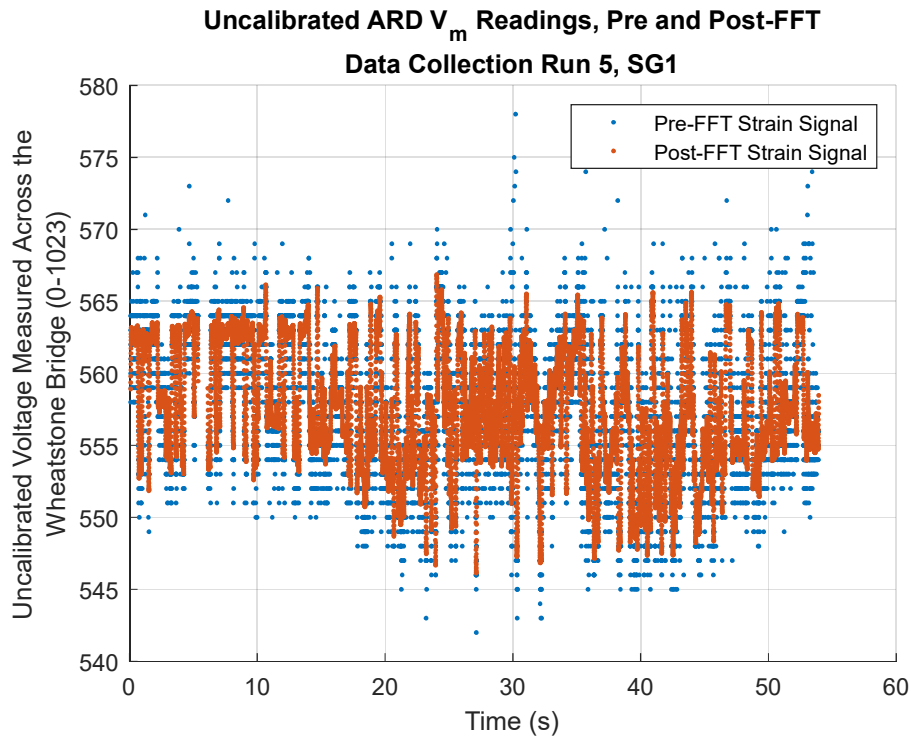


Figure 82: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 4, SG3.

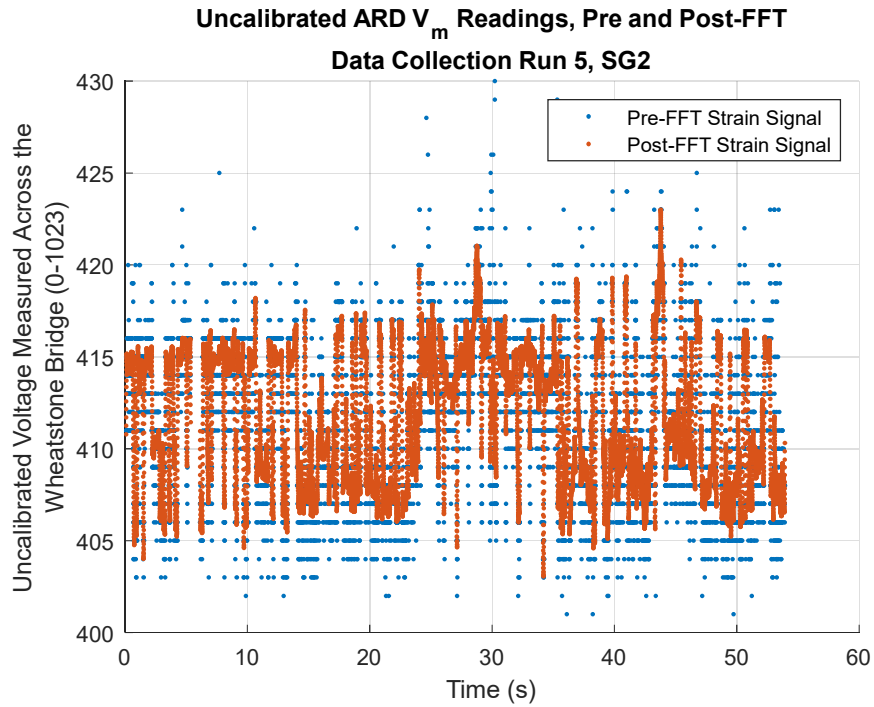


Figure 83: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 5, SG1.

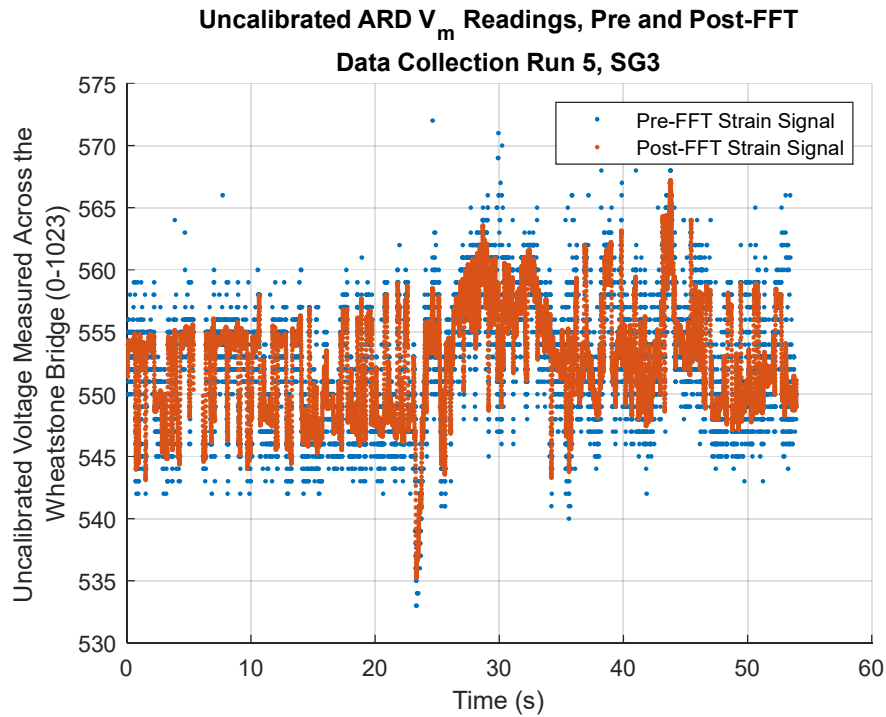


Figure 84: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 5, SG2.

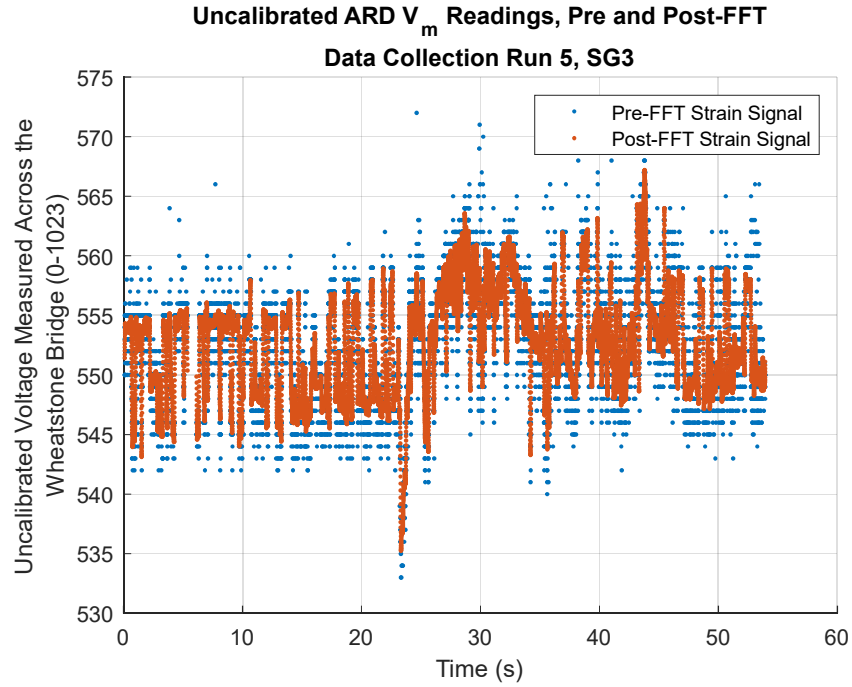


Figure 85: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 5, SG3.

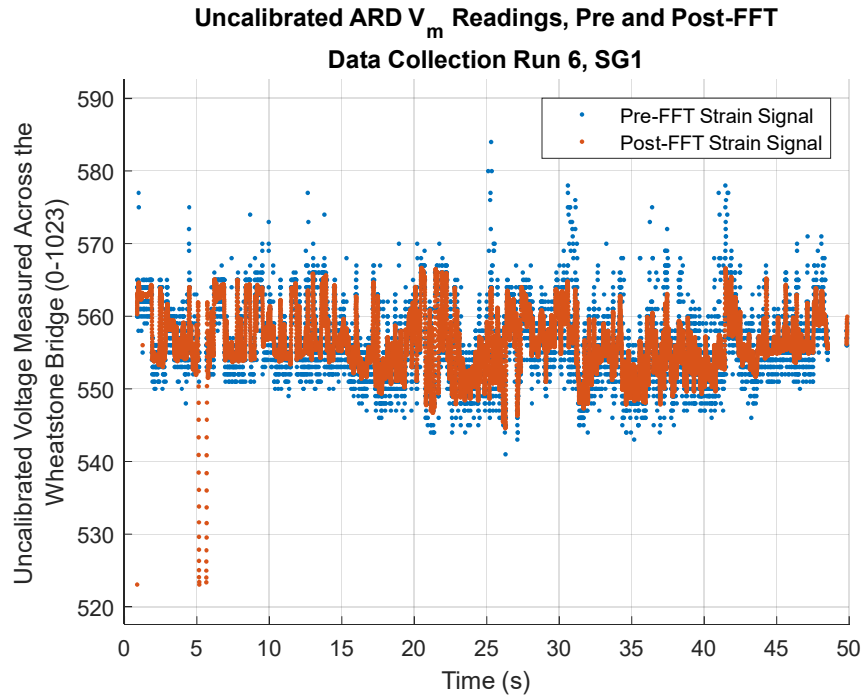


Figure 86: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 6, SG1.

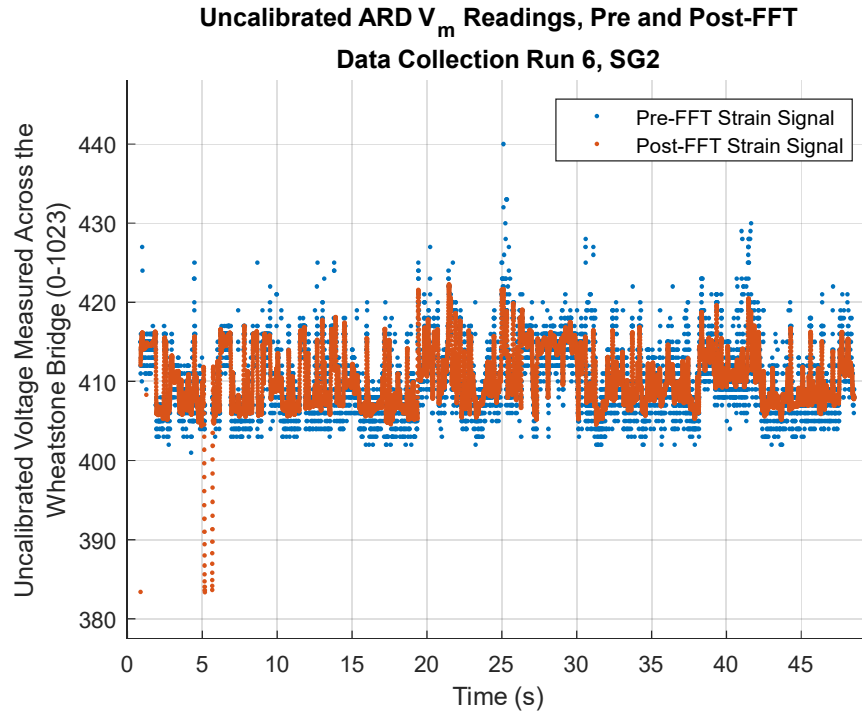


Figure 87: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 6, SG2.

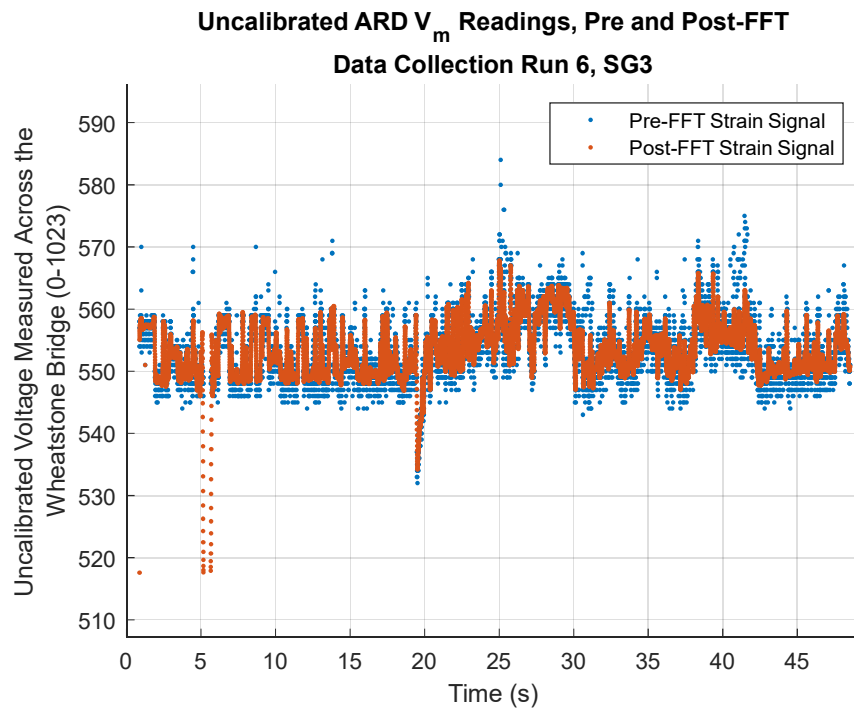


Figure 88: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 6, SG3.

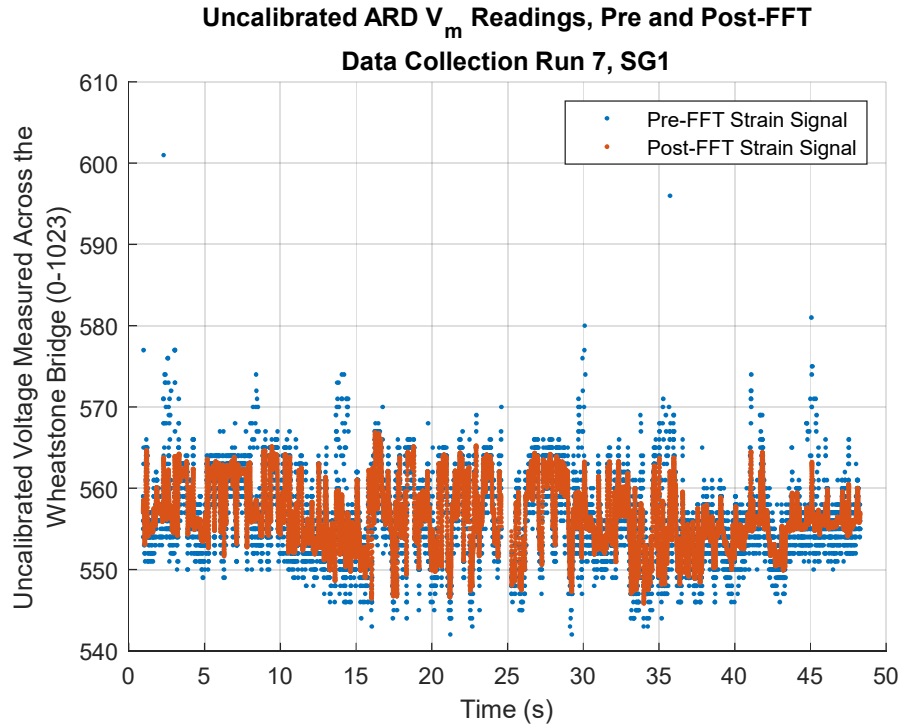


Figure 89: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 7, SG1.

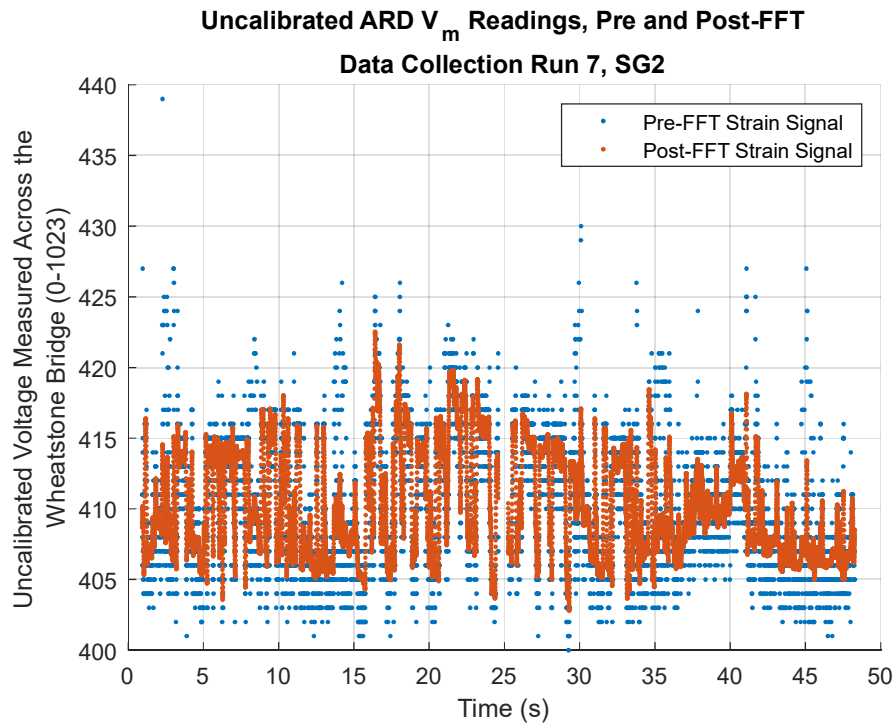


Figure 90: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 7, SG2.

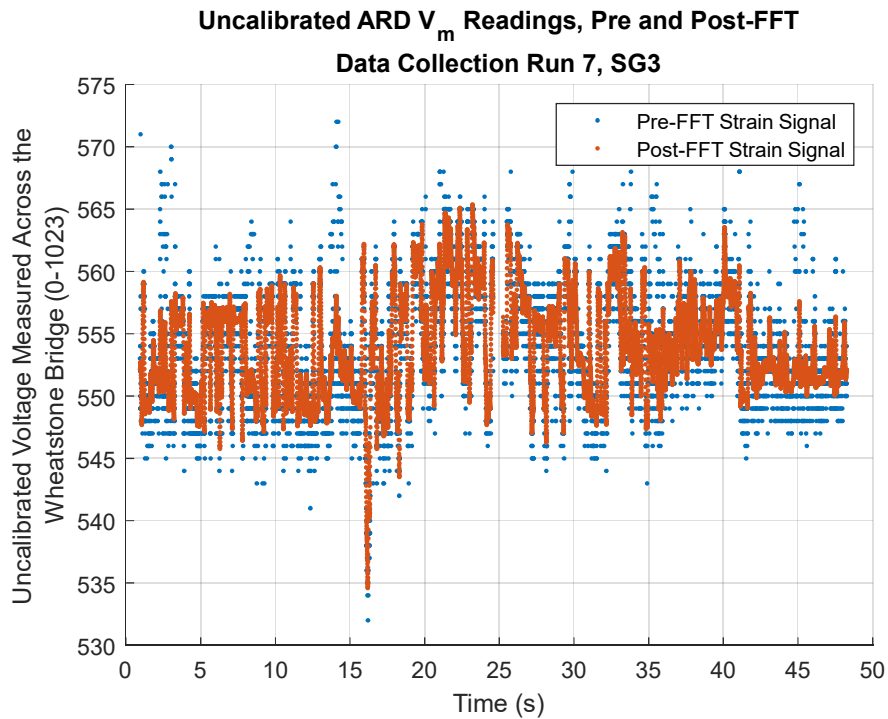


Figure 91: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 7, SG3.

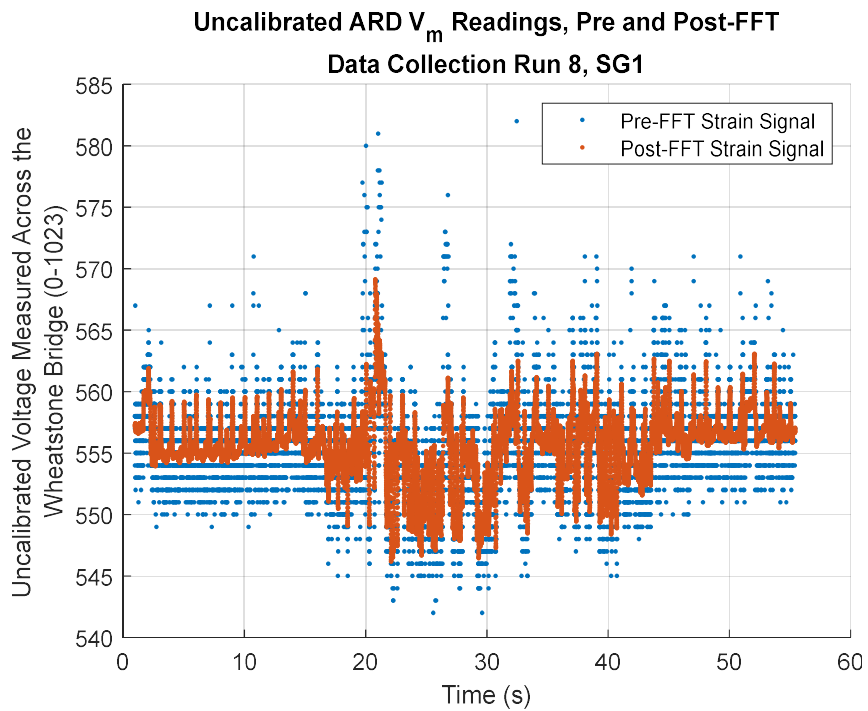


Figure 92: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 8, SG1.

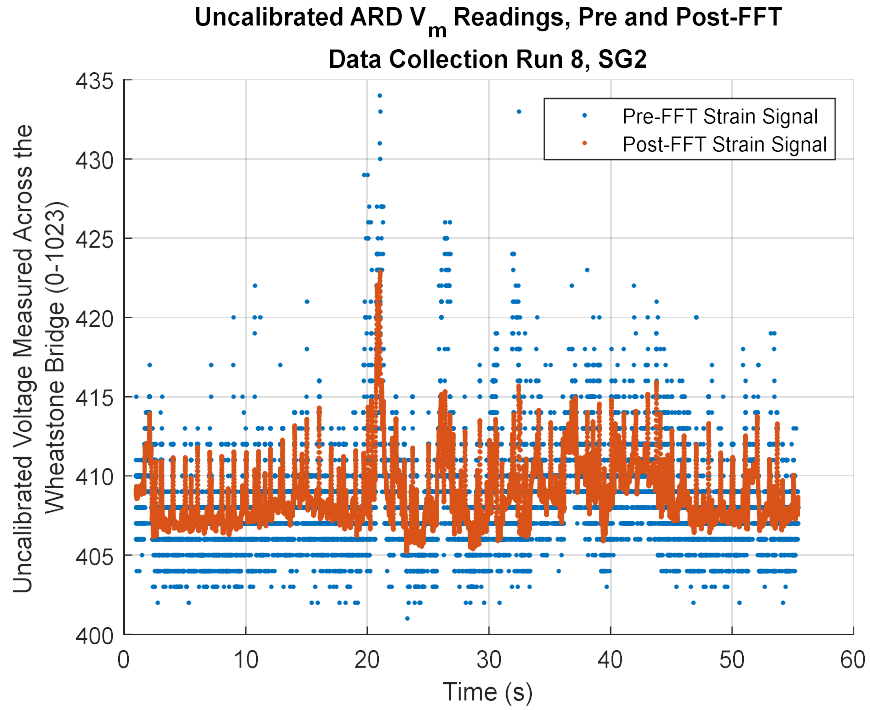


Figure 93: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 8, SG2.

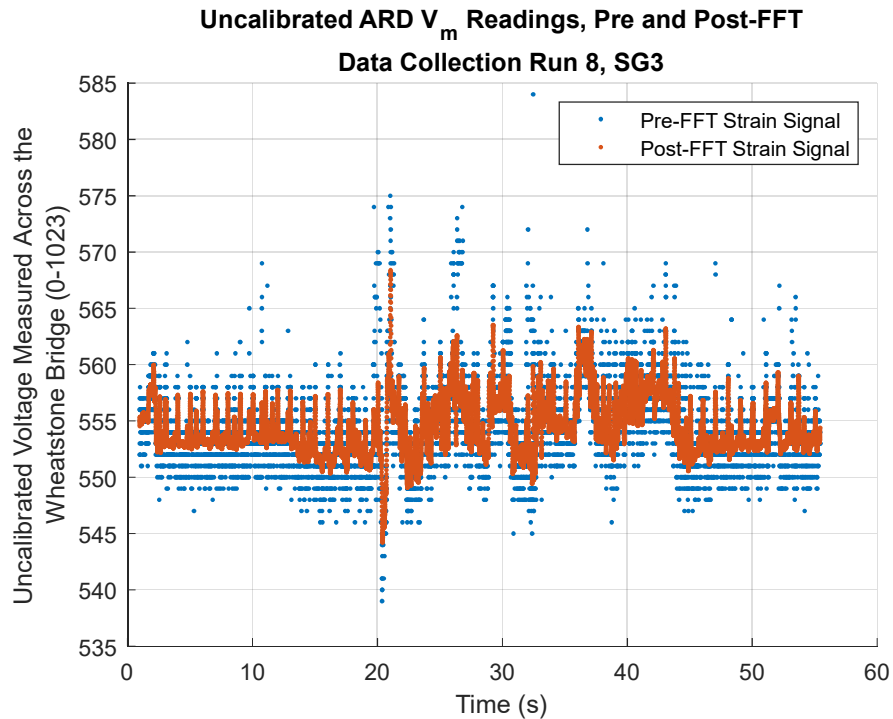


Figure 94: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 8, SG3.

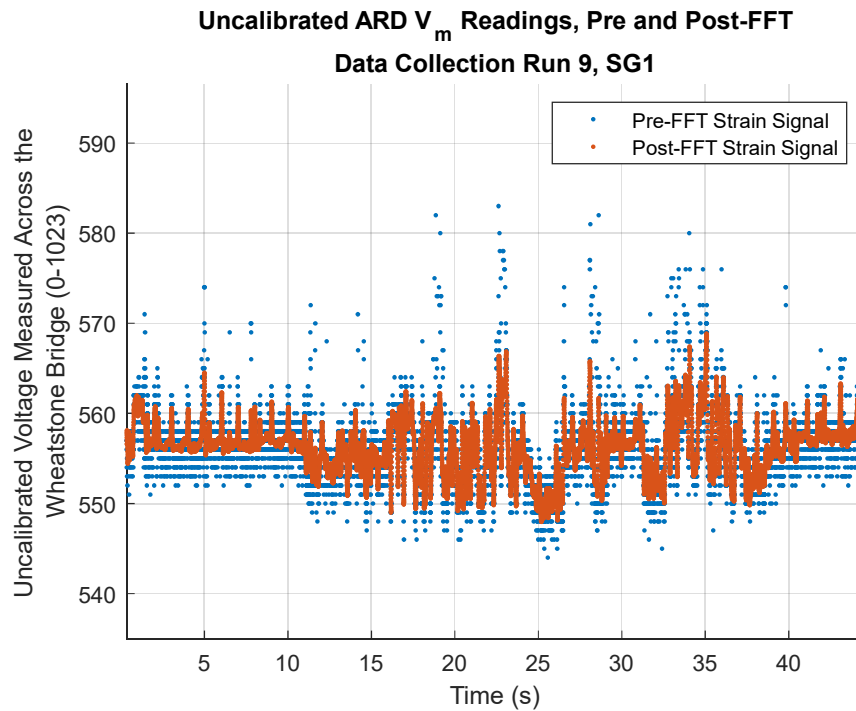


Figure 95: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 9, SG1.

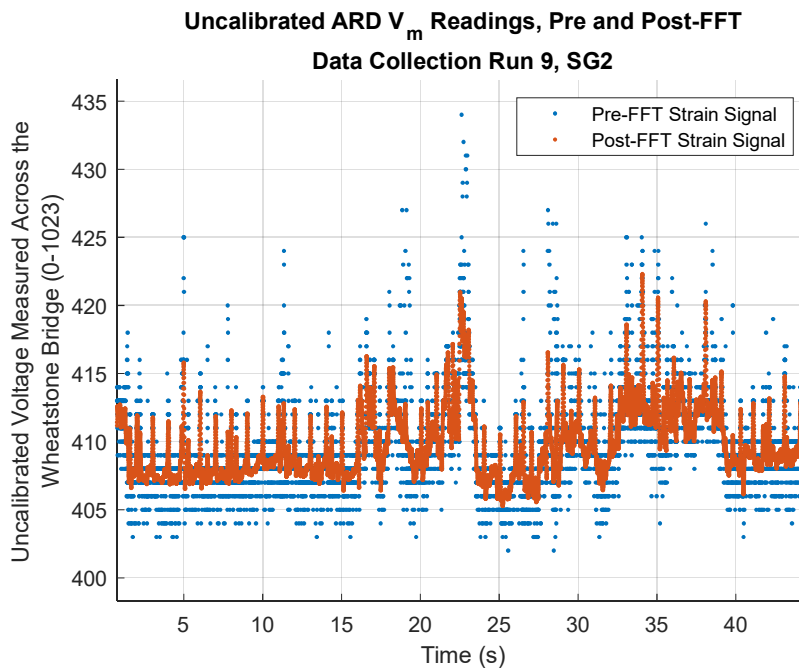


Figure 96: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 9, SG2.

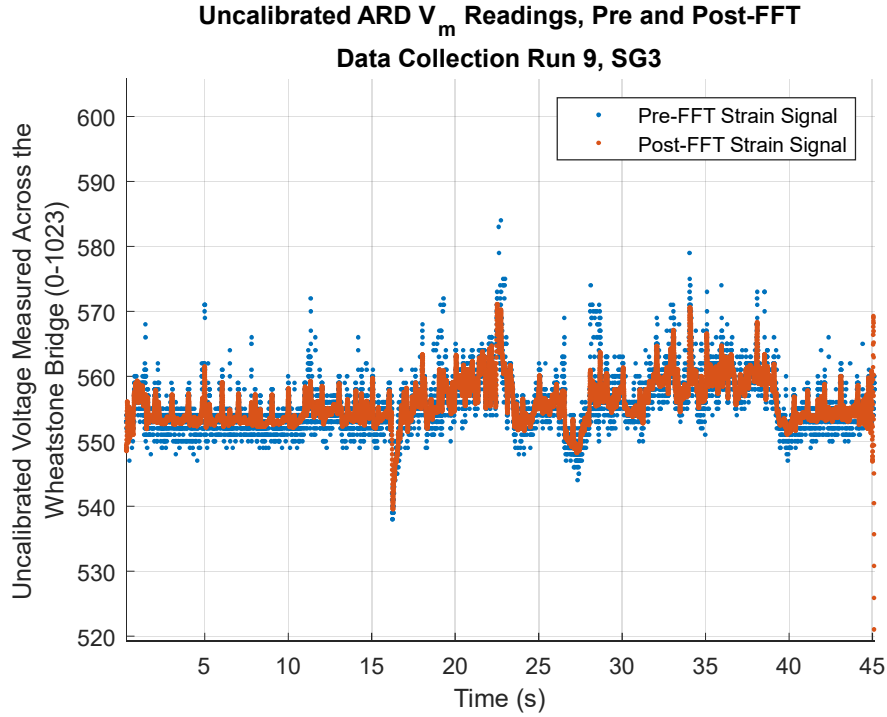


Figure 97: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 9, SG3.

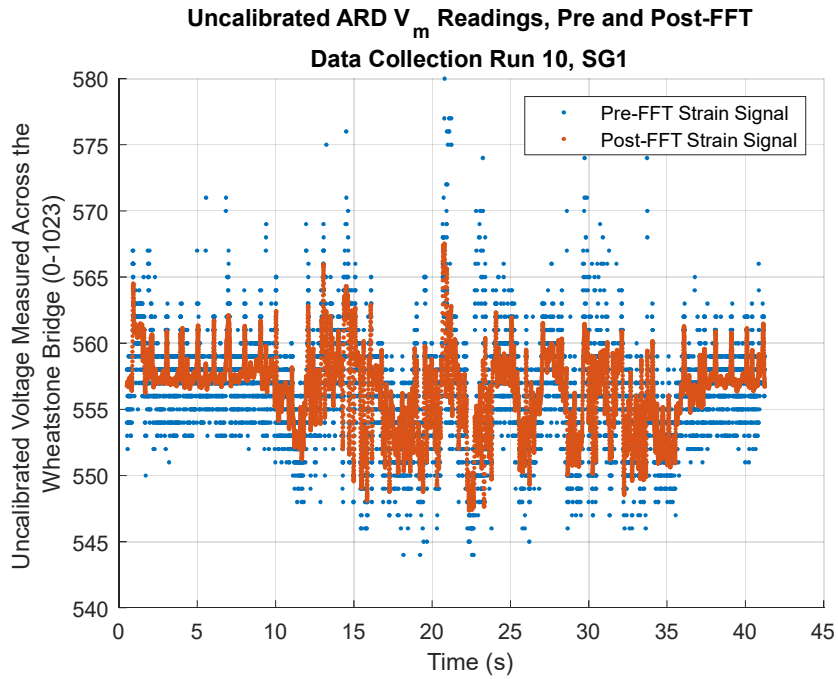


Figure 98: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 10, SG1.

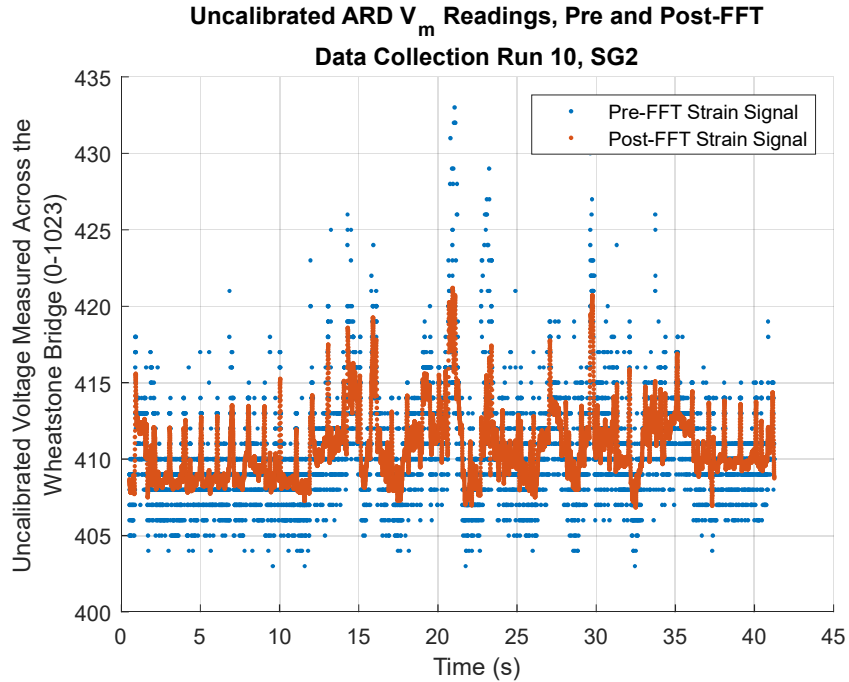


Figure 99: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 10, SG2.

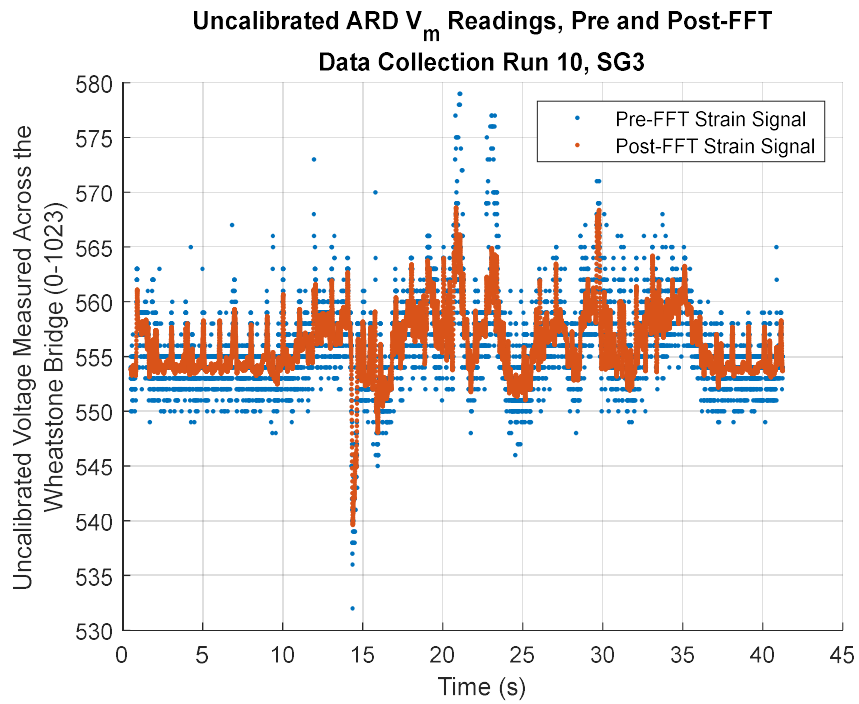


Figure 100: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 10, SG3.

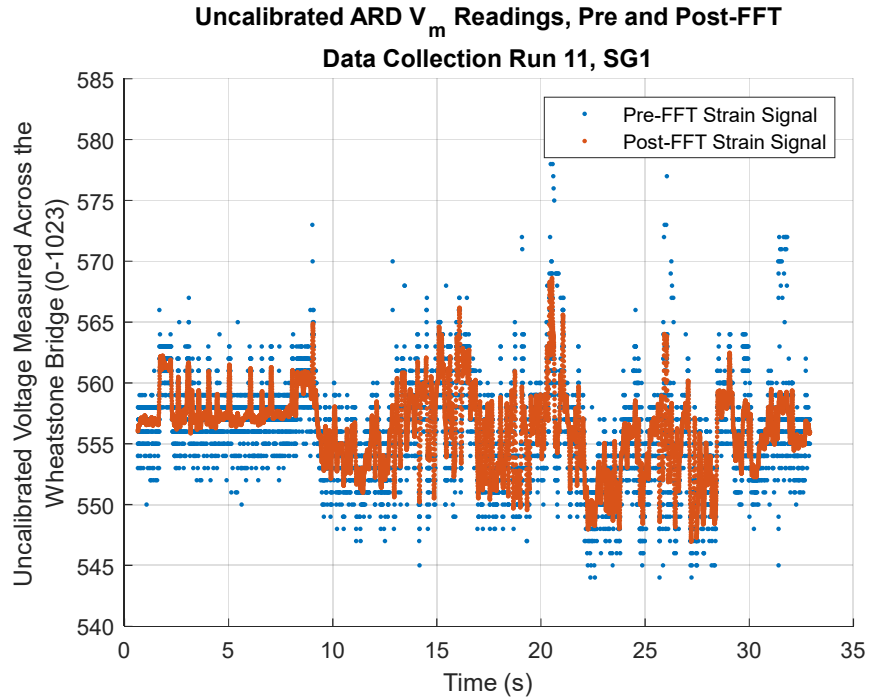


Figure 101: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 11, SG1.

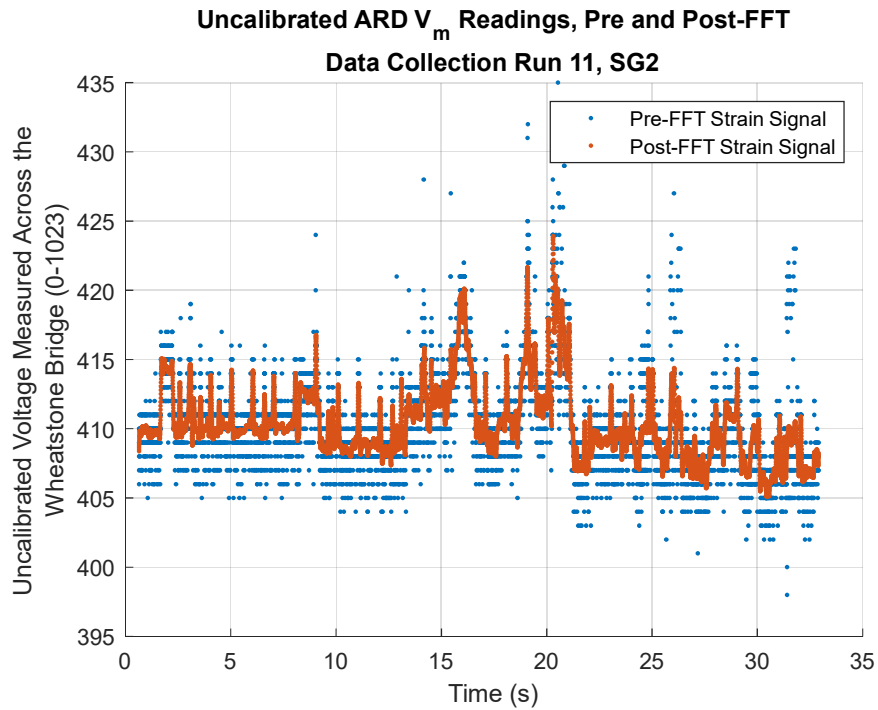


Figure 102: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 11, SG2.

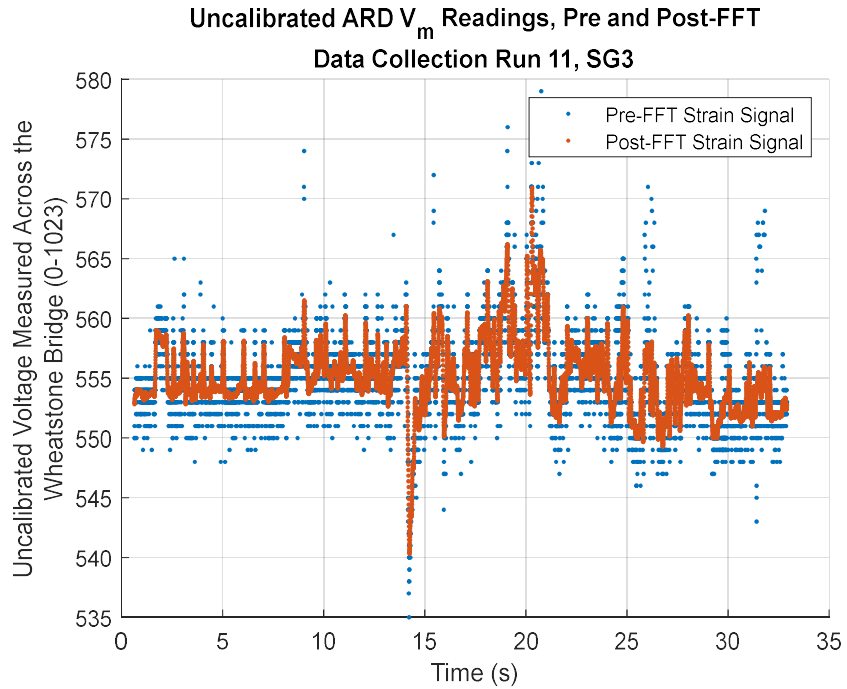


Figure 103: Uncalibrated ARD Measurement of Voltage Across the Wheatstone Bridge, V_{m_analog} , Pre-FFT and Post-FFT, Data Collection Run 11, SG3.

Event Window Plots. The bending and axial strain at SG1, SG2, and SG3 at the event window is plotted with velocity and the estimated parking calk location. Seen in Figure 104 to Figure 114.

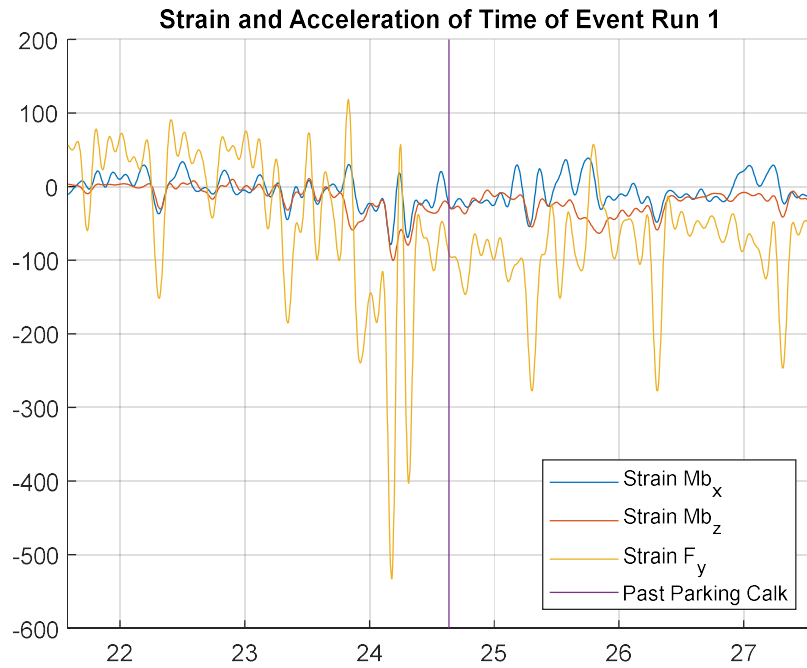


Figure 104: Strain Measured by Each Gauge During Event Run 1 with Parking Chalk Location Estimation and Velocity.

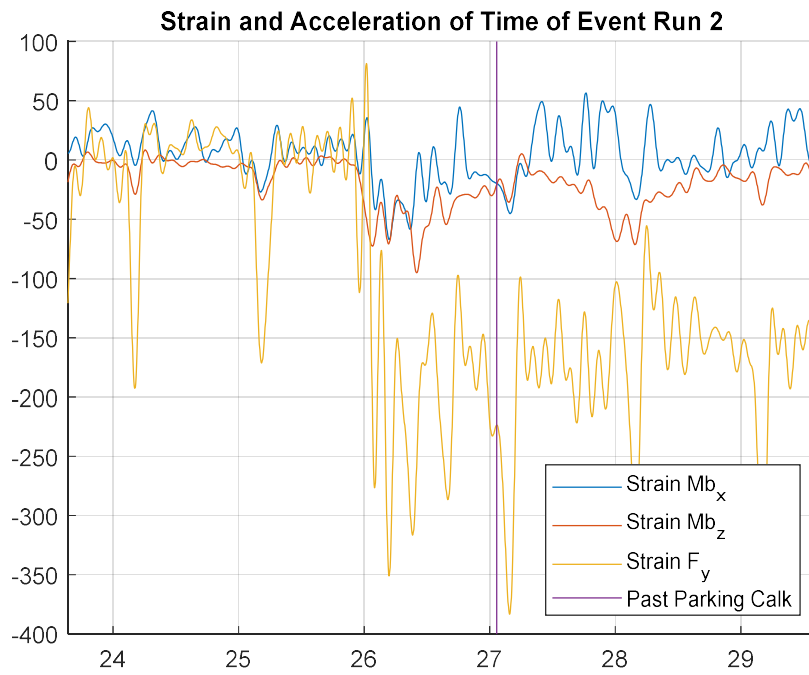


Figure 105: Strain Measured by Each Gauge During Event Run 2 with Parking Chalk Location Estimation and Velocity.

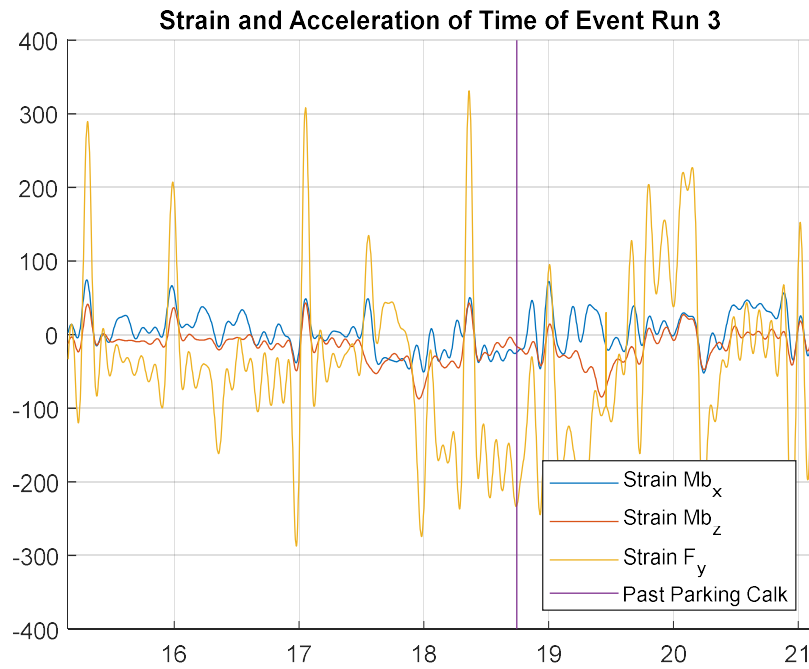


Figure 106: Strain Measured by Each Gauge During Event Run 3 with Parking Chalk Location Estimation and Velocity.

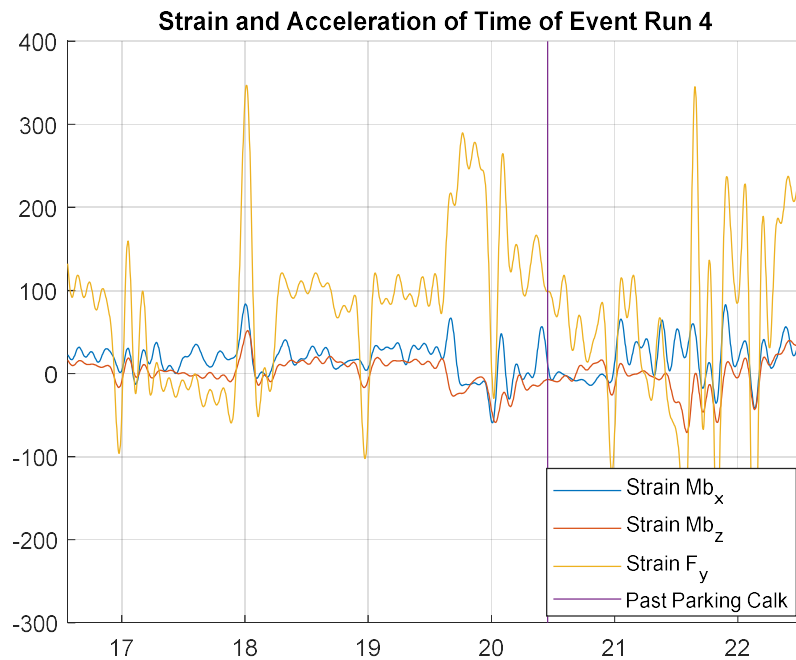


Figure 107: Strain Measured by Each Gauge During Event Run 4 with Parking Chalk Location Estimation and Velocity.

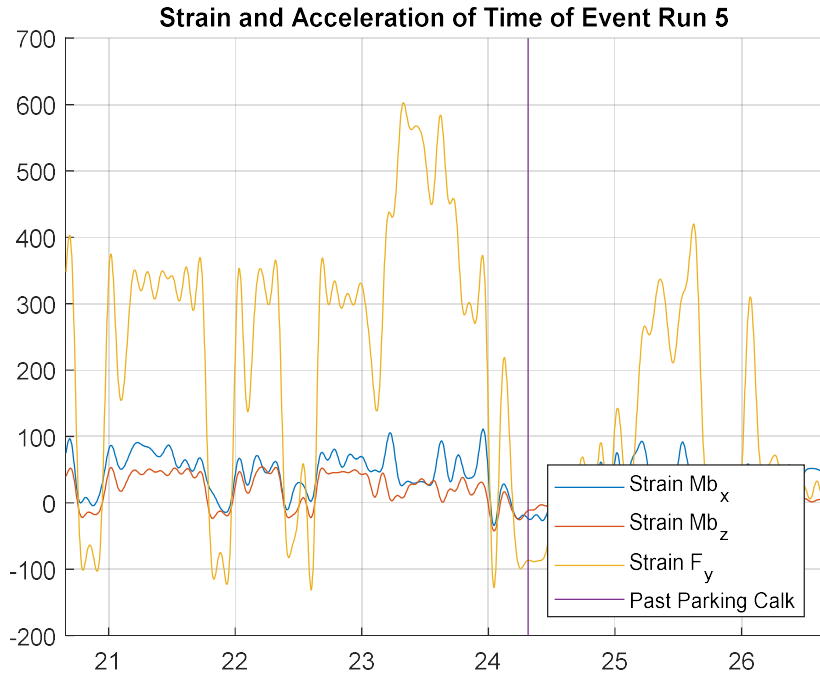


Figure 108: Strain Measured by Each Gauge During Event Run 5 with Parking Chalk Location Estimation and Velocity.

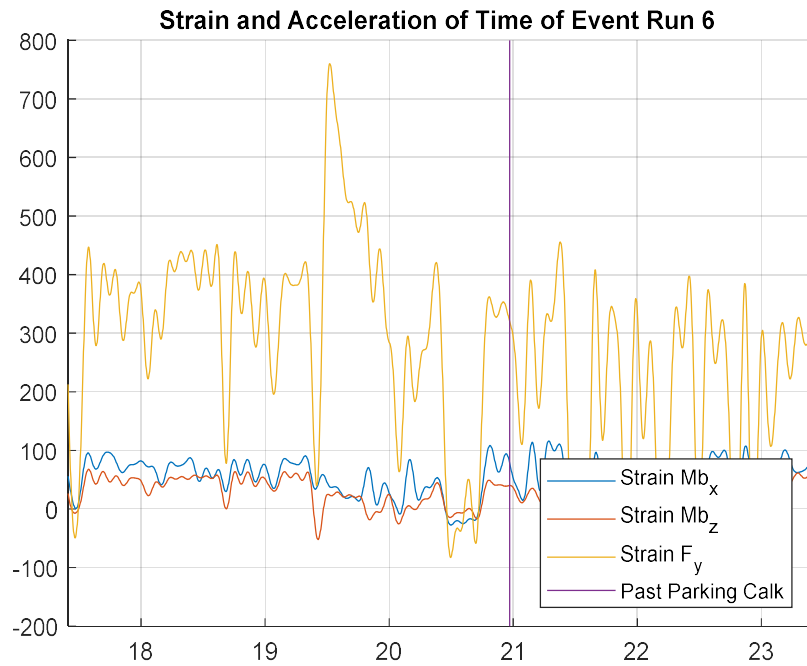


Figure 109: Strain Measured by Each Gauge During Event Run 6 with Parking Chalk Location Estimation and Velocity.

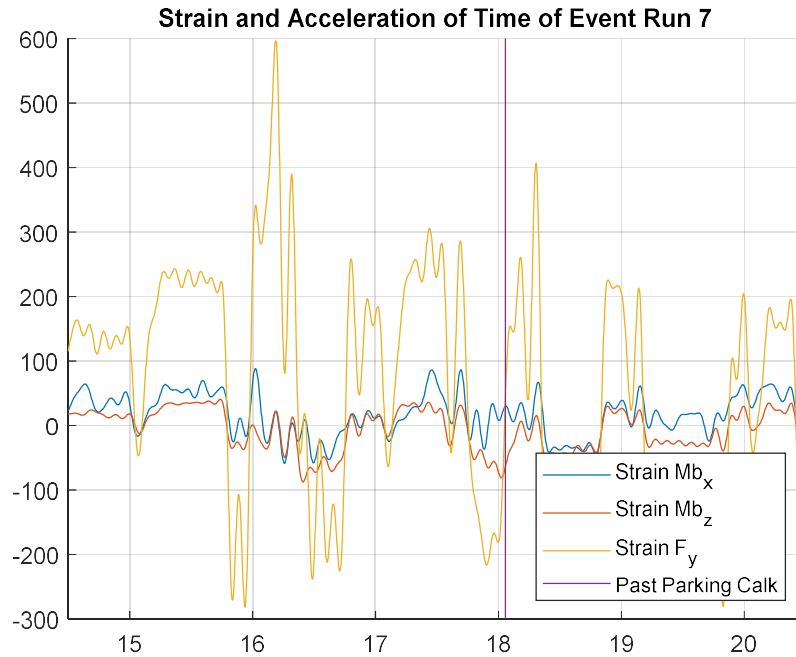


Figure 110: Strain Measured by Each Gauge During Event Run 7 with Parking Chalk Location Estimation and Velocity.

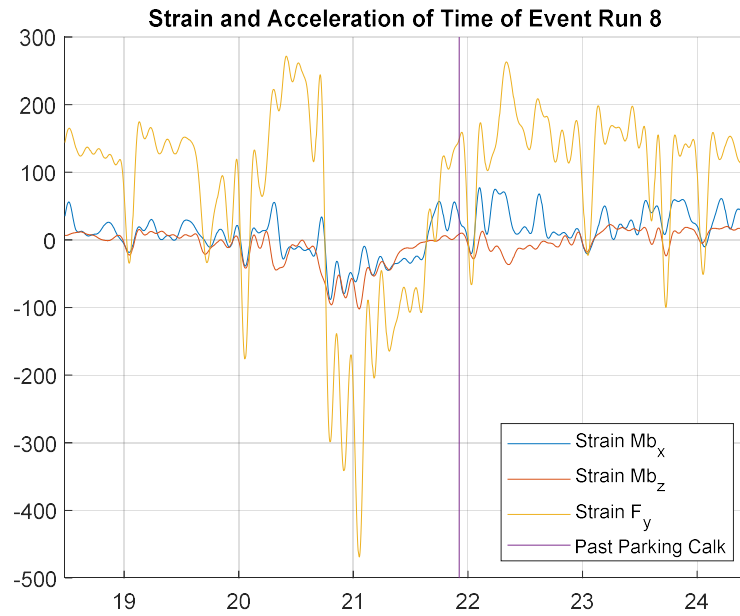


Figure 111: Strain Measured by Each Gauge During Event Run 8 with Parking Chalk Location Estimation and Velocity.

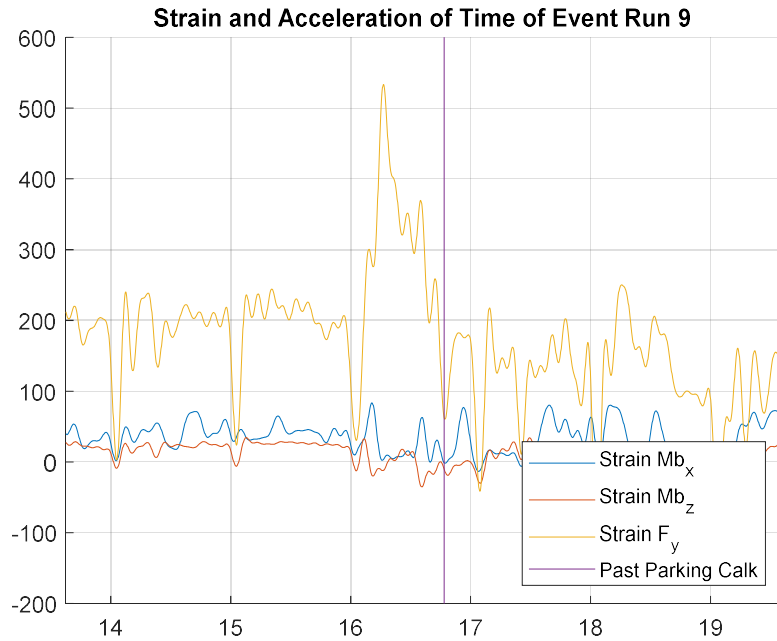


Figure 112: Strain Measured by Each Gauge During Event Run 9 with Parking Chalk Location Estimation and Velocity.

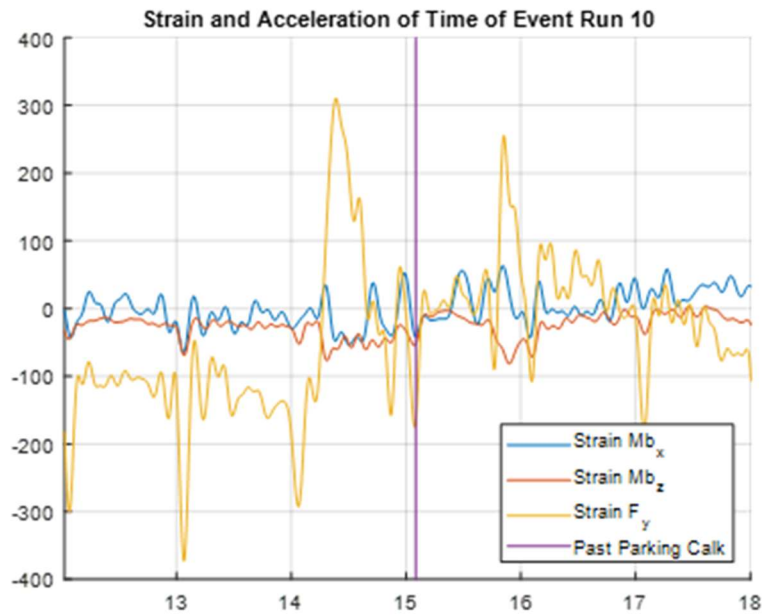


Figure 113: Strain Measured by Each Gauge During Event Run 10 with Parking Chalk Location Estimation and Velocity.

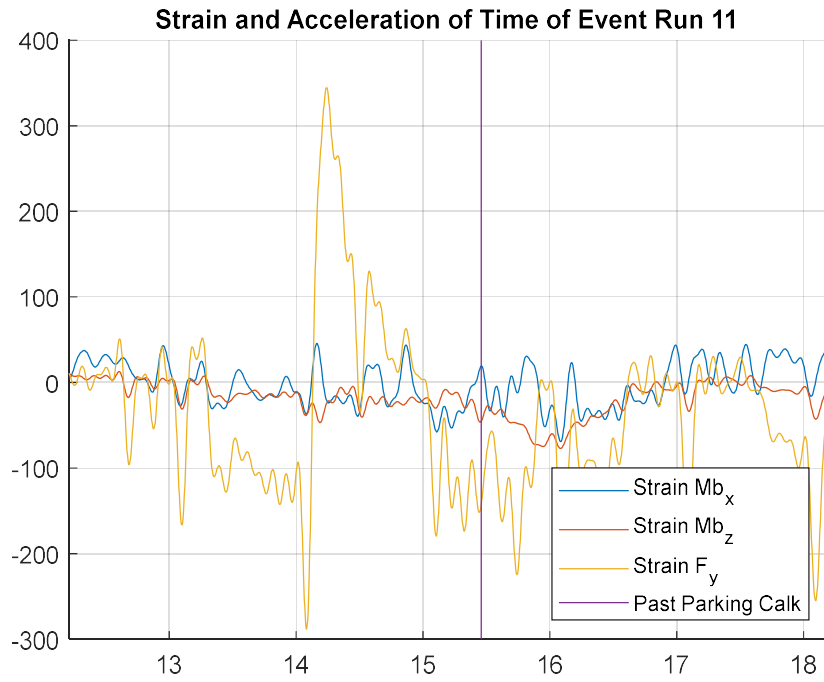


Figure 114: Strain Measured by Each Gauge During Event Run 11 with Parking Chalk Location Estimation and Velocity.