1-1-2016

Methods and Implementation of Fluid-Structure Interaction Modeling into an Industry-Accepted Design Tool

Donn R. Sederstrom
University of Denver

Recommended Citation

Sederstrom, Donn R., "Methods and Implementation of Fluid-Structure Interaction Modeling into an Industry-Accepted Design Tool" (2016). *Electronic Theses and Dissertations*. 1197.
https://digitalcommons.du.edu/etd/1197

Methods and Implementation of Fluid-Structure Interaction Modeling into an Industry-Accepted Design Tool

Abstract

Fluid-structure interaction (FSI) modeling is a method by which fluid and solid domains are coupled together to produce a single result that cannot be produced if each physical domain was evaluated individually. The work presented in this dissertation is a demonstration of the methods and implementation of FSI modeling into an industry-appropriate design tool. Through utilizing computationally inexpensive equipment and commercially available software, the studies presented in this work demonstrate the ability for FSI modeling to become a tool used broadly in industry.

To demonstrate this capability, the cases studied purposely include substantial complexity to demonstrate the stability techniques required for modeling the inherent instabilities of FSI models that contain three-dimensional geometries, nonlinear materials, thin-walled geometries, steep gradients, and transient behavior. The work also modeled scenarios that predict system failure and optimal design to extend service lifetime, thereby expanding upon current FSI literature. Four independent studies were performed, evaluating three separate modes of failure in FSI models, to demonstrate that FSI modeling is a viable design tool for widespread industry use.

The first study validates FSI modeling techniques by comparing the results of a thin-walled FSI geometry model under hydrostatic forces with existing experimental data.

The second study explored a parametric study that evaluated the factors influencing an FSI model containing a highly complex thermal-fluid fatigue model. This model involved dynamically changing temperature loads resulting in significant thermal expansion that led to material yielding and dynamic fatigue life.

The third study evaluated a thermal-fluid conjugate heat transfer problem. The model was tuned, validated, and optimized for lifetime, and the validation of the system was performed using experimental data.

The final study modeled the highly complex fluid and solid phenomena involved in a peristaltic pump where the goal was to demonstrate that the lifetime performance of the tubing could be altered by changing the geometry, material properties, and operating temperature. The model in this final study combined all the methods and techniques from the three earlier studies and applied them to a thin-walled tube geometry with nonlinear and temperature-dependent material properties to create large solid deformation and fluid motion.

Document Type

Dissertation

Degree Name

Ph.D.

Department

Mechanical Engineering

First Advisor

Corinne S. Lengsfeld, Ph.D.

Second Advisor

Peter Laz

Third Advisor

Ali Azadani

Keywords

Computational fluid dynamics, Finite element analysis, Fluid-solid interaction, Fluid-structure interaction, FSI, Multiphysics modeling

Subject Categories

Mechanical Engineering

Publication Statement

Methods and Implementation of Fluid–Structure Interaction Modeling into an Industry-Accepted Design Tool

---

A Dissertation

Presented to

the Faculty of the Daniel Felix Ritchie School of Engineering and Computer Science

University of Denver

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy

---

by

Donn R. Sederstrom

August 2016

Advisor: Corinne S. Lengsfeld

Author: Donn R. Sederstrom
Title: Methods and Implementation of Fluid–Structure Interaction Modeling into an Industry-Accepted Design Tool
Advisor: Corinne S. Lengsfeld
Degree Date: August 2016

ABSTRACT

Fluid–structure interaction (FSI) modeling is a method by which fluid and solid domains are coupled together to produce a single result that cannot be produced if each physical domain was evaluated individually. The work presented in this dissertation is a demonstration of the methods and implementation of FSI modeling into an industry-appropriate design tool. Through utilizing computationally inexpensive equipment and commercially available software, the studies presented in this work demonstrate the ability for FSI modeling to become a tool used broadly in industry.

To demonstrate this capability, the cases studied purposely include substantial complexity to demonstrate the stability techniques required for modeling the inherent instabilities of FSI models that contain three-dimensional geometries, nonlinear materials, thin-walled geometries, steep gradients, and transient behavior. The work also modeled scenarios that predict system failure and optimal design to extend service lifetime, thereby expanding upon current FSI literature. Four independent studies were performed, evaluating three separate modes of failure in FSI models, to demonstrate that FSI modeling is a viable design tool for widespread industry use.

The first study validates FSI modeling techniques by comparing the results of a thin-walled FSI geometry model under hydrostatic forces with existing experimental data.

The second study explored a parametric study that evaluated the factors influencing an FSI model containing a highly complex thermal-fluid fatigue model. This model involved dynamically changing temperature loads resulting in significant thermal expansion that led to material yielding and dynamic fatigue life.

The third study evaluated a thermal-fluid conjugate heat transfer problem. The model was tuned, validated, and optimized for lifetime, and the validation of the system was performed using experimental data.

The final study modeled the highly complex fluid and solid phenomena involved in a peristaltic pump where the goal was to demonstrate that the lifetime performance of the tubing could be altered by changing the geometry, material properties, and operating temperature. The model in this final study combined all the methods and techniques from the three earlier studies and applied them to a thin-walled tube geometry with nonlinear and temperature-dependent material properties to create large solid deformation and fluid motion.

ACKNOWLEDGEMENTS

I would like to thank my advisor Dr. Corinne Lengsfeld for her continued support, encouragement, and willingness to share her knowledge in the field of engineering and computational modeling. I would also like to thank Dr. Peter Laz for his guidance on optimization, material behavior, and computational modeling. I would like to recognize the current and past students in the computational modeling groups at the University of Denver for your advice, guidance, and company.

I would like to thank my family, parents, and past educators that have provided me with constant support by encouraging me to not accept limits and reminding me that anything can be accomplished with hard work. I would like to give a special thank you to my partner and teammate, Jessica, for her daily support and reminders. Your ever present tokens of support helped me push through to the end. This journey would truly not have been the same without you.

Finally, I would like to thank the Research and Development team at Johns Manville for funding as well as sharing their knowledge and desire to push the limits of computational modeling.

TABLE OF CONTENTS

Abstract ii

Acknowledgements iv

Table of Contents v

List of Tables ix

List of Figures xi

Chapter 1: Introduction 1
  1.1. FSI Modeling Methods 2
    1.1.1. What is FSI Modeling? 2
    1.1.2. System of Equations 3
    1.1.3. Mesh Coupling Methods 6
    1.1.4. Data Transfer Methods 10
    1.1.5. Mass Effect 16
  1.2. Why FSI Modeling is Difficult 18
    1.2.1. FSI Modeling History 19
    1.2.2. Thin-Walled Bodies 20
    1.2.3. Computational Instabilities and Benchmarking 20
    1.2.4. Validated FSI Models 21
    1.2.5. Material Properties 22
    1.2.6. Solid Contact in FSI Models 22
  1.3. Advancements in Computational Modeling 23
    1.3.1. CFD Modeling 23
    1.3.2. CFD Limitations 24
    1.3.3. FSI Modeling 24
    1.3.4. Modeling System Life 28
  1.4. Dissertation Overview 28
    1.4.1. Motivation 28
    1.4.2. Research Question 30

Chapter 2: Validation of a Thin-Walled Fluid–Structure Interaction Model with Experimental Results 33
  2.1. Abstract 33
  2.2. Introduction 34
  2.3. Problem Description 34
  2.4. Methods 36
    2.4.1. Experimental Methods 37
    2.4.2. CFD Numerical Methods 38
    2.4.3. FEA Numerical Methods 42

2.4.4. Interface between Fluid and Solid Domains.. 44
2.4.5. Solid Material Properties... 47
2.4.6. Optimization of Material Properties.. 48
2.4.7. Final Multiphysics Model.. 52
2.5. Results and Discussion 54
2.5.1. CFD Domain Results.. 54
2.5.2. FEA Results.. 55
2.5.3. Forced Coupled Model 57
2.5.4. Experimental Results... 58
2.5.5. Material Properties Optimization Results...................................... 60
2.5.6. Demonstration of Model Robustness with Water.......................... 64
2.6. Conclusion 66
2.7. Lessons Learned 66

Chapter 3: Lifetime Design for Coupled Fluid–Structure Interaction Model Under Thermal-Cyclic Loading 68
3.1. Abstract... 68
3.2. Introduction... 69
3.3. Problem Normalization.. 70
3.4. Problem Description 71
3.5. Methods 72
3.5.1. CFD Numerical Methods.. 73
3.5.2. FEA Numerical Methods.. 76
3.5.3. Fatigue Life Numerical Methods... 79
3.5.4. Interface between Fluid and Solid Domain 81
3.5.5. Parametric Study Methods.. 82
3.6. Results... 83
3.6.1. Steady-State Material Properties 83
3.6.2. Computational and Experimental Comparison.............................. 84
3.6.3. Strain-Life Analysis.. 90
3.6.4. Parametric Study.. 92
3.7. Conclusion 94
3.8. Lessons Learned 95

Chapter 4: Lifetime Design for Thermal-Fluid System.................................... 96
4.1. Abstract... 96
4.2. Introduction... 97
4.3. Problem Description 97
4.3.1. Problem Normalization... 98
4.3.2. Thermal System Failure.. 99
4.3.3. Design Constraints... 99
4.4. Methods 100
4.4.1. Experimental Methods... 101
4.4.2. CFD Numerical Methods.. 109

    - 4.4.3. Tuning the Computational Model 113
    - 4.4.4. Design Study 115
  - 4.5. Results 117
    - 4.5.1. Experimental Thermal Conductivity 117
    - 4.5.2. Computational Model Tuning 118
    - 4.5.3. Validation 120
    - 4.5.4. Design Study 122
  - 4.6. Conclusion 126
  - 4.7. Lessons Learned 126

Chapter 5: Lifetime Design of a Peristaltic Pump Using Fluid–Structure Interaction modeling 128
  - 5.1. Abstract 128
  - 5.2. Peristaltic Pump Background 129
    - 5.2.1. Peristaltic Pump Operation 130
    - 5.2.2. Peristaltic Pump Tubing 131
  - 5.3. Problem Description 135
  - 5.4. Methods 137
    - 5.4.1. CFD Numerical Methods 137
    - 5.4.2. FEA Numerical Methods 140
    - 5.4.3. Peristaltic Pump Tubing Properties 145
    - 5.4.4. FSI Numerical Methods 147
    - 5.4.5. Failure Analysis 150
    - 5.4.6. Parametric Study 150
  - 5.5. Results 152
    - 5.5.1. Base Condition Results 152
    - 5.5.2. Parametric Study 164
  - 5.6. Conclusion 165
  - 5.7. Lessons Learned 166

Chapter 6: Conclusions and Recommendations 167
  - 6.1. Novel Contributions and Limitations of Each Study 168
    - 6.1.1. Study 1: FSI Flap Validation 168
    - 6.1.2. Study 2: FSI Model with Thermal-Cyclic Loading 169
    - 6.1.3. Study 3: Thermal-Fluid Lifetime Design 170
    - 6.1.4. Study 4: FSI Model of Peristaltic Pump 171
  - 6.2. Conclusion 173

References 175

Appendices 186
  - Appendix A: Matlab Optimization Scripts for Study 1 186
  - Appendix B: Fluent User-Defined Functions for Study 2 199

Appendix C: Fluent User-Defined Functions for Study 3 210
Appendix D: ANSYS Mechanical APDL Custom Scripting for Study 4 213

LIST OF TABLES

Table 1.1: List of studies with details about key coupling features, available experimental data, and methods of validation. 32

Table 2.1: Fluid properties for the water and oil used in the computational models [4,74]. 40
Table 2.2: Parameters used for initial meshing, dynamic remeshing, and dynamic smoothing. 46
Table 2.3: Published material properties data for polyethylene-based materials [77]. 47
Table 2.4: Optimization search setup for optimization and equivalent values for the FSI model. 51
Table 2.5: Boundary conditions settings for final multiphysics model for both the fluid and solid domains. 54
Table 2.6: Optimization starting points, optimal solution, and search information for the material properties optimization. 61

Table 3.1: Normalization units and equations for presented data. 70
Table 3.1: Fluid and solid domain boundary conditions. 76
Table 3.2: Material properties used for computational modeling taken from a nickel-based steel alloy [81]. 77
Table 3.3: Stainless steel fatigue life properties [82]. 77
Table 3.4: Parametric study properties setup. 83
Table 3.5: Results from parametric study for alternative geometry changes, applied thermal conditions, material properties, and three-dimenstional temperature hot spots. 92

Table 4.1: Normalization units and equations for presented data. 99
Table 4.2: Avalible material properties for polyurethane insulation and air [90–96]. 102
Table 4.3: Material properties used in the computational model. 111
Table 4.4: Published specific heat bounds for 20% gelatin gel compared to computationally tuned specific heat value [84]. 114
Table 4.5: External boundary conditions for the top, side, and bottom exterior surfaces of the vessel when testing occurred in the thermal chamber. 115
Table 4.6: Values used to calculate the composite thermal conductivity of the vessel. 118

Table 5.1: Material properties for tubing used in peristaltic pumps from a variety of sources, including journal publications, Master’s theses, and manufacturing specifications. 132
Table 5.2: Published fatigue life properties for tubing used in peristaltic pumps from a variety of sources, including published data and manufacturing specifications. 135
Table 5.3: Material properties of water used in the CFD model. 139
Table 5.4: Boundary conditions for fluid and solid domains. 139

Table 5.5: The material properties for the linear, upper, and lower bounds were defined using a three parameter Mooney–Rivlin hyperelastic model. 146
Table 5.6: Dynamic meshing conditions. 149
Table 5.7: Cases investigated during the parametric study... 152
Table 5.8: Results for the parametric study including flow rate, stress, strain, and lifetime. 165

LIST OF FIGURES

Figure 1.1: Black lines represent the fluid mesh (Eulerian) and gray body represents the solid mesh (Lagrangian). Mesh motions with the fictitious domain method are shown from a to b, and mesh motions with the Arbitrary Lagrangian–Eulerian method is shown from c to d. 9
Figure 1.2: Conformal meshing interface (a) and non-conformal meshing interface (b). 10
Figure 1.3: Two-way monolithic time stepping for FSI modeling. 13
Figure 1.4: Two-way strong coupling algorithm used to transfer data back and forth between fluid and solid models. 14
Figure 1.5: Two-way weak coupling algorithm used to transfer data back and forth between fluid and solid models. 15
Figure 1.6: One-way data transfer for fluid to solid only (left) and solid to fluid only (right). 16
Figure 1.7: The mass effect experienced at the interface between a stiff solid model and an incompressible fluid model. 17

Figure 2.1: Schematic of polyethylene-based hyperelastic flap acting as a dam that resists the hydrostatic forces produced by the fluid column. 35
Figure 2.2: Image of test fixture with the 1/16 inch trapezoid flap with dimensions. 36
Figure 2.3: Deflected flap with three displacement points for measuring computational (left) and experimental (right) results. 38
Figure 2.4: Image of the initial mesh before deformation occurs (left) and 0.85 seconds (right). A finer mesh was desired at the fluid inlet and outlet and along the walls of the flap, while a courser mesh was desired through the bulk of the fluid. 40
Figure 2.5: Ramping point load as a function of time applied at the upper edge of the hyperelastic flap. 43
Figure 2.6: Diagram of the optimization controlled by Matlab that managed the FSI software linked by ANSYS System Coupling and custom scripts. 50
Figure 2.7: Forces as a function of time with time steps at 0.00125 and 0.005 seconds. If time steps are too small then oscillations are present, but if time steps are selected appropriately the oscillations are not observed. 53
Figure 2.8: Volume of fluid contour plot for oil and air for the uncoupled CFD model with the fluid level at 40 mm. 55
Figure 2.9: Results from mesh density study indicated four HEX20 elements in the bending direction produced accurate deflection results, but six elements were used. 56
Figure 2.10: Image of the desired mesh after a mesh density study was performed. 56
Figure 2.11: Deflection contour plot of the hyperelastic flap by a ramping point load for the uncoupled FEA model. 57
Figure 2.12: Volume of fluid contour plot for the forced coupled fluid domain (left) and x-directional deflection of the solid domain (right) at 0.52 seconds. 58

Figure 2.13: Experimental results for deflection of three points at five fluid levels using oil with vertical error bars at ±1 standard deviations and horizontal error bars at ±0.25 mm. 59
Figure 2.14: Experimental results for deflection of three points at four fluid levels using water with vertical error bars at ±1 standard deviations and horizontal error bars at ±0.25 mm. 60
Figure 2.15: Stress–strain behavior for very low density and optimized polyethylene-based hyperelastic flap [79]. 62
Figure 2.16: Computational results for deflection of three points as a function of oil height and experimental results for deflection of three points at five fluid levels. ... 62
Figure 2.17: Image of experimental deflection and overlaid optimized computational deflection for oil at a fluid height (red) of 30 mm. 63
Figure 2.18: Computational results for the tuned material properties and the mean published values with experimental results. 63
Figure 2.19: Computational and experimental data for the deformed flap caused by water. 65
Figure 2.20: Image of experimental and overlaid computational deflection for water (red) at a height of 27.5 mm. 65

Figure 3.1: Diagram of the fluid and solid domains with boundary names. 71
Figure 3.2: Flow of information for the one-way FSI model to the fatigue model containing the strain-life analysis. 73
Figure 3.3: Three temperature profiles were scaled, so the maximum temperature at each time point reflected the maximum temperature as a function of time. 75
Figure 3.4: Maximum temperature along exterior of part as a function of time. 75
Figure 3.5: Transient material properties behavior for no cyclic material properties changing (top), cyclic hardening (middle), and cyclic softening [83]. 78
Figure 3.6: Stress–strain curve for cyclic loading until a stable hysteresis loop is reached, resulting in steady-state material properties [83]. 79
Figure 3.7: Strain-life curves for fatigue life prediction using no mean stress correction, Morrow, and Smith-Watson-Topper mean stress correction factors. 80
Figure 3.8: Design layout for the thermal cap geometry. 83
Figure 3.9: Comparison of material properties definitions for material hardening over four, 1.0 s* cycles. 84
Figure 3.10: Contour plot of plastic strain indicating strain is positive along the exterior of the part in agreement with experimental data. 85
Figure 3.11: The fatigue failure location (right) lies on the edge of the temperature hot spot (left) where the temperature gradient is the greatest, as indicated by the white lines. 87
Figure 3.12: Contour lines of temperature with overlaid tool markings, radial crack, and hoop crack. 89
Figure 3.13: Principal stresses along the exterior of the part show the maximum and middle principal stresses are on the same order of magnitude, indicating the potential for cracks to form both radially and in the hoop direction. 89

Figure 3.14: Damage produced from a single load amplitude of 0.001, 0.002, 0.0025, 0.003, and 0.0035 (m/m). With the known damage contribution of each amplitude, the damage of a single cycle can be calculated. With the damage per cycle known, the number of cycles until failure can be determined. 91

Figure 4.1: Layers inside the vessel wall to promote maximum thermal resistance between ambient conditions and the cool interior. 102
Figure 4.2: Cut cross section of the insulating foam interior showing the thermoplastic liner and material voids in the bulk of the insulating foam. 103
Figure 4.3: Diagram of experimental setup with location of temperature measurements and heat loss. 104
Figure 4.4: Image of feet on the ice block and perishable item containers. 107
Figure 4.5: Gel and ice locations for the validation study (left) and design improvement study (right). The X represents the locations for temperature measurements both experimentally and computationally in drawers 1, 3, and 5. 108
Figure 4.6: Specific heat spike to account for the latent heat of melting ice. 113
Figure 4.7: Comparison of computational and experimental results for an eight hour empty vessel test with various thermal conductivity values. 119
Figure 4.8: Results of computational model with various specific heat values for the gels compared to experimental results. 120
Figure 4.9: Experimental and computational data averages for drawer 3 with temperature bounds at ± 0.036 Ө. The top plot contains the experimental and computational boundary conditions labeled outside and the bottom plot contains a closer view of the experiment and computational results. 121
Figure 4.10: Computational results for maximum temperature after 60 hour test as a function of ice mass with uniformly and non-uniformly distributed ice. 123
Figure 4.11: Comparison of maximum temperature in each drawer with the simplified ice mass alteration model (left) and ice replacement model (right). 124
Figure 4.12: Comparison of ice location and maximum recorded temperature in drawer 1 under various configurations. Location of ice block is indicated by “ice” and temperature measurements indicated by *X*. 125

Figure 5.1: Model overview for the FSI model with named components in the peristaltic pump. 136
Figure 5.2: Diagram of the solid motion and resulting fluid flow directions. 136
Figure 5.3: Fluid tetrahedron mesh with temperature and flow boundary conditions. 139
Figure 5.4: Solid domain with QUAD8 elements along the surface of the cam and base with HEX20 elements through the volume of the tubing with boundary conditions defined. 141
Figure 5.5: Possible material properties paths for stress–strain material properties behavior between the unstressed and 100% strain measurement. 146
Figure 5.6: Diagram of the system coupling data transfer methods between the fluid and solid models. 148

Figure 5.7: Cross section of three-dimensional tetrahedron mesh at 0, 0.25, and 0.5 seconds. 153
Figure 5.8: The flow rate as a function of time over two pumping cycles. 154
Figure 5.9: Fluid velocity contour plot at 0, 0.125, 0.25, 0.375, and 0.5 seconds. 155
Figure 5.10: Static temperature contour in the fluid at 0, 0.125, 0.25, 0.375, and 0.5 seconds. 156
Figure 5.11: Maximum von Mises stress and maximum principal strain in the tubing as a function of time. 158
Figure 5.12: Maximum stress and strain in the constriction direction as a function of time. 158
Figure 5.13: Minimum stress and strain in the constriction direction as a function of time 159
Figure 5.14: Normal strain in the constriction direction at 0.25 seconds with maximum strain of 0.47 and minimum strain of -1.09. 159
Figure 5.15: Normal stress in the constriction direction as 0.25 seconds with maximum strain of 3.62 kPa and minimum strain of -35.7 kPa. 160
Figure 5.16: Contact status between tubing and base at 0.25 seconds. 161
Figure 5.17: Contact status between tubing and cam at 0.25 seconds. 161
Figure 5.18: Contact status for tubing self-contact at 0.25 seconds. Status in near because 0.5 mm gap has been specified to ensure fluid domain is present. 162
Figure 5.19: Contour of temperature at 0.25 seconds in the solid domain. 163
Figure 5.20: Contour of tubing life* for the base condition with shorter lifetime in red and longer lifetime in blue. 164

CHAPTER 1: INTRODUCTION

Fluid–structure interaction (FSI) modeling is a computational modeling technique in which multiple physical phenomena are modeled together to produce a single result that cannot be produced if each physical phenomenon was modeled individually [1–4]. These phenomena can include conjugate heat transfer, fluid mechanics, and solid mechanics. In order for an engineer to capture the full environment that a system or component will experience, FSI modeling may be required. Examples of systems that require FSI modeling are the motion of heart valves, the stress in a turbine blade due to kinetic loading and thermal expansion, the dynamic interactions of a diaphragm pump, the large deformation of a peristaltic pump, the interactions inside a breathing lung, the vibrations inside heat exchangers, and thin structures used for heat shields [1,2,5–12]. These examples require the Computational Fluid Dynamics (CFD) and Finite Element Analysis (FEA) models be linked together so that the results of each model impart forces on the other. Traditionally, the phenomena of fluid dynamics, heat transfer, solid mechanics, electromechanics, electromagnetics, vibrations, and chemistry have been evaluated separately [4,9,13,14]. However, with more powerful computers, commercially available software, and new techniques to add computational stabilities, engineers can now apply computational techniques to increasingly complex systems by linking multiple domains and analysis techniques together [1]. This increased complexity is evident both in the model size and in the ability to capture the full multiphysics environment.

1.1. FSI Modeling Methods

1.1.1. What is FSI Modeling?

FSI modeling is a subcategory of multiphysics modeling that involves a fluid domain and a solid domain [1,2,4]. Multiphysics modeling is a computational modeling method in which multiple physical phenomena are modeled together, with the results of one phenomenon directly affecting the outcome of another. With multiphysics modeling, the same results cannot be achieved if the phenomena are modeled independent of one another. These separate phenomena can include fluid dynamics, heat transfer, solid mechanics, electromechanics, electromagnetics, vibrations, acoustics, and chemistry [4,9,13,14]. FSI modeling is a subcategory of multiphysics modeling that comprises linking fluid dynamics and solid mechanics together to generate results that cannot be attained by modeling the domains independent of each other [3]. FSI modeling allows for displacement, force, pressure, and temperature data to pass back and forth between the fluid and solid domains. This allows for the linking of conjugate heat transfer, fluid mechanics, and solid mechanics together in a single system. Several methods are available for linking CFD and FEA models together to create an FSI model, including monolithic coupling, weak coupling, and strong coupling. Regardless of the method, when CFD and FEA models are coupled together, two challenges are introduced, as discussed below.

The first challenge that arises when coupling CFD and FEA models together is coupling the two independent mesh domains together while still accounting for the differences in mesh formulation and motion. This challenge arises from the fundamental differences between the Lagrangian mesh (utilized in FEA models), which deforms as a function of mass motions, and the Eulerian mesh (utilized in CFD models), which is fixed at all points in space and time [1]. A more detailed discussion of the fundamental differences and methods for accounting for these differences is found in the section entitled “FSI Modeling Methods.”

The second challenge is to transfer data between domains in a manner that mitigates instabilities, fluctuations, and non-physical phenomena at the domain interfaces. These instabilities arise from the mass effect, data transfer methods, and magnification of instabilities or shock waves at the interface. A more detailed discussion of these instabilities and mitigation methods is discussed in the “Data Transfer Methods,” “Mass Effect,” and “Computational Instabilities” sections.

1.1.2. System of Equations

The computational model containing the fluid domain utilizes the three-dimensional Navier–Stokes equations in conjunction with the continuity, volume fraction, and energy equations, Equations 1 through 4 respectively, utilizing a pressure-based solver for subsonic incompressible flow, along with the k-epsilon turbulence model. In the pressure-based solver, the momentum and continuity equations, Equations 1 and 2 respectively, are used in combination to calculate the pressure field. Additionally, because the model contains two fluids with a discrete interface, the volume fraction equation must be solved to conserve species, and the mass balance equation must be evaluated to conserve the overall mass of the system, Equations 2 and 3 respectively. The k-epsilon turbulence model utilizes Equations 5 and 6 to define k and epsilon respectively, and Equation 7 to define the turbulent viscosity.

$$\frac{\partial}{\partial t}(\rho\vec{v}) + \nabla\cdot(\rho\vec{v}\vec{v}) = -\nabla p + \nabla\cdot(\bar{\bar{\tau}}) + \rho\vec{g} + \vec{F} \tag{1}$$

$$\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\vec{v}) = S_m \tag{2}$$

$$\sum_{q=1}^{n}\alpha_q = 1 \tag{3}$$

$$\frac{\partial}{\partial t}(\rho E) + \nabla\cdot\left(\vec{v}(\rho E + p)\right) = \nabla\cdot\left(k_{eff}\cdot\nabla\mathrm{T} - \sum_j h_j\vec{J}_j + \bar{\tau}_{eff}\cdot\vec{v}\right) + S_h \tag{4}$$

Where t is time, ρ is density, $\vec{v}$ is the velocity vector, ∇ is the derivative in three-dimensional space, p is pressure, $\bar{\bar{\tau}}$ is the stress tensor, $\vec{g}$ is gravity, $\vec{F}$ is external body forces, S_m is a mass source term, α is the fluid volume fraction, E is the total fluid energy, k_{eff} is the effective thermal conductivity of the fluid, T is the temperature, h is the enthalpy, J is the diffusion flux, and S_h is a volumetric energy source

$$\frac{\partial}{\partial t}(\rho k) + \frac{\partial}{\partial x_i}(\rho k u_i) = \frac{\partial}{\partial x_i}\left(\frac{\partial k}{\partial x_j}\left(\mu + \frac{\mu_t}{\sigma_k}\right)\right) + G_k + G_b - \rho\varepsilon - Y_M + S_k \tag{5}$$

$$\frac{\partial}{\partial t}(\rho\epsilon) + \frac{\partial}{\partial x_i}(\rho\epsilon u_i) = \frac{\partial}{\partial x_i}\left(\frac{\partial\epsilon}{\partial x_j}\left(\mu + \frac{\mu_t}{\sigma_\epsilon}\right)\right) + C_{1\varepsilon}\frac{\epsilon}{k}(G_k + C_{3\varepsilon}G_b) - C_{2\varepsilon}\rho\frac{\epsilon^2}{k} + S_\epsilon \quad \textbf{(6)}$$

$$\mu_t = \rho C_\mu \frac{k^2}{\epsilon} \quad \textbf{(7)}$$

Where k is turbulent kinetic energy, u is velocity, μ is viscosity, μ_t is the turbulent viscosity, G is generation of turbulent kinetic energy, Y represents fluctuation due to compressibility, S is a user-defined source term, ε is the rate of dissipation, and $C_{1\varepsilon}$, $C_{2\varepsilon}$, C_μ, σ_k, and σ_ε are constants with all of the associated subscripts i, j, k, and t representing direction and time references. Each of these equations is defined for the fluid present in each control volume prescribed by the fluid mesh. If multiple fluid species or a volume of fluid model is evaluated, this set of equations will be evaluated for each fluid in the domain.

The computational model utilized the three-dimensional strain displacement, nodal displacement, and stress equations, Equations 8 through 10 respectively, to solve for the deformation, stress, strain, and forces across each node in the solid domain.

$$[\boldsymbol{B}] = [\boldsymbol{\partial}][\boldsymbol{N}] \quad \textbf{(8)}$$

$$\{\boldsymbol{\varepsilon}\} = [\boldsymbol{B}]\{\boldsymbol{D}\} \quad \textbf{(9)}$$

$$\{\boldsymbol{\sigma}\} = [\boldsymbol{E}]\{\boldsymbol{\epsilon}\} \quad \textbf{(10)}$$

Where B is the strain displacement, ∂ is the four-dimensional gradient (time and space), N is element shape function, ε is strain, D is nodal displacements, σ is stress, and E is modulus of elasticity.

1.1.3. Mesh Coupling Methods

When creating an FSI model, the first challenge is coupling the two independent mesh domains together while still accounting for the fundamental differences in the mesh formulation of each domain. FEA modeling utilizes a Lagrangian mesh where the finite element mesh is fixed to the mass and moves in space as a function of the mass motion [15]. In contrast, CFD modeling utilizes an Eulerian mesh where the finite element mesh is fixed in time and space with the mass passing through the mesh [16]. This means that the Lagrangian mesh is able to deform and move positions as a function of the fluid domain inputs; however, the solid domain displacement of the Lagrangian mesh cannot be directly applied to the fixed fluid domain.

FSI modeling has three primary methods for transferring the critical information between the fluid and solid domains: (1) the Lattice Boltzmann method, (2) the fictitious domain, and (3) the Arbitrary Lagrangian–Eulerian method. Each method is described in more detail below.

(1) The Lattice Boltzmann method utilizes a set of equations in which the fluid is represented as a discrete set of particles rather than the continuous flow represented by the Navier-Stokes equations [17–19]. Compared to the numerical solution of the Navier-Stokes equations, the Lattice Boltzmann method requires less computational time, but is limited in its ability to model both fluid flow and conjugate heat transfer in

conjunction with compressible flow [19]. Thus, this method may not be sufficient for modeling complex fluid dynamics problems.

(2) The fictitious domain method does not model the exact interface between the fluid and solid regions, but keeps the fluid domain (Eulerian mesh) fixed at all times and only allows the solid domain (Lagrangian mesh) to deform [5,20,21]. At each time step the location of the solid domain is interpreted into the fluid domain by prescribing a zero velocity value at the elements most closely linked to the surface of the solid [20]. It is best to imagine these two models as completely independent of one another, but layered on top of each other to achieve the interaction, Figure 1.1. The fictitious domain method is useful because the Lagrangian mesh of the solid is free to deform, yet it does not require alteration or remeshing of the Eulerian mesh of the fluid [20,21]. One of the disadvantages of the fictitious domain method is the instabilities that arise as the interface between the two models moves over time [22,23]. One way to minimize this instability is to use the fictitious domain method with adaptive meshing of the fluid domain, which allows the fluid domain to more accurately define the edge of the solid domain [22,23]. Theoretically, the fictitious domain method can be used to model flexible thin flaps; however, the vast majority of research groups using this method apply it to model rigid bileaflet heart valves that do not experience measurable deflection [6,23–25]. Research groups using the fictitious domain method have claimed to produced transient FSI models, but have not done so with a single transient model that operates through the entire motion of the flap. Instead, these groups have evaluated a handful of fixed flap angles—15, 30, 45, and 60 degrees—under steady-state conditions [25–27].

(3) The Arbitrary Lagrangian–Eulerian method utilizes a Lagrangian mesh to represent the solid domain and an Eulerian mesh to represent the fluid domain, while allowing for a seamless interface over which data can be transferred, Figure 1.1 [2,28,29]. Both non-conformal and conformal mesh interfaces can be utilized with the Arbitrary Lagrangian–Eulerian method, Figure 1.2. The challenge with using the Arbitrary Lagrangian–Eulerian method is maintaining the mesh quality of the Eulerian mesh (fluid domain) as it deforms as a function of the Lagrangian mesh (solid domain). The mesh motion in the traditionally fixed Eulerian mesh is implementation through dynamic meshing. If large deformations are present, automated remeshing steps must be taken in the Eulerian mesh to maintain sufficient element quality. If the element quality is not maintained, instabilities can be generated causing unphysical pressure, temperature or displacement gradient, thus leading to computational failure [2,30,31]. One of the benefits of using the Arbitrary Lagrangian–Eulerian method is that the full toolbox of CFD and FEA methods is available. The downside to Arbitrary Lagrangian–Eulerian modeling is the possible introduction of instabilities at this mesh interface, the possibility of low element quality, and increased computational time due to dynamic remeshing and automated remeshing [1,2,30,31]. The Arbitrary Lagrangian–Eulerian model also allows for both monolithic and partitioned data transfer methods as discussed in the section titled "Data Transfer Methods" [1,2,30,31].

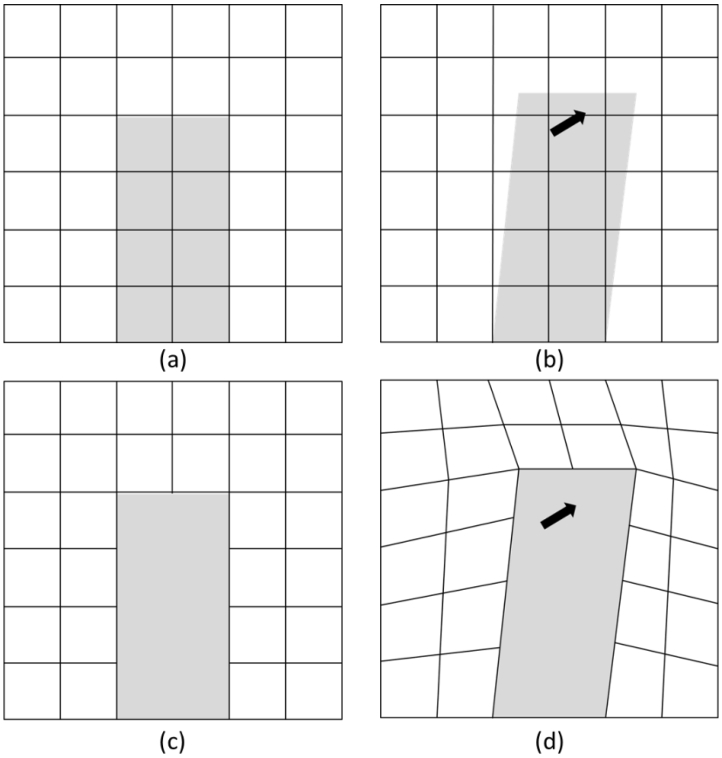


Figure 1.1: Black lines represent the fluid mesh (Eulerian) and gray body represents the solid mesh (Lagrangian). Mesh motions with the fictitious domain method are shown from a to b, and mesh motions with the Arbitrary Lagrangian–Eulerian method is shown from c to d.

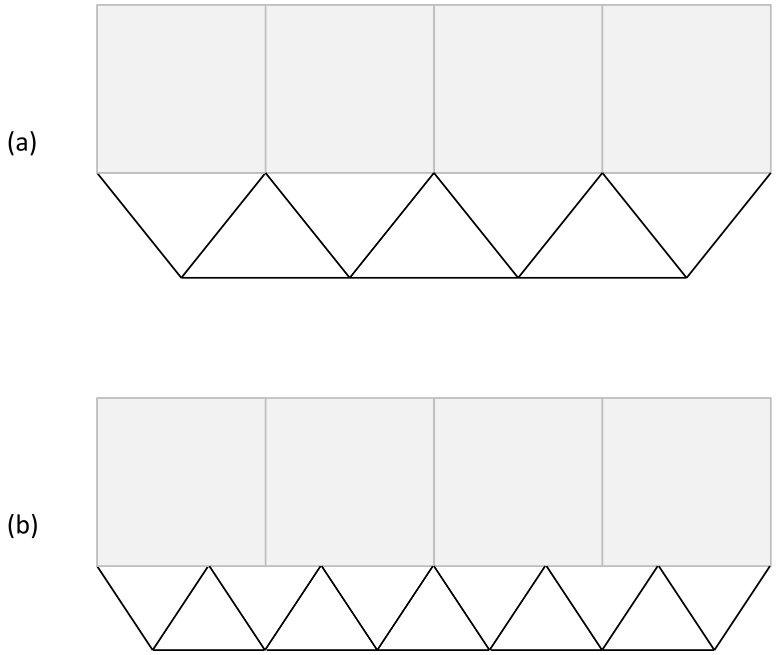


Figure 1.2: Conformal meshing interface (a) and non-conformal meshing interface (b).

1.1.4. Data Transfer Methods

Data transfer between the discrete domains requires defining the frequency and direction shared information is passed. The frequency by which information is passed is defined by the coupling type: monolithic coupling, strong coupling, and weak coupling [2,30]. While the type and direction of the information passed can be one-way or two-way.

Monolithic coupling involves solving both the fluid and solid system of equations simultaneously as a single system (matrix) of equations, Figure 1.3. Generally, monolithic data transfer utilizes custom computational codes and requires extremely large and powerful computers compared to other coupling methods [9]. Additionally, these custom computational codes are limited to the specific multiphysics phenomena of each unique problem and require a significant amount of code customization for each problem [32,33]. Some specific areas where customization is often required include wave motion, vibrations, and heat transfer [1,34–36]. Because these custom codes do not allow use of the full CFD and FEA features available in commercial codes and require large computational resources, the monolithic coupling method is not utilized by most companies.

In contrast, both strong and weak coupling—referred to as "partitioned approaches"—pass data between the fluid and solid models in an attempt to solve the two systems of equations separately, but with shared boundary conditions [30,37,38]. Both strong and weak coupling approaches are available in commercially available software like ANSYS Multiphysics, COMSOL Multiphysics®, STAR-CCM+, and MpCCI Co-simulation [3,4,14,39]. The difference between strong and weak coupling methods is when and how often data is passed from one model to another with respect to each time step. In strong coupling, each domain is evaluated once and then the data is exchanged between the models, Figure 1.4. Then the same time step is reevaluated using the results from the other domain as updated boundary conditions. This process of exchanging data between the domains is repeated until a converged solution is reached in both domains, then the next time step is taken and the data exchange process repeats. In weak coupling, data is exchanged a maximum of one time between domains before the next time step is taken thus no check is performed to ensure a converged data transfer has been reached, Figure 1.5. In addition to not checking for a converged data transfer, the data may be transferred less frequently, which leads to very weak coupling and eventually one-way coupling.

Each coupling method has its advantages and disadvantages; thus, each model presents a unique challenge. Traditionally, weak coupling produces results in a faster timeframe because each domain is only evaluated once per time step, but the model tends to be more unstable and converges to less accurate answers if deformation is large or the deformation occurs rapidly [2,30]. The strong coupling method can resolve some of these issues, but takes more computational time than weak coupling and cannot fully eliminate all of the instabilities from the mass effect. A monolithic method allows the evaluation of very unstable models, but at the expense of significantly increased computational time relative to a partitioned approach [30].

Each of these methods can have one-way or two-way data transfer between each model. The types of data that can be transferred include temperature, heat transfer, pressure, force, and displacement, among others. One-way coupling only passes data in one direction, meaning the model either passes data from the fluid to the solid or from the solid to the fluid. One-way coupling is a useful tool to reduce the computational time required to evaluate a model [2,4]. However, this coupling can only be utilized when the results of one model will have insignificant effects on the other model. An example of when one-way coupling can be used is a skyscraper under a wind load. The wind load creates a unique pressure profile on the building, causing it to deform; however, the small deformation has negligible effects on the pressure profile around the building, resulting in the same wind load on the building after the deformation is applied. Two-way coupling is required when data is passed in both directions between the fluid and solid models [2,4]. This type of coupling is necessary for models with large deformations, like a heart valve or extreme thermal expansion problems, because the results of one model will significantly alter the boundary conditions and outcome of the other. If large displacements are present in addition to heat transfer between models, multiple different data transfer types can be used to transfer displacement, pressure or force, temperature, and heat transfer.

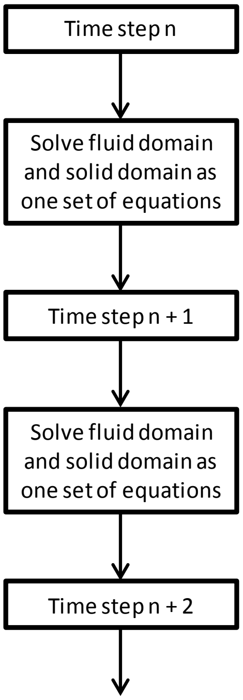


Figure 1.3: Two-way monolithic time stepping for FSI modeling.

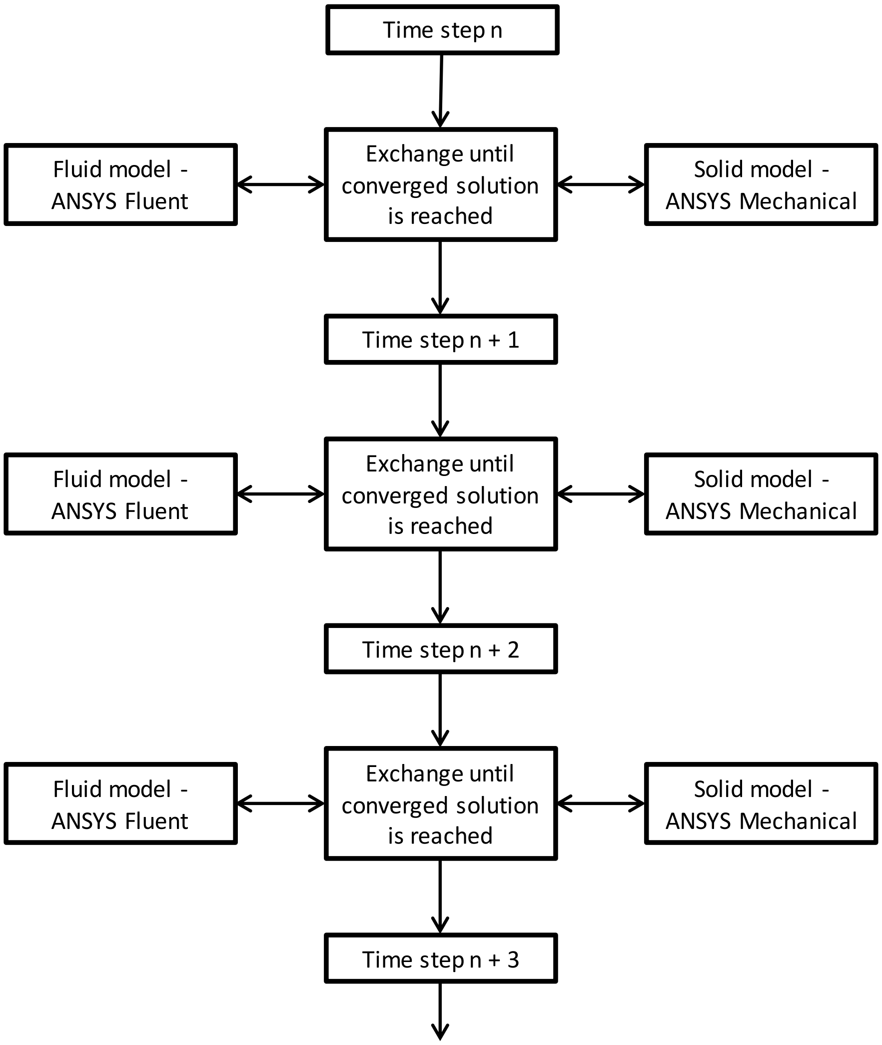


Figure 1.4: Two-way strong coupling algorithm used to transfer data back and forth between fluid and solid models.

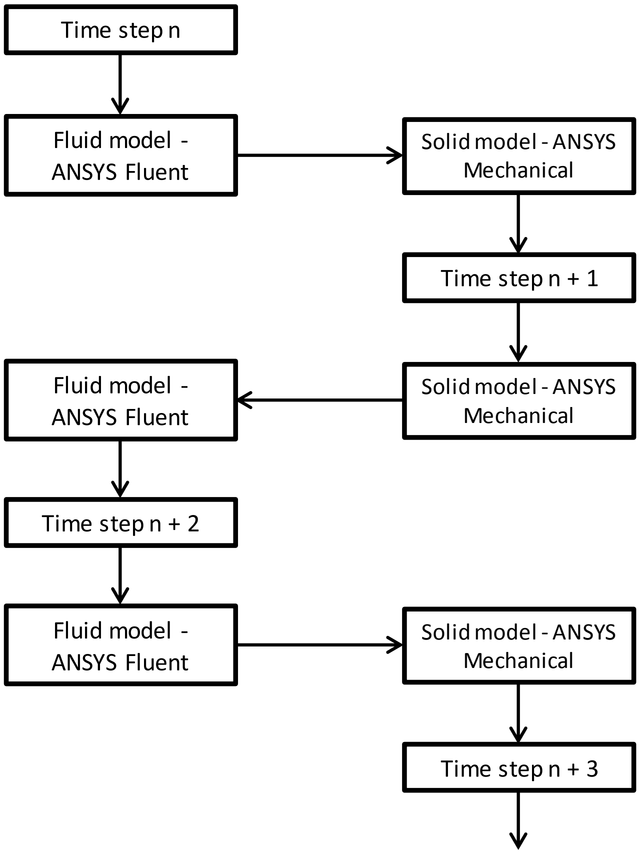


Figure 1.5: Two-way weak coupling algorithm used to transfer data back and forth between fluid and solid models.

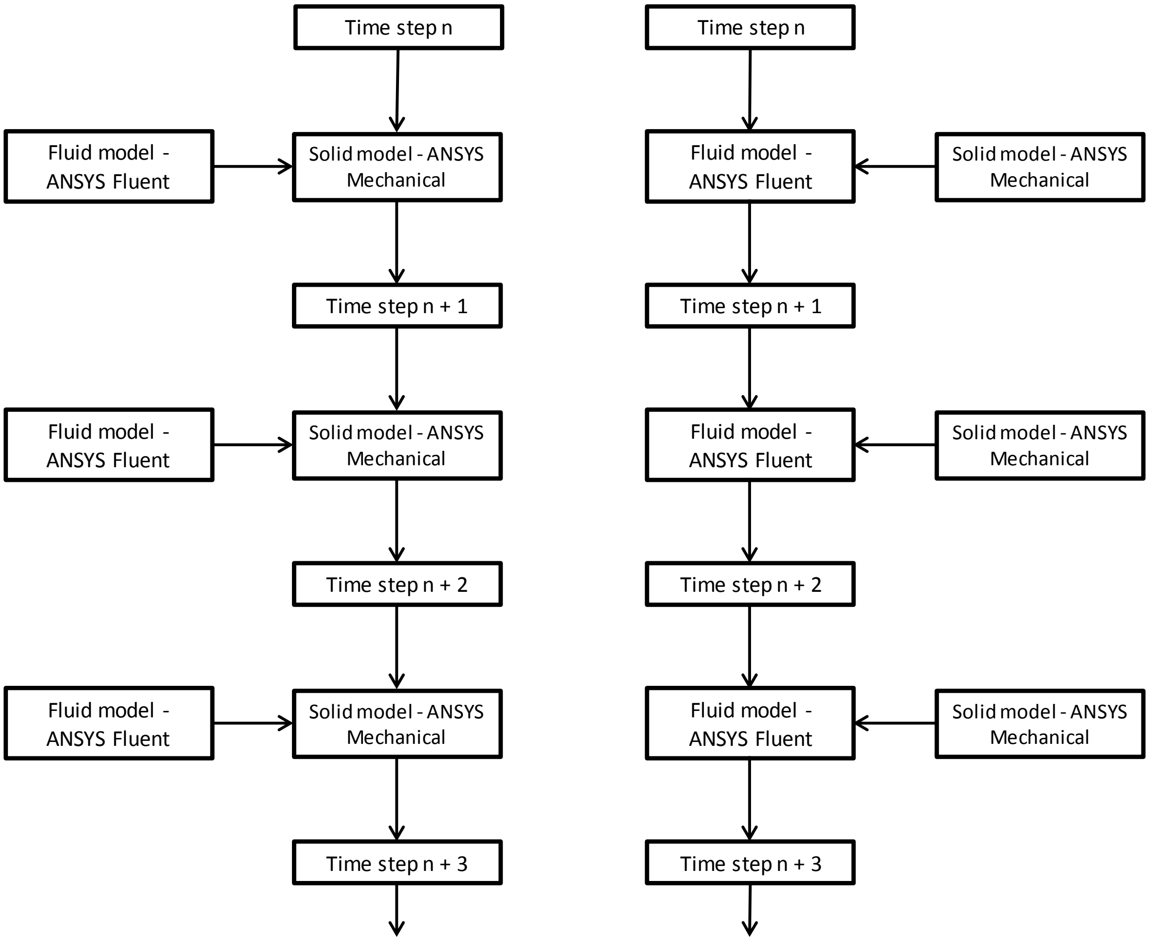


Figure 1.6: One-way data transfer for fluid to solid only (left) and solid to fluid only (right).

1.1.5. Mass Effect

In many cases, FSI models contain large amounts of deformation and/or thin-walled solid components, which tends to result in instabilities at the fluid and solid interface. These instabilities are primarily caused by the mass effect, which occur when a stiff body is interfaced with an incompressible fluid and movement is present in the solid, resulting in fluid compression or expansion, Figure 1.7 [2,4,31,37]. As the solid model moves, so does the interface and the attached fluid mesh, which results in a change in volume of the fluid domain. If the fluid is treated as incompressible, i.e. standard water or oil, and the fluid volume changes without a change in mass, an instantaneous change in pressure and density will occur, creating a shock. When the discontinuous change is translated back to the solid, it creates large artificial pressure gradients. This results in singularities at the interface that eventually lead to oscillations and computational divergence. This shock is a computational artifact, and by adding a small amount of compressibility to fluids like oil and water, some mass effects can be controlled and mitigated.

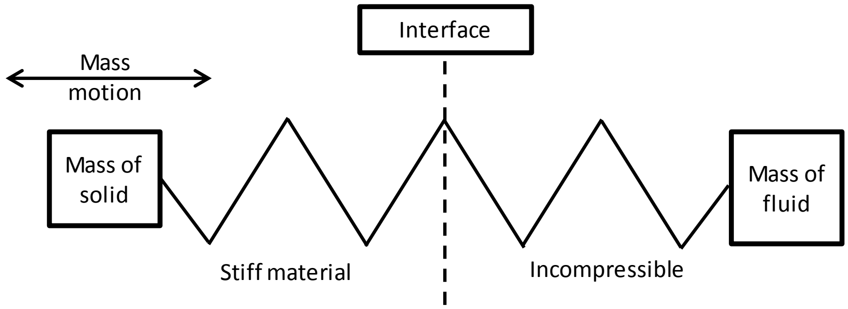


Figure 1.7: The mass effect experienced at the interface between a stiff solid model and an incompressible fluid model.

The susceptibility of an FSI model to the mass effect can be estimated through a stability equation, Equation 11 [2]. In this equation, ρ_s and ρ_f are the densities of the solid and fluid respectively, h_s is the thickness of the solid, R is the radius of the fluid passageway, and L is the length of the fluid passageway. However, this equation has limitations because it does not take fluid flow behavior or material stiffness into account. It only assesses the stability of the geometry and thus does not address any characterization for the fluid flow or solid deflection. By utilizing fluid with higher viscosity, solid with stiffer materials, low fluid flow rates, and stabilizing techniques, stability can be introduced to the model to reduce the magnitude of the shock and increase the damping of any shocks that do appear. If instabilities do persist, artificial damping, fine-tuned relaxation factors, and load ramping of data transfer can be applied to manipulate the coupled information between the interfaces to add stability.

$$\frac{\rho_s h_s \pi^2 R}{\rho_f 2L^2} > 1 \tag{11}$$

1.2. Why FSI Modeling is Difficult

All FSI models introduce challenges that can result in an inaccurate solution and computational divergence. In particular, repeatedly using the output of a computational model as the input to another computational model can result in compounding errors from repeatedly using the same slightly incorrect values. Consequently, the more times data is passed, the larger the compounding error. FSI models also experience convergence issues caused by the mass effect, residual convergence between each domain, possible ramping of data between each interface, methods by which data is passed between interfaces, frequency by which data is passed between interfaces, and magnification of any instabilities at the interface. Furthermore, FSI modeling also experiences the same instabilities and convergence challenges as individual CFD and FEA models.

1.2.1. FSI Modeling History

Publications containing application-based multiphysics models began to emerge in 1999 and continued until 2003 [5,25,29,40–42]. During this time, the results from multiphysics models and FSI models were thought to accurately represent the desired physics system. However, around 2003, the understanding of mass effect due to mesh motion in an Eulerian mesh sparked a split between multiphysics and FSI modeling, causing deep questioning about the accuracy of previously conducted FSI models. Multiphysics modeling does not suffer from the same instabilities as FSI modeling because an Eulerian mesh is not used; thus, multiphysics modeling did not suffer a setback and continued to develop into a robust design tool. Meanwhile, between 2003 and 2008 FSI modeling experienced a period where publications shifted from application-based to investigation aimed to better understanding of the mass effect [22,23,34,38,43,44]. Beginning in 2008, sufficient understanding of the mass effect in FSI modeling existed such that with monolithic coupling could be performed in a stable manner [32,33]. However, due to the previously discussed disadvantages of monolithic coupling, it was still not a suitable tool for industry application. It was not until around 2012 that stability techniques were able to control the mass effect enough to allow a return to application-based modeling [2,45–47]. Despite this, a validation gap still remains between the computational results from FSI models and quantitative experimental data. A more detailed discussion about FSI validation follows in the section titled “Validated FSI Models.”

1.2.2. Thin-Walled Bodies

Thin-walled FSI models experience all the same challenges as thick-walled FSI models, but have several uniquely challenging issues due to the geometric setup of thin-walled problems that can magnify existing instabilities. Generally, thin-walled FSI problems experience larger deflection relative to the wall thickness caused by small forces and oftentimes utilize non-metal materials that do not exhibit the behaviors of linear material properties. In the solid domain, these small forces can cause large deflection and any instabilities, even if small, can quickly escalate to computational divergence. If large deflections are present in the solid, the Lagrangian mesh of the fluid domain will require remeshing to maintain sufficient element quality. If the element quality is not maintained at any point, small instabilities can escalate to computational divergence [15].

1.2.3. Computational Instabilities and Benchmarking

As with all computational models, validation is paramount to ensure the model accurately represents the physical system. Because FSI modeling couples two computational models together with an interface that can allow damping and relaxation factors, validation may be more important in FSI models than in single domain computational models. Additionally, because computational data is passed from one model to another, even a small error can compound into a larger error as time progresses. Furthermore, the gathering of quantitative experimental data for the validation of an FSI model can be challenging because of the oscillations and unsteady nature of the system. To account for the lack of qualitative data, the FSI industry has adopted the term "benchmarking," which means a qualitative agreement of trends and *not* a quantitative comparison to experimental data. Several publications are available that benchmark FSI modeling, but these publications lack the quantitative experimental data to directly indicate validation of a computational model to physical results [44,32,33]. These theory-based benchmarks are sufficient for demonstrating competency in modeling techniques; however, they lack relevance in design applications because simplified geometries, linear material properties, constant temperature, and two-dimensional assumptions cannot always be applied. Therefore, extreme care should be taken to validate and document each step of an FSI model.

1.2.4. Validated FSI Models

As of the date of this dissertation, a fully encompassing dataset is not available in literature that directly compares quantitative experimental data to computational results. In the last five years, some validation work has been performed, but limitations exist due to modeling assumptions and voids in the results. In a study completed by Tian et al., six FSI validations were performed; however, the applicability of these validations are limited because they do not contain sufficient experimental data for a quantitative validation [48]. Of the six validated cases, only one contained experimental data, but that data only contained results from a single time point, making a true transient validation difficult to assess. Three of the cases contain only a fluid or solid model, not both. Finally, two of the cases contain FSI modeling and compared results to previously published articles used for benchmarking FSI models, but these benchmarking models do not contain results measured from an actual system and contain unphysical flow conditions and fluid properties [33,37,38,44]. As a whole, these cases show qualitative validation is possible, but lack the quantitative data that would enable acceptance of the techniques to generate optimized designs.

1.2.5. Material Properties

In addition to the instability sources discussed in the “Data Transfer Methods” and “Mass Effect” sections, accurate solid material properties are vital to an accurate and stable FSI model. A significant area of interest in FSI modeling focuses on nonlinear material properties and when large deflections are present, i.e. when deflection is greater than the thickness of the material. Furthermore, these nonlinear material properties can have directional-dependent material properties with highly elastic characteristics, resulting in deformation at relatively low forces, i.e. heart and artery tissue [49]. If the measured material properties, orientation of the material properties or applied force is inaccurate, the resulting deflection of the deformable material can be inaccurate and unstable.

1.2.6. Solid Contact in FSI Models

Accurately modeling the surface contact within the solid domain of an FSI model is critical if flow passages are being constricted with eventual stoppage of the fluid flow. If friction in present at the contact regions, heat generation can result, leading to temperature gradient in both the fluid and solid domains. If a fluid channel is severely restricted and contact between solid surfaces occurs, resulting in flow stoppage, extra care must be taken to ensure the fluid mesh quality is maintained and does not reach a point of singularity. Implementing the proper dynamic mesh setting and contact definitions can ensure an appropriate mesh is maintained. If heat generation is present at the contacting surfaces due to friction, the thermal energy must be maintained across the fluid and solid interface by accounting for any material properties that might fluctuate as a function of temperature.

1.3. Advancements in Computational Modeling

1.3.1. CFD Modeling

CFD modeling was originally born out of the need for fluid dynamicists to understand experimental results [16]. The techniques and understanding gained from these early models laid the foundation for today's CFD industry. Currently, CFD modeling is used in a wide range of industries—aerospace, pharmaceutical manufacturing, medical devices, mining, petroleum, automotive, and manufacturing, among others—and for countless applications [2,4,50–52]. The CFD toolbox is currently able to capture most fluid dynamics phenomena, including high-speed compressible flow, phase change, cavitation, evaporation, chemistry, conjugate heat transfer, and multi-phase flow with and without discrete interfaces [3,4,39]. Furthermore, these complex fluid systems are capable of being linked to probabilistic and optimization software, allowing for automated design space exploration and optimization. CFD optimizations are readily performed in industries such as aerospace and space systems design to evaluate optimal aerodynamic performance, turbine mixing and output efficiencies, and heat management in spacecraft, satellites, and heat exchangers [50–56]. However, all of these design explorations are limited to steady-state or beginning of life applications as opposed to observing the full life span of the system right up to failure [7,8]. To date, few CFD studies have explored the design space at the end of life or under non-optimal conditions where fouling, fatigue, and wear may affect the efficiency of the design. A major contributor to not exploring this design space is the difficulty of acquiring accurate end-of-life boundary conditions and geometric configurations. This data can be gathered either through detailed measurements of the real system near end of life or by evaluating a transient model that incorporates fouling, fatigue, wear, and other aging factors to dynamically account for changes over the life of the system. These transient analyses require significantly more computation time compared to steady-state models, leading many to avoid these types of analyses.

1.3.2. CFD Limitations

The fundamentals of CFD modeling utilize the techniques of a control volume and a Lagrangian mesh—conservation of momentum, mass, and energy—where the mass passed through a mesh fixed in space. Previous work has demonstrated that well-characterized motion of a Lagrangian mesh can be modeled in cases such as rotating turbine blades or piston cylinder motion [1,4,9,15,16,57,58]. Due to this limitation, traditional CFD modeling alone cannot model cases such as heart valves, diaphragm pumps, peristaltic pumps or the expansion of a lung because the Lagrangian mesh would need to change dynamically as a function of the forces imparted between the fluid and solid domains [2,6,10–12,24,26,29,42].

1.3.3. FSI Modeling

FSI modeling came about from the need to understand how a fluid domain reacts as a function of the solid domain and how a solid domain reacts as a function of the fluid domain. Key cases demonstrating the interdependent results include models of heart valves, diaphragm pumps, the flapping of a flag, and the expansion of a lung where large deformations are present [1,2,26,44,33,59,60]. These inaugural FSI studies were evaluated between 1999 and 2003 by initially utilizing stiff materials, such as metals, and later transitioning into more flexible materials, such as rubbers, plastics, and fabrics [26,27,59]. These original models were also evaluated at fixed steady-state time points instead of transiently, i.e. at a valve opening of 0, 30, 60, and 90 degrees rather than at all angles, with the valve opening as a function of time [11,27]. These original studies contained little solid material deformation, which resulted in quasi-stable models. Although FSI modeling at this time was capable of modeling small deflections, the major area of interest in the FSI community was in more flexible materials experiencing large deflections, resulting in ever-increasing computational instabilities [20]. These instabilities originate from the data transfer methods and mass effects due to changes in control volumes [2,31,37,38]. Due to these instabilities and the lack of stability techniques, the FSI modeling industry experienced a stagnant period from 2003-2008. Within the last 10 years, additional research created a greater understanding of the mechanisms driving the numerical instabilities, enabling the development of methods and techniques to mitigate these issues relating to data transfer and the mass effect [2,30,31,37,38,32]. Currently, commercially available software codes—ANSYS Multiphysics, COMSOL Multiphysics®, STAR-CCM+, and MpCCI Co-simulation—are available that allow for multiphysics modeling in addition to coupling multiphysics modeling with optimization and probabilistic techniques [3,4,14,39]. Despite the availability of viable commercially available software codes, a literature review found that while there is sufficient work pertaining to the numerical methods for coupling fluid and solid systems, there is minimal work demonstrating the successful implementation of a fully encompassing two-way FSI model with validated results [4]. Several well-established models have claimed to optimize FSI modeling, but have done so using one-way data transfer using steady-state conditions [7,8,58]. A more detailed discussion relating to the limitation of one-way and two-way coupling is provided in the section titled "Data Transfer Methods." Other studies have demonstrated optimal designs, but have done so with a parametric study involving less than a dozen designs [61]. Although commercial codes are available and used by companies, as previously mentioned, minimal work has been produced demonstrating the successful implementation of a fully encompassing two-way FSI model with validation [7,8]. Of these few successful documented models, none contains a direct comparison between modeling results and experimental data [25,42,60]. Several publications are available that contain a theoretical dataset with matching FSI results, but these datasets are theoretical only and contain fluid flow assumptions that are physically unreasonable and could not be reproduced experimentally [30,38,44,32,33,48]. Thus, these datasets and modeling results still leave a gap between direct comparisons of computational results to experimental data.

FSI modeling has shown increasing success in the ability to evaluate previously difficult geometries, such as models containing contact, heat valves, and parachutes [45–47,62–72]. Each of these scenarios remains difficult to simulate due to the thin-walled geometries, highly elastic material properties, and highly turbulent fluid flows. A significant portion of the FSI work published between 2010 and 2015 comes from Yuri Bazilevs and Kenji Taskizawa; their work includes heat valves, cerebral aneurysms, parachutes, wind turbines, and new modeling techniques for implementing contact in FSI models [46,47,62,64–66,68–73]. Models for cerebral aneurysms and heart valves have improved, but many of the simulations do not contain contact. For further discussion of these limitations, refer to the section “Solid Contact in FSI Models” for an outline of the difficulties of modeling contact and FSI models [46,47,62–64,73]. Therefore, these models do not carry the process through to completion and leave a gap for improvement. The models that do contain contact are evaluated using a monolithic approach that is computationally intensive and not sustainable for industry application [45,62]. A more detailed discussion about the monolithic approach is covered in the section titled “Data Transfer Methods.”

Similarly, the stable modeling of parachutes has also been performed, but requires assumptions that simplify the fluid flow field and movement of the fabric. To attain computational stability, parachutes are treated as a porous membrane with air passing through it, whereas the true process has air passing through specifically designed openings in the parachute, not through the fabric itself [67–70,72]. Additionally, only the fully deployed stable motion of the parachute is being modeled and not the opening and deployment [67–70,72]. The modeling of large scale three-dimensional wind turbines with large deflections has been demonstrated on a 60 meter diameter blade [65,66]. However, this is not a thin-walled structure and the large deformation is not significant when compared to the large blade diameter and resulting computational mesh element size [65,66]. In 2014, additional methods for tracking contact in FSI models were developed to allow easier application of contact relative to the methods used in the Arbitrary Lagrangian mesh, but these improvements have yet to be implemented in commercially available codes or generate published results on an application basis [45,71].

1.3.4. Modeling System Life

A validated computational model of any type—CFD, FEA, FSI, and multiphysics—can be used to understand the operation of a system and further used as a tool to improve the performance of the system. The performance of systems can be determined by efficiency, aerodynamic performance, power output, heat management, strength, weight, fatigue life, and time before system failure. The failure point is highly dependent on the application and desired performance of the system. Previous optimization work has been conducted using CFD and FEA modeling to determine and improve these failure points. Modeling system life using FSI modeling has been performed, but only using weak-coupled methods on a steady-state basis [7,8]. To the author's knowledge no lifetime modeling has been performed on a transient model using a strong-coupling method.

1.4. Dissertation Overview

1.4.1. Motivation

Commercial software packages are currently available that allow engineers to produce FSI models using strong- and weak-coupling techniques. Many publications are available that utilize these packages on an application basis to evaluate the design of existing systems. However, to the author’s knowledge, a complete quantitative validation of these commercially available software packages does not exist. This work sets out to provide a complete quantitative validation and then test the performance accuracy of the models by evaluating design alternatives both computationally and experimentally.

FSI modeling is a tool that could be useful to engineers in countless industries. However, to date, FSI modeling has not demonstrated the ability to meet industry standards for evaluation time, cost, ease of use, and reliability. Before FSI modeling can be used in industry applications, it must demonstrate it can produce accurate results through means of a quantitatively validated study (Study 1 - Chapter 2). Three areas of particular interest for FSI modeling include cyclic thermal cycles, thermal management of a closed system, and the operational performance of a peristaltic pump as it ages, each of which was investigated for this dissertation. The application of cyclic thermal cycles is applicable for representing the cycles of turbine blades, burners, engines, ovens, and furnaces (Study 2 - Chapter 3). The use of thermal management of a closed system is appropriate for modeling spacecraft, electronics, medical organ and therapeutic protein transport, and food transportation and storage (Study 3 - Chapter 4). The ability to know the operational performance of a peristaltic pump as it ages is needed in industries such as pharmaceutical manufacturing, medical devices, and mining (Study 4 - Chapter 5).

Previous work has been conducted in several of these areas using monolithic coupling methods that utilized custom scripting and supercomputers. For FSI modeling to become an appropriate design tool for industry application, it must have the ability to be performed using commercially available software packages and be evaluated on computationally inexpensive equipment. Accordingly, the FSI models evaluated in this dissertation utilized only these such tools.

1.4.2. Research Question

The objective of this research is to apply FSI modeling on computational inexpensive equipment using commercially available software in such a way to demonstrate its effective use as an industry design tool. To do this, the cases under study needed to be sufficiently complex to exploit the inherent instabilities of FSI methodology, i.e. three dimensional, nonlinear materials, thin walls, steep gradients in both time and space, and transient behavior. Accordingly, this work purposely pushes the boundaries of the current capabilities of FSI modeling with case studies designed to incorporate these instabilities and therefore demonstrate that FSI modeling is capable of solving models involving complex instabilities. The work also sought scenarios that predicted system failure and optimal design to extend service lifetime. Thereby, extending the literature in the FSI area on these previously overlooked application and providing implementation strategies for successful simulations. Four independent studies were performed, evaluating three separate modes of failure in FSI models, Table 1.1.

Study 1: The first study provides a validation for FSI modeling techniques by comparing the results of a thin-walled FSI geometry under hydrostatic forces with experimental data. To the author’s knowledge, this study provides the first robust dataset allowing for direct comparison of a fundamental yet all-encompassing three-dimensional experiment and computational model with nonlinear material properties and large material deflection.

Study 2: The second study conducts a parametric study that evaluates the factors influencing an FSI model containing a highly complex thermal-fluid fatigue model. This model involves dynamically changing temperature loads resulting in significant thermal expansion that led to material yielding and dynamic fatigue life. This model laid the foundation for the processes used in the subsequent studies for performing fatigue analysis within FSI models.

Study 3: The third study looks at a multiphysics conjugate heat transfer problem. The model was tuned, validated, and optimized for lifetime. The validation of the thermal-fluid system was performed using readily available experimental data. The study demonstrates the use of phase change behavior and pushes the limits of possible transient evaluations from the order of second and minutes to days.

Study 4: The final study evaluates the highly complex fluid and solid phenomena involved in a peristaltic pump FSI model where the desire is to determine the factors that influence the lifetime and failure methods for the tubing in the pump. The model in this combines all the methods and techniques from the three earlier studies and applied them to a thin-walled tube geometry with nonlinear and temperature-dependent material properties to create large solid deformation and fluid motion.

All studies utilized ANSYS Multiphysics for the setup and evaluated the CFD, FEA, and FSI models on inexpensive desktop workstations—HP xw8600 workstations with Intel Xeon CPU’s operating at 2.66 GHz and valued at about $5000. Matlab was used to manage the optimization and design of experiment for the computational models. This research shows that FSI modeling and lifetime design can be implemented using commercially available software evaluated on relatively inexpensive computational resources. This demonstrates that all sizes of design groups and companies can use FSI modeling in a cost effective manner.

Table 1.1: List of studies with details about key coupling features, available experimental data, and methods of validation.

	Problem Physics	Industry Goal	Novel Contributions	Objective
Study 1	Hydrostatic forces deform hyperelastic dam	Provide the methods and qualitative validation of an FSI model	Provide a qualitative validation of an FSI model	Qualitative validation of computational FSI and experimental results
Study 2	Thermal expansion due to thermal cycles	Extend the lifetime of our industry collaborator's currently designed part	Couple a transient FSI model with fatigue life analysis	Demonstrate FSI modeling can be used to perform fatigue life analysis
Study 3	Thermal management using phase change materials	Extend the lifetime of our industry collaborator's currently designed part	Estimate the lifetime of a thermal-fluid system	Demonstrate FSI modeling can be used to preform thermal-fluid lifetime analysis
Study 4	Fluid and thermal flow as a function of solid pumping motion	Identify parameters that improved the lifetime of peristaltic pump tubing	Model an industry-applicable model using industry-appropriate methods	Demonstrate lifetime prediction in 3-D, thin-walled, two-way FSI model

CHAPTER 2: VALIDATION OF A THIN-WALLED FLUID–STRUCTURE INTERACTION MODEL WITH EXPERIMENTAL RESULTS

2.1. Abstract

Fluid–structure interaction modeling has become more available due to the increased computational power of modern computers and stability of algorithms employed. However, limited literature currently exists for validation of a thin-walled geometry simulation to experimental data. This work measures, computes, and captures the deflection of a three-dimensional hyperelastic flap as it resists the hydrostatic pressure of a fluid column. Deflection results from experimental and computational analysis were directly compared. The computational model was tuned to a single operating condition through an automated optimization that adjusts the solid material properties to minimize the squared difference between the computational model and the experimental results. To illustrate the completeness of the tuned material properties gained from the optimization, a secondary computational model and experiments were evaluated with a secondary fluid. The results of the primary and secondary models in conjunction with the experimental results indicated a thin-walled fluid–structure geometry can be modeled to accurately predict the defection behavior nonlinear.

2.2. Introduction

The field of fluid–structure interaction (FSI) modeling is the study of how the fluid domain and solid domain interact to create a coupled system with results that cannot be achieved by Computational Fluid Dynamics (CFD) or Finite Element Analysis (FEA) modeling independently. Today, commercially available codes exist that can model these coupled systems without custom scripting [3,4,14,39].

In many cases, FSI models contain a large amount of deformation and/or thin-walled solid components, which tends to result in computational instabilities at the fluid and solid interface. These instabilities are primarily caused by the mass effect that results from the expansion of an incompressible fluid and the presence of a solid mass at the interface [31]. Thin-walled geometries magnify this instability because the amount of compression and expansion per unit volume is more significant compared to thick-walled geometries. Although commercial codes are available today for FSI modeling, to the author's knowledge, a quantitative validation of an FSI model with experimental data has yet to be conducted.

2.3. Problem Description

The problem evaluated for this study is a trapezoid-shaped polyethylene-based rubber elastic flap acting as a flexible dam. Figure 2.1 shows the schematic of the test fixture with the inlet, outlet, and elastic flap labeled. As fluid builds up behind the flap, a hydrostatic pressure sufficient to deflect the flap is generated, and the flap deflects as a function of the fluid height. The deflection at three points along the height of the flap was measured as a function of fluid height. The flap is a trapezoid shape with a height of 9.5

cm, base width of 7 cm, and top width of 4.5 cm, resulting in a side wedge angle of 75 degrees, as depicted in Figure 2.2. This trapezoid shape allows a true three-dimensional model to be evaluated while still leaving a perpendicular surface to enable easy and undistorted imaging of the deflection through the course of the experiment. The flap is made of generic polyethylene-based rubber 1/16 inch thick that has been pre-fatigued to ensure material hardening and fatigue did not factor into the results. Initially, generic polyethylene was used as a base material in the computational model until the optimization evaluated the exact material properties [4]. Oil was utilized for the material properties optimization of the polyethylene. For the secondary test, the tuned material properties for the flap were utilized while water provided the hydrostatic pressure.

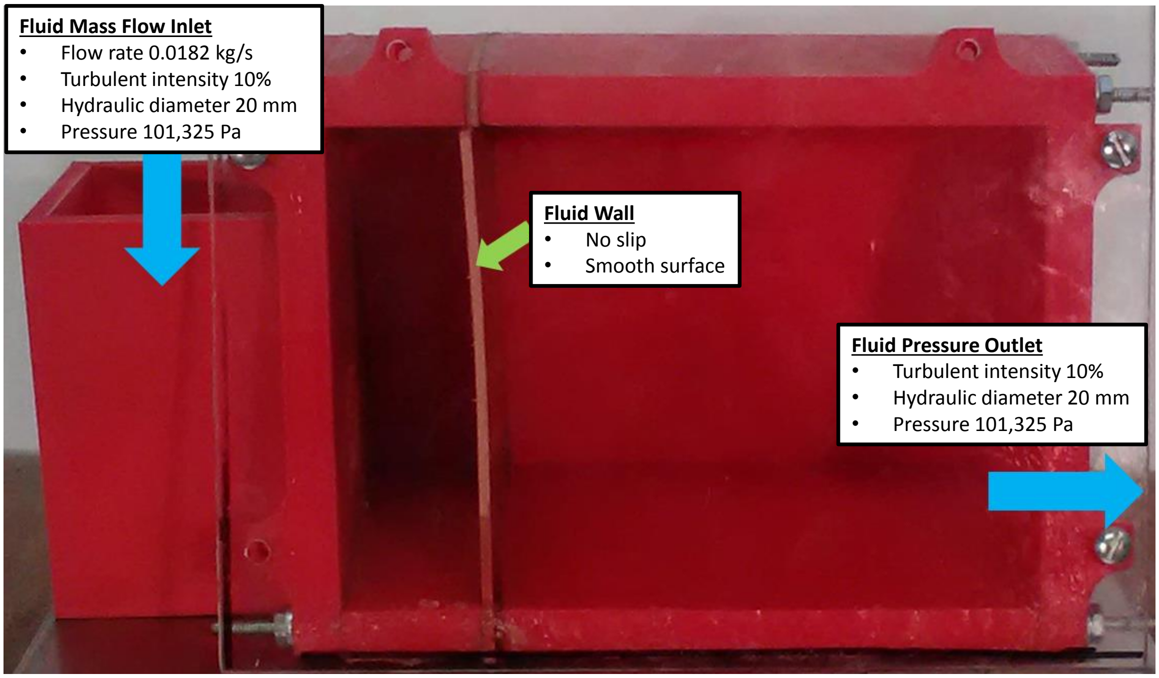


Figure 2.1: Schematic of polyethylene-based hyperelastic flap acting as a dam that resists the hydrostatic forces produced by the fluid column.

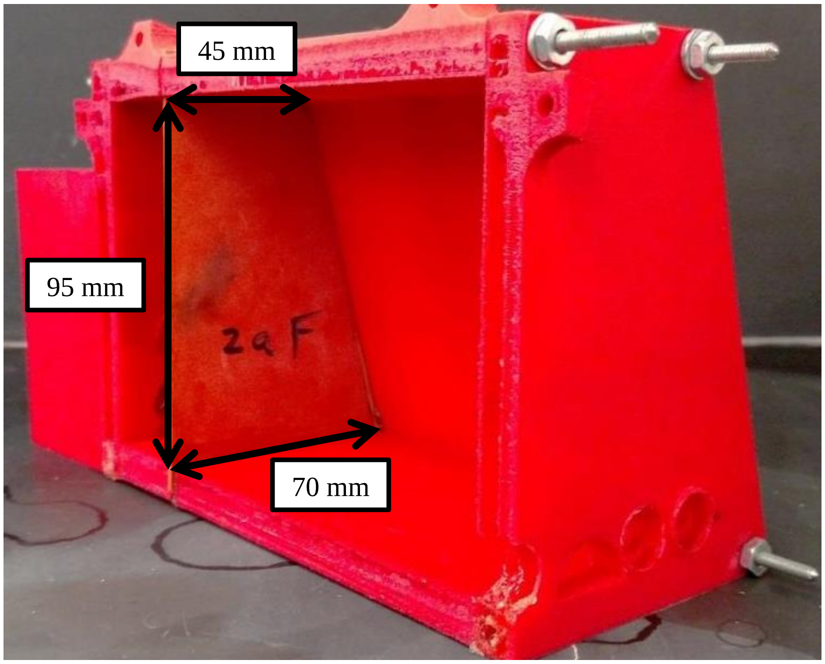


Figure 2.2: Image of test fixture with the 1/16 inch trapezoid flap with dimensions.

2.4. Methods

To set up an FSI model and evaluate it in the most efficient manner possible, several preliminary tests were conducted to evaluate the fluid domain and solid domain separately. This separate testing was done to ensure each independent model ran without failure and the resulting outputs were within a physically acceptable range. Only after the models were successfully implemented independently were they coupled using ANSYS System Coupling. For simplicity, this FSI model assumed that no fluid passes around the flap as it deflects as a function of fluid height; this will eliminate the need for modeling contact and the narrow fluid channel that forms as the flap moves away from the wall and fluid passes around the flap.

2.4.1. Experimental Methods

A custom-built test fixture was created from polylactic acid (PLA) and an acrylic glass sheet. The test fixture was designed so that the acrylic glass sheet was vertical, allowing for clear edge definition while imaging the test fixture throughout the duration of the experiment. The test fixture was designed to allow flap removal and replacement regardless of material thicknesses and types, Figure 2.2. The chamber was three-dimensionally printed using PLA so the exact dimensions were known for inputs into the computational model. Another reason PLA was selected is for its ability to withstand both water- and oil-based fluids.

Polyethylene-based rubber elastic flaps were cut from rubber gasket sheets and pre-fatigued to ensure that material hardening and fatigue did not affect the results. The pre-fatigue process consisted of flexing each flap between angles of ±180 degrees 100 times. A Phantom v7 camera with a 105 mm 1:2.8 Nikon lens was used to capture black and white images of the deflecting flap as a function of time. Images were captured at a rate of 90 frames per second, exposure time of 45 microseconds, and resolution of 800x600 pixels. The experiment was quantified by placing a grid with known spacing on the acrylic glass, so the fluid level and flap displacement could be measured. The deflection at three points along the height of the flap—30, 50, and 70 mm above the base—was measured as a function of fluid height at intervals of 2.5 mm starting at a height of 20 mm. No measurable deflection was present at fluid levels lower than 20 mm. The results were then averaged to determine the mean displacements and standard deviations.

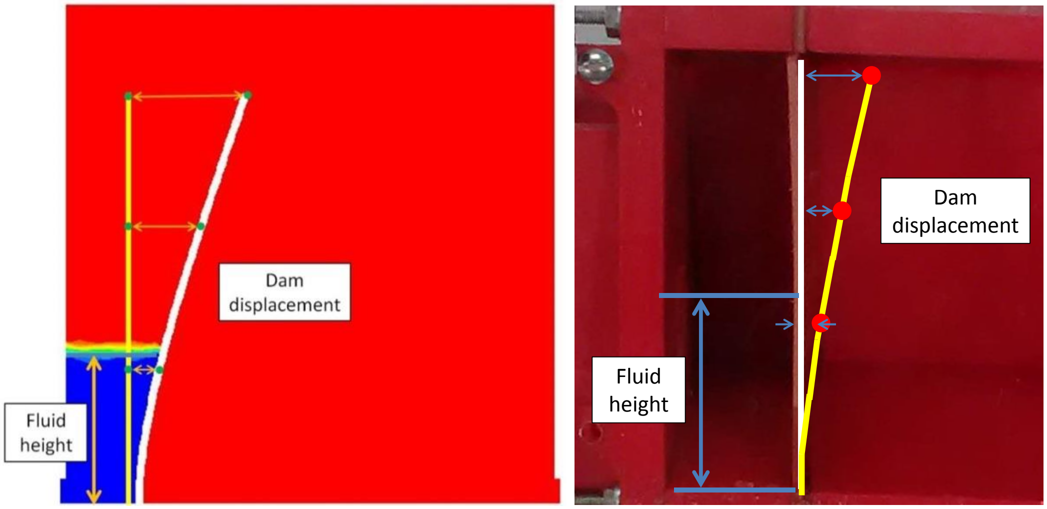


Figure 2.3: Deflected flap with three displacement points for measuring computational (left) and experimental (right) results.

2.4.2. CFD Numerical Methods

The fluid domain was evaluated using CFD modeling with the robust and commercially available software ANSYS Fluent 15, which is capable of solving complex fluid flow and heat transfer problems in three-dimensional geometries as a function of time. The pre-processor used for generating the geometry and mesh was ANSYS Design Modeler, which will be discussed in detail in the section titled “Interface between Fluid and Solid Domain.”

The computational model containing the fluid domain utilized the three-dimensional Navier–Stokes equations in conjunction with the continuity and volume fraction equations outlined in Chapter 1, Equations 1-3 respectively, utilizing a pressure-based solver due to the subsonic incompressible flow, along with the k-epsilon turbulence model. In the pressure-based solver, the momentum and continuity equations, Chapter 1

Equations 1 and 2 respectively, were used in combination to calculate the pressure field. Additionally, because the model contains two fluids with a distinct interface, the volume fraction equation must be solved to conserve species, and the mass balance equation must be evaluated to conserve the overall mass of the system, Chapter 1, Equations 3 and 2 respectively. The k-epsilon turbulence model utilizes Chapter 1, Equations 5 and 6 to define k and epsilon respectively, and Chapter 1, Equation 7 to define the turbulent viscosity. A turbulence model was used because turbulence and recirculation was present in the liquid region as the fluid height increased, while the bulk of the fluid model operated under laminar conditions.

Each of these equations is defined for water and air at each control volume prescribed by the fluid mesh, resulting in two sets of equations being evaluated over the entire domain. The size of the three-dimensional mesh is approximately 520,000 tetrahedron elements, but this number varies as the flap deforms and dynamic meshing occurs, Figure 2.4. The material properties for compressible water and canola oil used in the computational model are defined in Table 2.1. The model was evaluated using standard relaxation for pressure, density, body forces, momentum, turbulence kinetic energy, turbulent dissipation rate, and turbulent viscosity of 0.3, 1.0, 1.0, 0.7, 0.8, 0.8, and 1.0 respectively, until all scaled convergence values were below 1.0E-3. Defining water and oil as compressible liquids provided needed stability to the computational model by reducing the mass effect experienced at the FSI interface. Oil is less dense than water, so at the same fluid height, the resulting hydrostatic pressure acting on the flap is less, thus the deflections are smaller and a greater fluid volume is present for the mass effect to be dampened over.

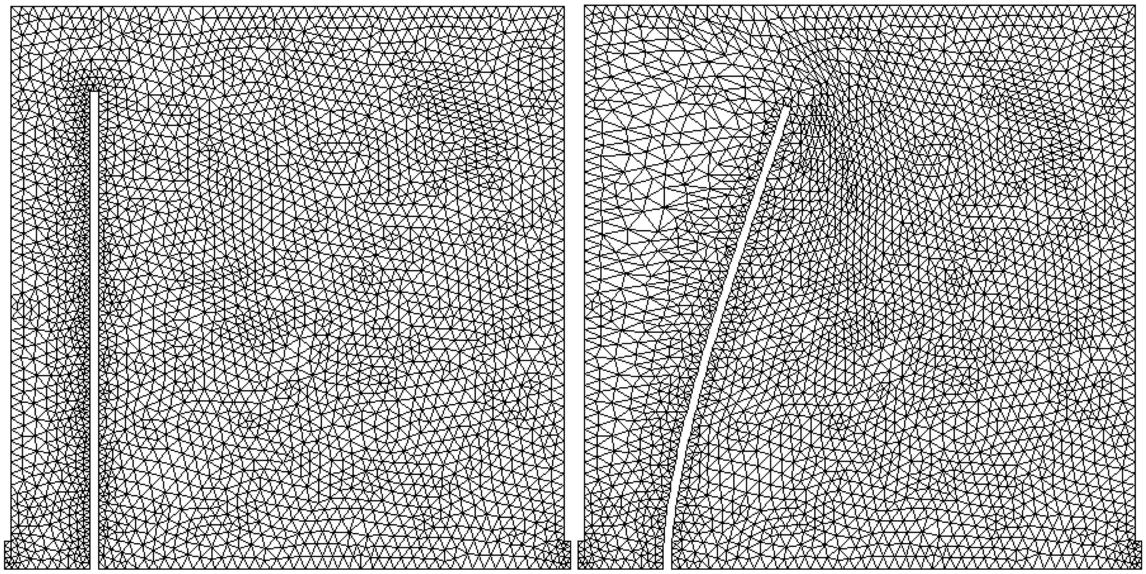


Figure 2.4: Image of the initial mesh before deformation occurs (left) and 0.85 seconds (right). A finer mesh was desired at the fluid inlet and outlet and along the walls of the flap, while a courser mesh was desired through the bulk of the fluid.

Table 2.1: Fluid properties for the water and oil used in the computational models [4,74].

Water	Viscosity (kg/m-s)	1.00E-03
	Reference Pressure (Pa)	101325
	Reference Density (kg/m^3)	998.2
	Reference Bulk Modulul (Pa)	2.20E+09
	Density Exponent	7.15
Canola oil	Viscosity (kg/m-s)	7.16E-02
	Reference Pressure (Pa)	101325
	Reference Density (kg/m^3)	915
	Reference Bulk Modulul (Pa)	2.20E+09
	Density Exponent	7.15

The fluid domain was initially evaluated independently of the solid domain and without dynamic meshing. This uncoupled CFD model allowed for greater understanding of the mesh cell size sensitivity, convergence criteria as a function of flow rate, and required convergence time as a function of time step size and number of iterations. The information gathered from evaluating just the CFD model without dynamic meshing provided valuable insight into what time step size and flow rate allowed the optimal balance between a reliably stable fluid solution and overall computational time required to evaluate the model.

The computational time required to evaluate the model is a function of the total number of time steps required (time step size) and time required to evaluate each time step (computational time per time step). While increasing the time step size does reduce the number of time steps required, increasing the time step size also increases the computational time per time step. Therefore, a balance between increasing the time step size while only marginally increasing the computational time per time step is paramount to evaluating the computational model in as little time as possible. This understanding of the time step is important because once the fluid and solid domains are coupled, the same time step must be used to evaluate each domain. Although the computational time required to evaluate the CFD models independently may not be significant, when the CFD and FEA models are coupled together, the computational time increases exponentially, thus making small increases in computational efficiencies important.

Finally, a mesh density investigation of the CFD domain was performed when the fluid domain was uncoupled in order to explore the proper mesh density and assess which portions of the model required a finer mesh and which regions could tolerate a courser mesh. Because a large fluid domain was present, it was not desirable to have a uniformly fine mesh over the entire domain. Figure 2.4 shows the desired mesh at time zero before deflection occurs. The mesh density investigation indicated the size and regions where course and fine mesh required implementation, Figure 2.4.

2.4.3. FEA Numerical Methods

The solid domain was evaluated using FEA with the commercially available software ANSYS Mechanical 15. The pre-processor used for generating the geometry and mesh was ANSYS Design Modeler, which will be discussed in detail in the section titled "Interface between Fluid and Solid Domains."

The computational model utilized the three-dimensional strain displacement, nodal displacement, and stress equations, Chapter 1, Equations 8-10 respectively, to solve for the deformation, stress, strain, and forces across each node in the model. The solid mesh contains approximately 4,400 HEX20 elements, with a thickness of six elements in the bending direction.

The solid domain was initially evaluated uncoupled from the fluid domain with a point load applied to the top of the flap. This point load was a function of time and increased linearly from 0 to 0.1 N over 2 seconds, Figure 2.5. This uncoupled model was used to understand the stability and limitations of the hyperelastic material, determine proper time step size, and perform a mesh density study. Similar to the CFD domain, the relationship between time step and iterations per time step was explored to determine the most efficient time step combination for evaluating the computational model. A mesh density study was performed to determine the minimum number of elements required in the bending direction of the flap to produce accurate results.

The gap that forms between the edge of the flap and the test fixture was ignored in order to avoid modeling the contact and small fluid channel that formed around the flap as it deflected. This assumption differs from the actual experimental operations where fluid passes around the flap, but avoids modeling the complexity of contact and fluid channel constrictions due to contact separation. For a detailed discussion of the causes of these complexities due to contact, see the section in Chapter 1 titled "Solid Contact in FSI Models."

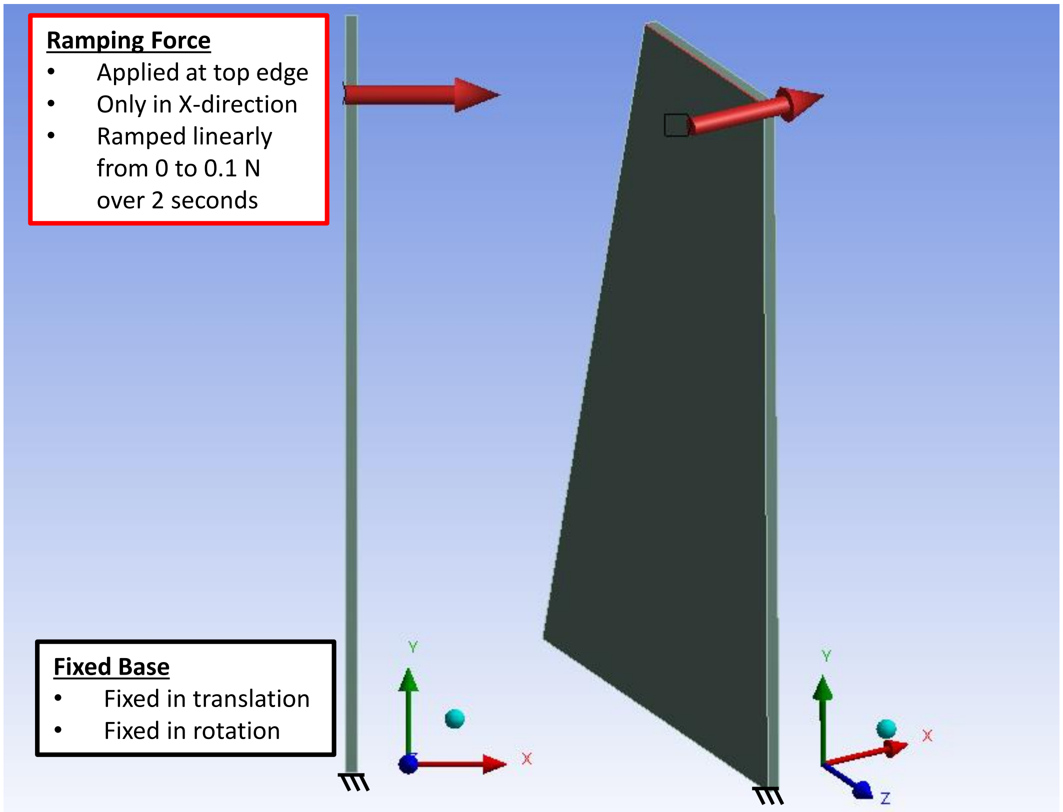


Figure 2.5: Ramping point load as a function of time applied at the upper edge of the hyperelastic flap.

2.4.4. Interface between Fluid and Solid Domains

The pre-processor used for generating the geometries and mesh for both the fluid and solid domains was ANSYS Design Modeler. This commercially available software package has features that allow the fluid and solid domains to be created together and then separated for meshing and analysis. This enables faces to be linked for a more seamless data transfer and interface compliance in the FSI coupling algorithms. ANSYS Design Modeler also contains the capability to generate a conformal mesh.

ANSYS System Coupling was used to transfer data between the fluid and solid domains in a weak-coupled manner at each time step with data transfer once per time step. A relaxation factor of 1.0 was used for transferring data between each domain, i.e. no ramping function or damping was introduced and the full loads were applied at each data transfer. The fluid domain provided pressure loads to the solid domain and the solid domain provided nodal displacements to the fluid domain at each 0.005 second time step. A side view of the fluid domain tetrahedron element mesh at time zero and 0.85 seconds is depicted in Figure 2.4.

Due to the thin-walled nature of the flap, the stability number (Chapter 1, Equation 11) of this model was 0.1, resulting in a potentially unstable FSI model. To mitigate solution divergence due to the geometric configuration, the model was evaluated using the weak-coupled manner with the liquid allowed to exhibit minor compressibility behavior.

Upon completion of modeling the CFD and FEA models separately, the models were coupled together; however, the ramping point load was kept active in the FEA model, fluid flow was disabled in the CFD model, and dynamic meshing was enabled in the CFD model. These changes were made for two reasons: First, the ramping force in the FEA model would immediately cause deflection of the fluid domain, thus testing the dynamic meshing parameters of the CFD domain; and second, the fluid flow was a potential source of instabilities and added unnecessary complexity. If failure did occur while using this setup, the user would immediately know whether the failure originated from the dynamic meshing or from the system coupling setup.

The methods by which dynamic meshing is performed in Fluent are completely different when they are evaluated with two-dimensional versus three-dimensional models. When a two-dimensional model is used, the dynamic meshing is performed on a surface mesh alone, whereas in a three-dimensional model the dynamic meshing is performed on surface mesh and volume mesh [57]. Many software suppliers and publications provide tutorials outlining two-dimensional dynamic meshing techniques for surface meshing, but few tutorials are provided for three-dimensional volume meshing. The challenges associated with dynamic meshing arise from the difficulty in maintaining mesh quality on both surface and volume meshes. When using three-dimensional dynamic meshing in Fluent, a structured mesh cannot be used, i.e. tetrahedron elements must be used as opposed to hexahedron or wedge-shaped elements. For this reason, a tetrahedron mesh was used in Fluent, Figure 2.4. It is well-documented that maintaining element quality in a CFD and FEA mesh is critical and if the element quality does become poor, instabilities can occur [15,57]. In CFD and FEA modeling, these element quality issues can lead to rounding errors, negative volumes, and mass loss, causing the model to produce inaccurate answers, errors, and computational divergence. All computational anomalies generated in the CFD or FEA models are magnified at the interface. Therefore, it is critical to avoid poor element quality when evaluating an FSI model. To sustain a quality mesh throughout the large amount of deflection seen in the hyperelastic flap model, two dynamic meshing features were utilized: dynamic smoothing and remeshing of faces and volumes. When performing these dynamic meshing operations it is important to maintain the course and fine mesh in the proper location. If a fine mesh is generated uniformly across the entire volume, unnecessary increases in computational time will result. Additionally, it is desirable to only remesh the regions of the mesh where poor element quality is present. Dynamic meshing operations are a function of the total number of remeshed cells, so reducing the area or volume over which the remeshing is performed can save computational time. Table 2.2 outlines the dynamic meshing parameters and locations where meshing was achieved.

Table 2.2: Parameters used for initial meshing, dynamic remeshing, and dynamic smoothing.

Location	Minimum cell size (mm)	Maximum cell size (mm)	Maximum skewness	Dynamic smoothing - diffusion parameter
Inlet	1.0	1.0	0.7	0.5
Outlet	1.0	1.0	0.7	0.5
Oil side of dam	0.5	1.0	0.7	0.5
Air side of dam	1.0	1.5	0.7	0.5
Bulk fluid	0.5	3.0	0.7	0.5

2.4.5. Solid Material Properties

The material properties of polyethylene-based rubber vary widely based on the chemical makeup of the rubber, resulting in a wide range of material elasticity and nonlinear behavior [75,76]. Table 2.3 shows published material properties data for polyethylene-based materials. To achieve matching results between the computational model and the experiment, the elasticity of this material needed to be precisely defined. To determine the elasticity, an investigation was performed to first determine the best elasticity model to define the hyperelastic behavior of the flap, and second, an optimization was performed to tune the values used to characterize the elasticity model. Due to the uncertainty inherent with material properties behavior, a two-term Neo-Hookean stress–strain curve was used, allowing the optimization to determine whether a linear or nonlinear material best defined the experimental results.

Table 2.3: Published material properties data for polyethylene-based materials [77].

Name	Data Source	Mean (GPa)	Min (GPa)	Max (GPa)
Polycarbonate and Polyethylene Terephthalate Blend	Youngs modulous - MatWeb	2.655	1.16	4.15
High Density Polyethylene	Youngs modulous - MatWeb	0.805	0.510	1.100
Low Density Polyethylene	Youngs modulous - MatWeb	0.352	0.221	0.483
Very Low Density Polyethylene	Flexural Modulus - MatWeb	0.080	0.045	0.115
Chlorosulfonated Polyethylene Rubber	Modulus at 100% - MatWeb	0.012	0.0047	0.0189
Sanitary Gasket/O-Ring	Modulus at 100% - MatWeb	0.007		

2.4.6. Optimization of Material Properties

Multiple different optimization techniques are available as design tools for engineers to minimize an objective function with associated constraints. When coupling computational modeling with optimization, many optimization techniques cannot be used because the performance parameter—defined by the objective function—cannot be directly calculated, i.e. the computational model must be evaluated in order to determine the performance parameter. For this reason, a search-based optimization algorithm was used, which requires a starting point where the search will begin. A gradient-based search pattern was then performed using an iterative method until a minimum value was found within the user-defined tolerance. This optimization utilized the active-set algorithm because it allows for nonlinear gradients, which is expected as the design variables are perturbed [78]. This method does have limitations because it is a line search method that can result in finding local minimums rather than the global minimum. For this reason, the optimization was performed several times from different starting points to ensure the correct global minimum was found. To the author's knowledge, to date the tuning of material properties using optimization techniques has never been conducted on an FSI model.

The performance parameter used to drive the optimization of the flap properties was calculated by the squared difference between the experimental values and the computational values, Equation 12. Where η is the total error, d_{exp} is the deflection of the experimental point plus and minus the vertical error bars, and d_{comp} is the deflection of the computational point. The fluid height was measured at 2.5 mm intervals and the associated deflection was measured at three locations along the flap at each interval. This resulted in between 12 and 15 data points to compare the computational model and experimental results.

$$\boldsymbol{\eta} = \sum\left(\boldsymbol{d_{exp}} - \boldsymbol{d_{comp}}\right)^{\mathbf{2}} \tag{12}$$

The optimization algorithm was employed by Matlab, which was also used to interface the ANSYS software package containing Design Modeler, Fluent, Mechanical, and System Coupling for the FSI analysis. This was accomplished using custom scripting found in Appendix A and execution through the DOS command prompt. Figure 2.6 shows a flow diagram for the optimization routine with the FSI model receiving the material properties parameters before evaluating the model and providing the flap deflection data back to the optimization routine. The objective function was evaluated to minimize the squared difference between the computational and experimental deflection.

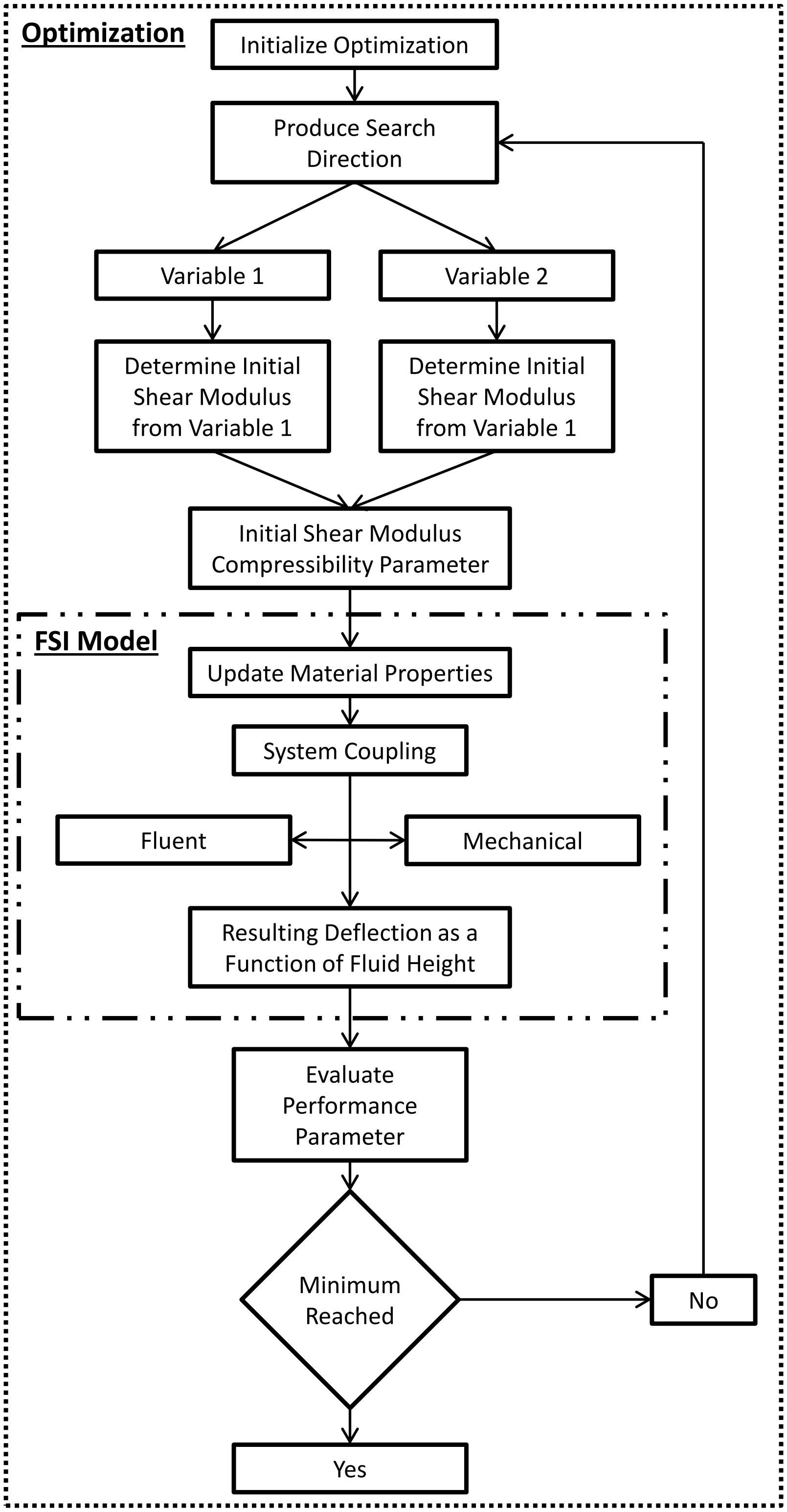


Figure 2.6: Diagram of the optimization controlled by Matlab that managed the FSI software linked by ANSYS System Coupling and custom scripts.

The optimization of the Neo-Hookean material properties utilized two variables to determine the best material property fit with the experimental data. Variable 1 represented the initial shear modulus and variable 2 represented the incompressibility parameter. Due to the orders of magnitude difference between the initial shear modulus and incompressibility parameter, normalized values were used to represent the initial shear modules and incompressibility parameter within the optimization routine. Once the optimization search direction was determined using the normalized values, the actual values used in the FSI model were than calculated from the normalized values. Both the normalized and standard values for the optimization setup are provided in Table 2.4. The flexible nature of the material resulted in an increasingly unstable model as the material became more flexible, thereby resulting in unconverged solutions and termination of the optimization search. To account for these instabilities and resulting failures in the search pattern, the optimization was evaluated five times from different starting points to ensure the global minimum was reached in a stable design space.

Table 2.4: Optimization search setup for optimization and equivalent values for the FSI model.

	Optimization Inputs	Resulting FSI Input
Minimum Step	0.01	-
Maximum Step	0.10	-
Performance Tolerance	0.01	1E-08 m^2
Lower Bound Variable 1	8.5	1.98 MPa
Upper Bound Variable 1	15.0	11.2 MPa
Lower Bound Variable 2	5.0	1.70E-05
Upper Bound Variable 2	15.0	5.10E-05

2.4.7. Final Multiphysics Model

Only after the fluid domain, solid domain, and forced coupling model were evaluated was the final multiphysics of the model applied. This process of evaluating each model independently built a greater understanding of the instabilities contributed by each portion of the model. In the process of combining the fluid and solid domains with the fluid flow, large amounts of instabilities due to the mass effect were introduced, as outlined in Chapter 1. To counteract instability due to the mass effect, the fluid was treated as compressible to both reduce the magnitude of the mass effect and dampen the shock from the mass effect across the bulk fluid volume. Oil was used to tune the flap properties because the density and viscosity does not promote as much rapid deflection, thus resulting in a more stable computational model. The model's performance accuracy was then tested with water as a more challenging simulation. Furthermore, altering the time step did not directly increase computational stability. It is hypothesized this is because smaller computational times reduce the rate or magnitude in which forces are applied from one time step to the next, thus increases the relative magnitude of the artificially produced instabilities. Figure 2.7 demonstrates this hypothesis by showing that the hydrostatic force acting on the flap is constantly trending up, while the fluctuations in the force values at smaller time steps cause oscillations. If time steps are small enough, these fluctuations become significant and can cause oscillations in the hydrostatic force, resulting in model failure. If appropriate time steps are selected, these oscillations are not observed, thus leading to added stability at larger time steps. The boundary conditions for both the fluid and solid domain coupled into the final multiphysics model were evaluated using a time step of 0.005 seconds until the fluid reached a height of 30.0 mm, Table 2.5.

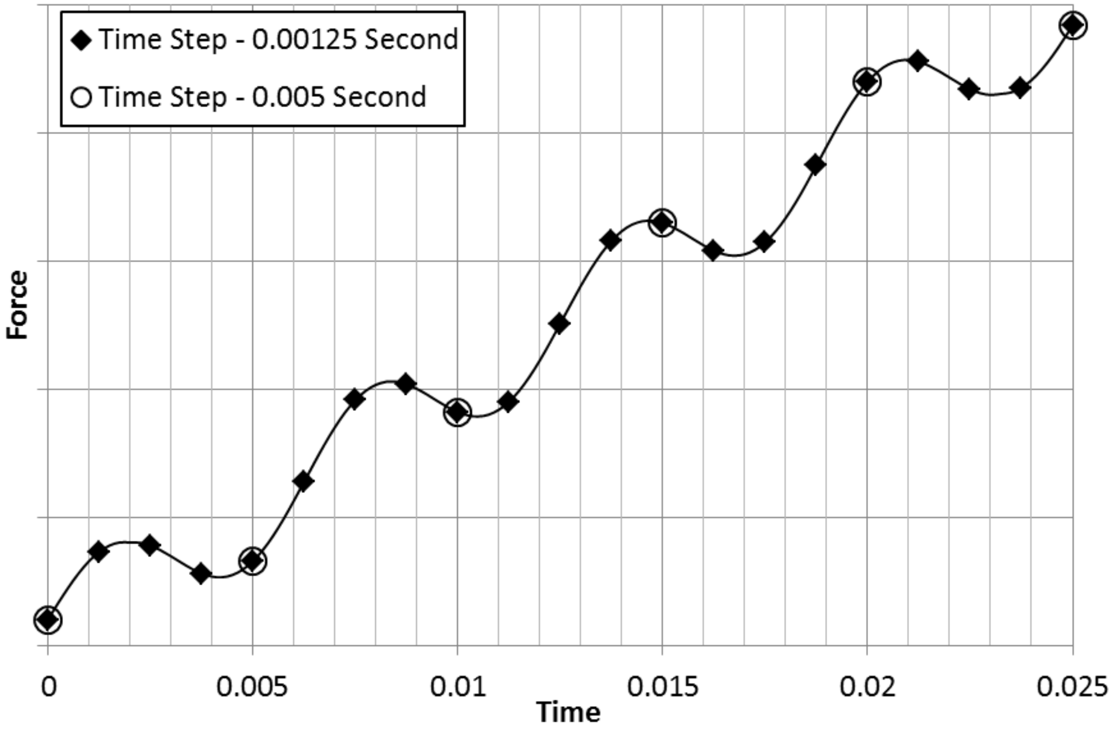


Figure 2.7: Forces as a function of time with time steps at 0.00125 and 0.005 seconds. If time steps are too small then oscillations are present, but if time steps are selected appropriately the oscillations are not observed.

Table 2.5: Boundary conditions settings for final multiphysics model for both the fluid and solid domains.

Boundary Condition Name	Domain	Type	Settings
Fluid inlet	Fluid	Mass flow inlet	Flow rate = 0.0182 kg/s Turbulent intensity = 10% Hydraulic diameter = 20 mm Pressure = 101,325 Pa
Fluid outlet	Fluid	Pressure outlet	Turbulent intensity = 10% Hydraulic diameter = 20 mm Pressure = 101,325 Pa
Fluid solid interface	Fluid	Wall	No slip Smooth surface Coupled with dynamic meshing to allow for motion
Solid base	Solid	Fixed support	Fixed in all degrees of translation Fixed in all degrees of rotation
Solid-fluid-interface	Solid	Fluid structure interaction	Free in all degrees of freedom Displacement values were passed to fluid domain

2.5. Results and Discussion

2.5.1. CFD Domain Results

In the computational model containing the CFD model alone, fluid passed in the model at a flow rate of 0.018 with a time step of 0.05 seconds. It was observed that a maximum time step size of 0.05 seconds could be used, but required significantly more computational time per time step compared to the 0.005 second time step. Therefore, it was more time efficient to evaluate the model using 0.005 seconds per time step. The results of the mesh investigation indicated elements along the inlet, outlet, and flap should have an edge length of 0.5 mm with elements no larger than 3 mm edge length in the bulk of the fluid. This provided enough resolution at the volume of fluid interface to determine the fluid depth as a function of deflection and a sufficient number of elements to allow proper flow characterization. The fluid domain initially started with 520,000 tetrahedron elements, but this number changed as a function of dynamic meshing. Figure 2.8 shows the results of the CFD model at a fluid level of 40 mm.

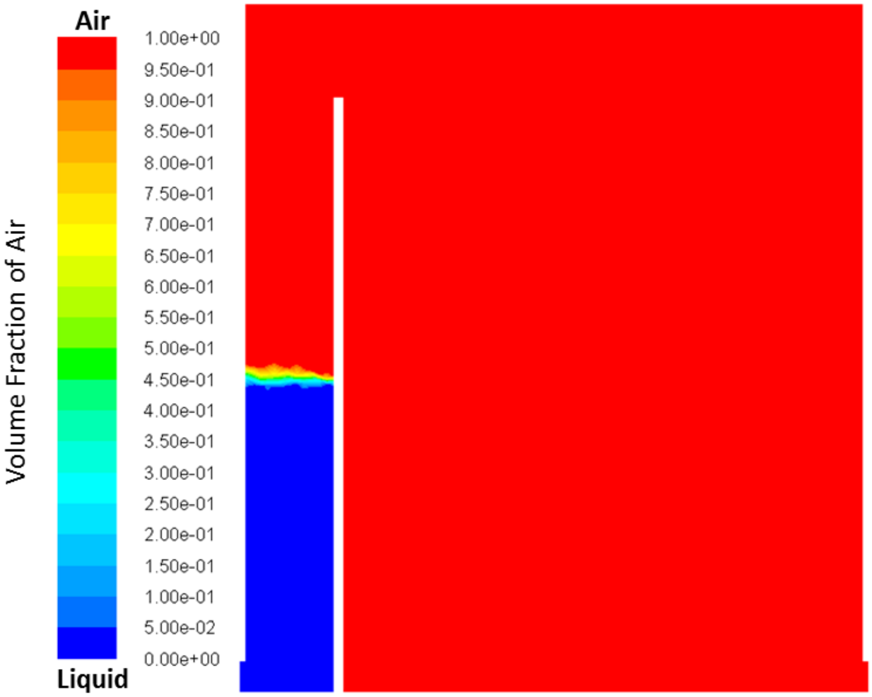


Figure 2.8: Volume of fluid contour plot for oil and air for the uncoupled CFD model with the fluid level at 40 mm.

2.5.2. FEA Results

In the computational model containing the FEA model alone with a point load ramping force, the most effective time step was 0.01 seconds. It was observed that a maximum time step size of 0.025 seconds could be used, but required significantly more computational time per time step compared to a 0.01 second time step. Therefore, it was more efficient to evaluate the computational model using 0.01 seconds per time step. The results of the mesh density study indicated four HEX20 elements in the bending direction of the flap produced sufficient results, Figure 2.9. However, the mesh density study only used a point load as compared to the distributed hydrostatic force, thus six elements were used in the bending direction, Figure 2.10. Figure 2.11 shows the deflection of the hyperelastic flap as a function of the ramping force without CFD-coupled results.

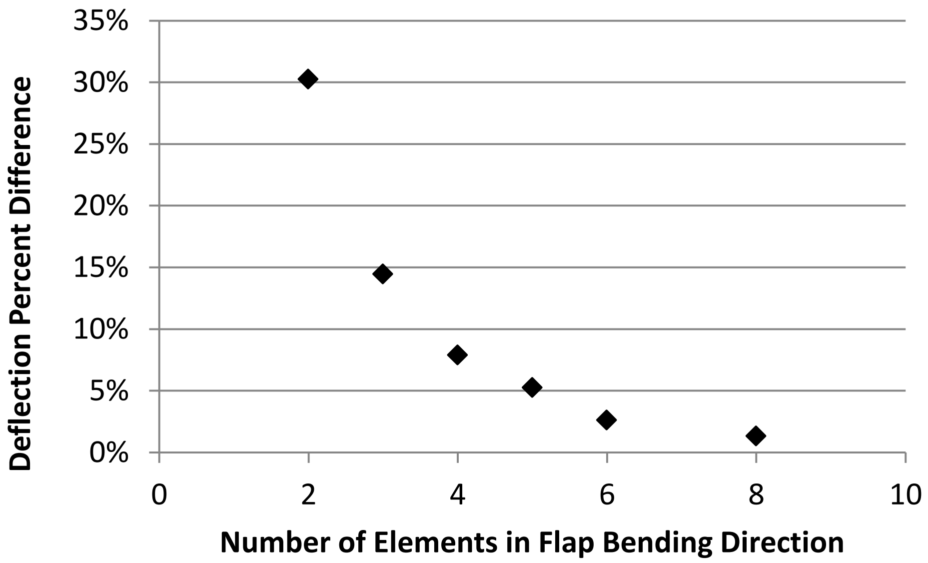


Figure 2.9: Results from mesh density study indicated four HEX20 elements in the bending direction produced accurate deflection results, but six elements were used.

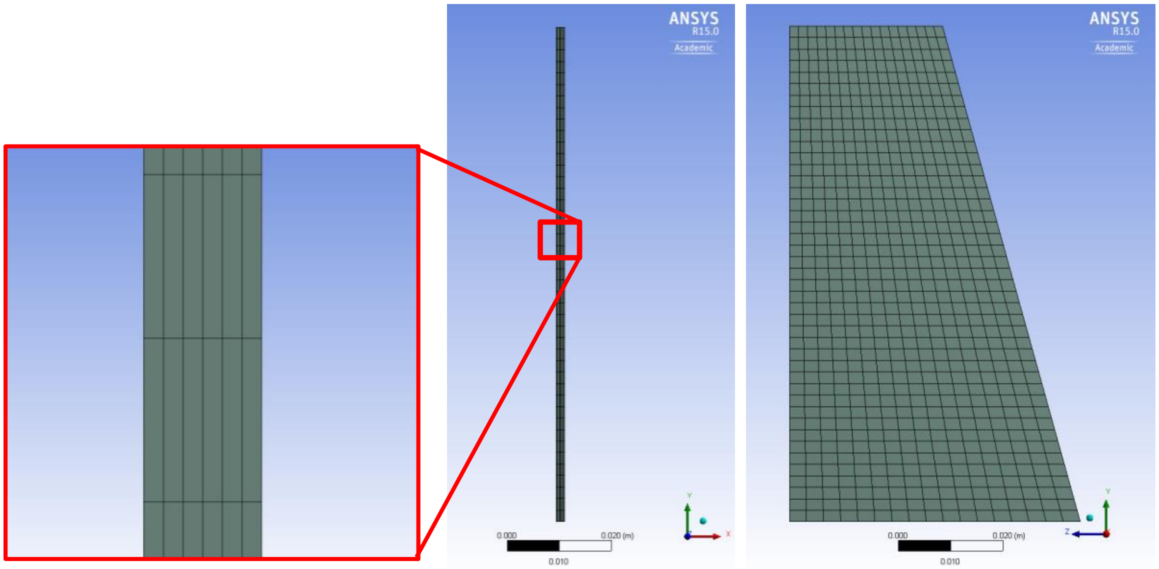


Figure 2.10: Image of the desired mesh after a mesh density study was performed.

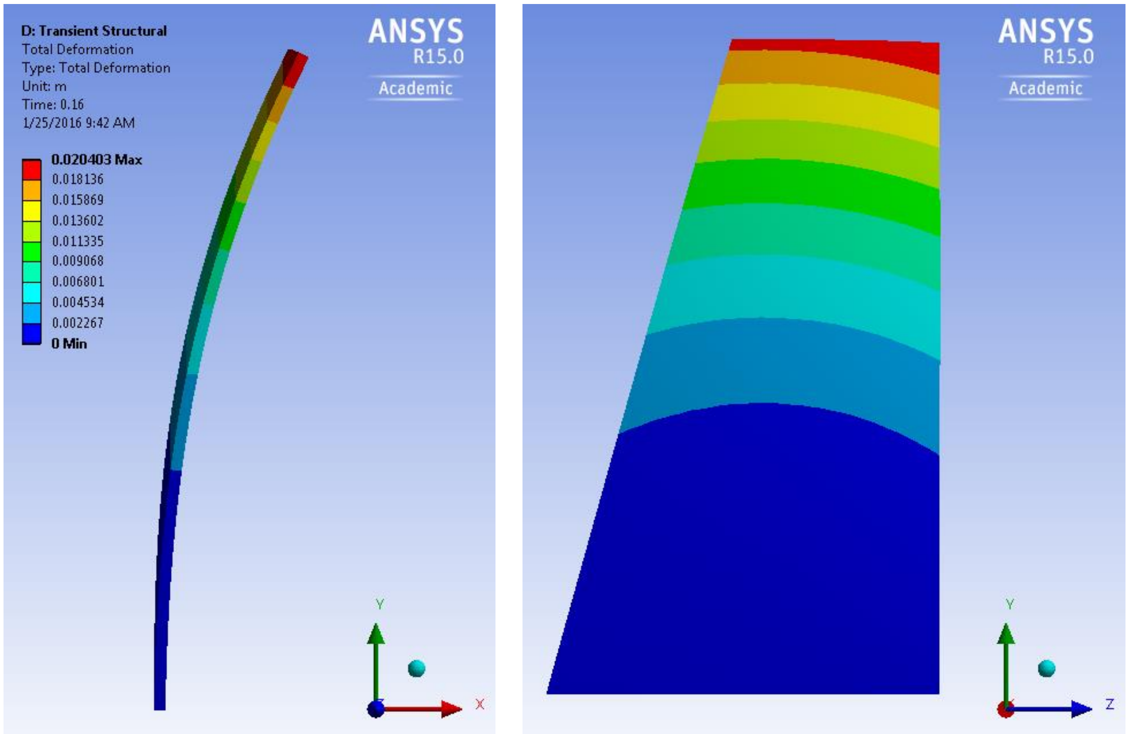


Figure 2.11: Deflection contour plot of the hyperelastic flap by a ramping point load for the uncoupled FEA model.

2.5.3. Forced Coupled Model

In the computational model containing the forced coupled CFD and FEA models, the dynamic meshing parameters were tuned to provide the most efficient meshing conditions while still maintaining element quality. It was observed that a time step of 0.005 seconds produced the most efficient results while still maintaining computational convergence. Larger time steps ran the risk of generating negative volumes in the fluid domain and too small of time steps produced oscillating forces at the interface. Figure 2.12 shows the fluid domain and solid domain at 0.52 seconds into the application of the point load.

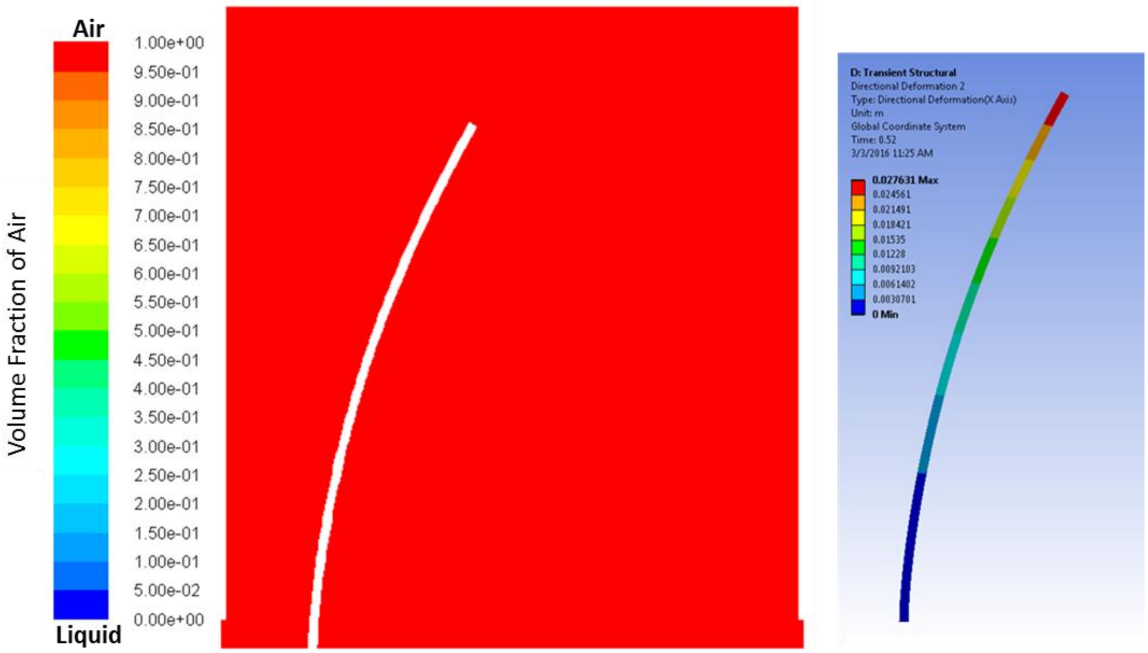


Figure 2.12: Volume of fluid contour plot for the forced coupled fluid domain (left) and x-directional deflection of the solid domain (right) at 0.52 seconds.

2.5.4. Experimental Results

Experimental tests were performed to measure the deflection of the flap as a function of oil fluid height. The results when using oil are shown in Figure 2.13. The mean deflection at an oil height of 30 mm was 15.1 mm and standard deviation of 1.2 mm. Vertical error bars are shown at plus or minus one standard deviation using a normal distribution for the variability in the experimental measurements. The horizontal error bars are shown at plus or minus 0.25 mm determined by the uncertainty of the fluid height measurements.

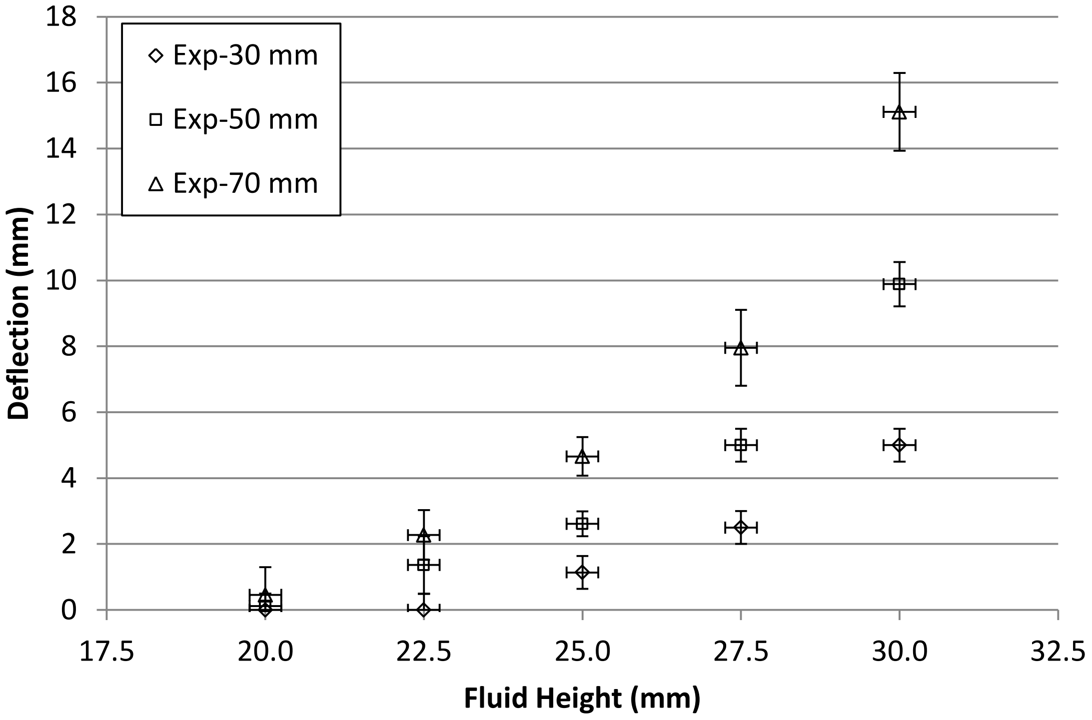


Figure 2.13: Experimental results for deflection of three points at five fluid levels using oil with vertical error bars at ±1 standard deviations and horizontal error bars at ±0.25 mm.

Using the same flaps and methods, the experiment was repeated six times using water rather than oil. Figure 2.14 shows the results for three points at four fluid heights using water. The mean deflection at a water fluid height of 27.5 mm was 12.9 mm and maximum standard deviation of 0.9 mm. Vertical error bars are shown at plus or minus one standard deviation using a normal distribution for the variability in the experimental measurements. The horizontal error bars are shown at plus or minus 0.25 mm determined by the uncertainty of the fluid height measurements.

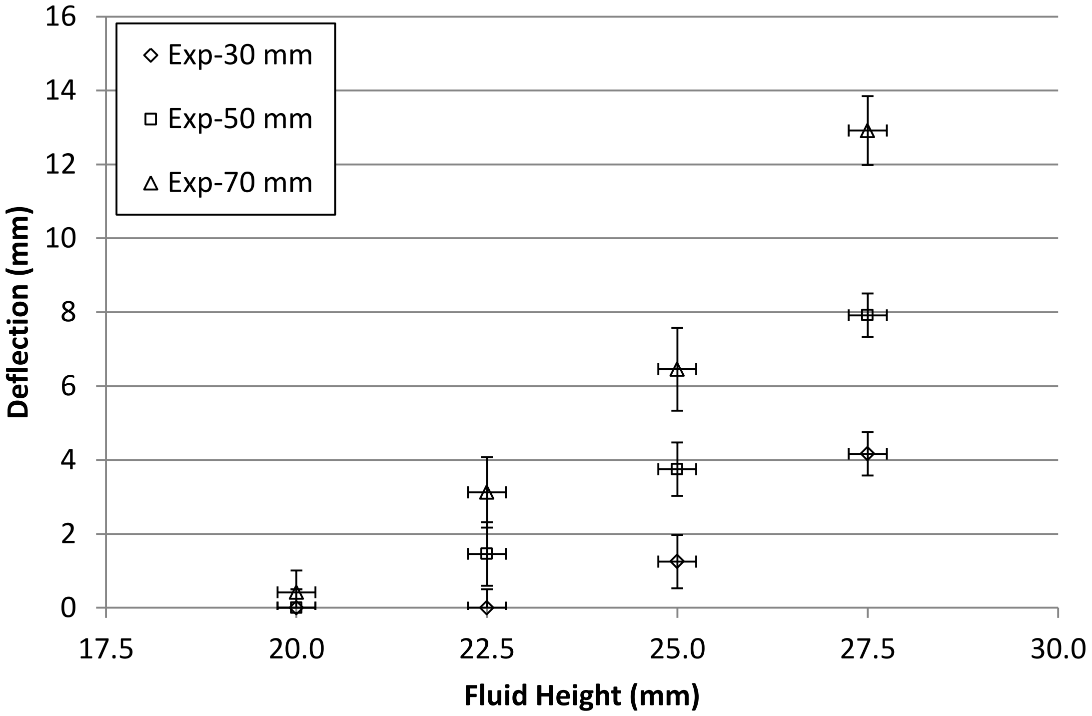


Figure 2.14: Experimental results for deflection of three points at four fluid levels using water with vertical error bars at ±1 standard deviations and horizontal error bars at ±0.25 mm.

2.5.5. Material Properties Optimization Results

The results for the material properties investigation determined that a nonlinear material properties classification provided the best results. A Neo-Hookean hyperelastic material properties definition was used to characterize the nonlinear material behavior. The results from the optimization of the Neo-Hookean parameters determined the initial shear modulus and incompressibility parameters to be 13.271 MPa and 3.4381E-5 respectively. This Neo-Hookean behavior aligns well with the linear characterization of the very low density polyethylene, Figure 2.15. The optimization evaluated 116 different conditions in order to converge to the optimal solution. The percentage error between the sums of the computationally measured distances over the sum of the experimentally measured distances is 1.3%. The numerical results with the optimal material properties and experimental results using oil are shown in Figure 2.16. The computational results fall within the error bars of the experimental results except at a fluid height of 30.0 mm when the computational results are outside the experimental error bars. This is due to the assumption that no fluid passes around the flap in the computational model, although this is not the case during the experiment. The reason this difference does not show up at lower fluid heights is the rate of fluid passing around the flap increases exponentially at higher deflections, thus the assumption becomes more important at 30 mm. Figure 2.17 shows the experiment at an oil level of 30 mm with the optimized computational results at 30 mm overlaid on top. Figure 2.18 shows the comparison between the deflection of the mean published material properties and the tuned material properties. Although only a small change in material properties was present, the deflection error in the matching between the computational model containing the mean value and the experimental results was 14.4%.

Table 2.6: Optimization starting points, optimal solution, and search information for the material properties optimization.

	Initial Conditions		Optimized Results				
	Initial Shear Modulus (Pa)	Incompressibility Parameter	Initial Shear Modulus (Pa)	Incompressibility Parameter	Iterations	Models Evaluated	Error (m^2)
Start point 1	1.29E+07	2.51E-05	1.46E+07	3.38E-05	3	13	2.40E-05
Start point 2	1.14E+07	5.70E-05	1.35E+07	3.32E-05	5	28	1.53E-05
Start point 3	1.98E+07	5.10E-05	1.32E+07	3.40E-05	6	34	1.10E-05
Start point 4	1.55E+07	4.43E-05	1.33E+07	2.81E-05	5	17	1.34E-05
Start point 5	1.32E+07	3.40E-05	1.32E+07	3.40E-05	0	24	1.10E-05

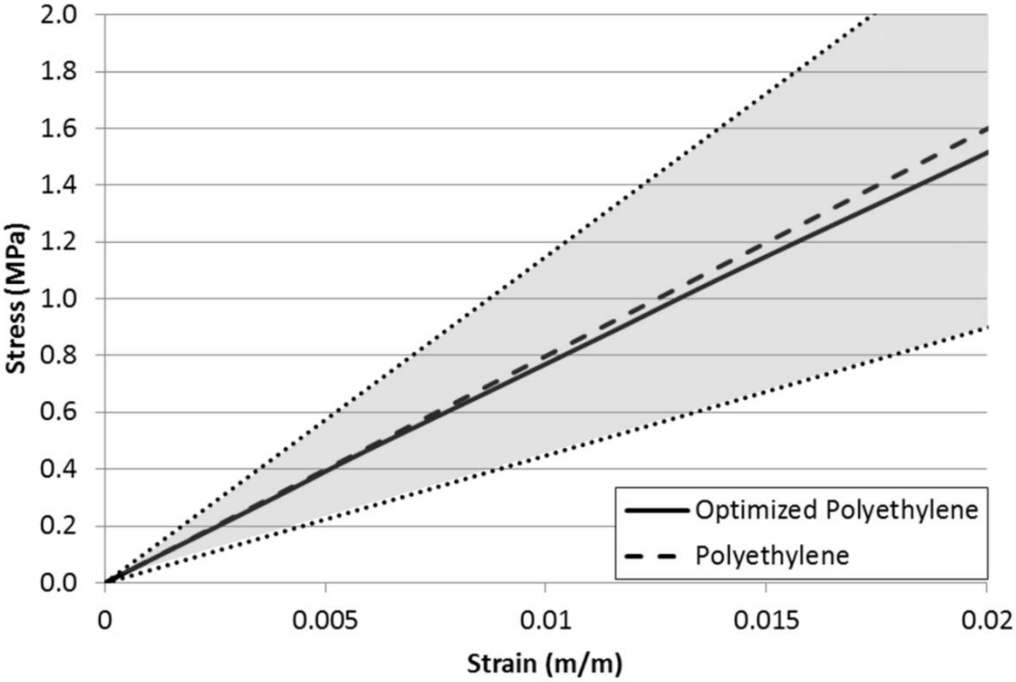


Figure 2.15: Stress–strain behavior for very low density and optimized polyethylene-based hyperelastic flap [79].

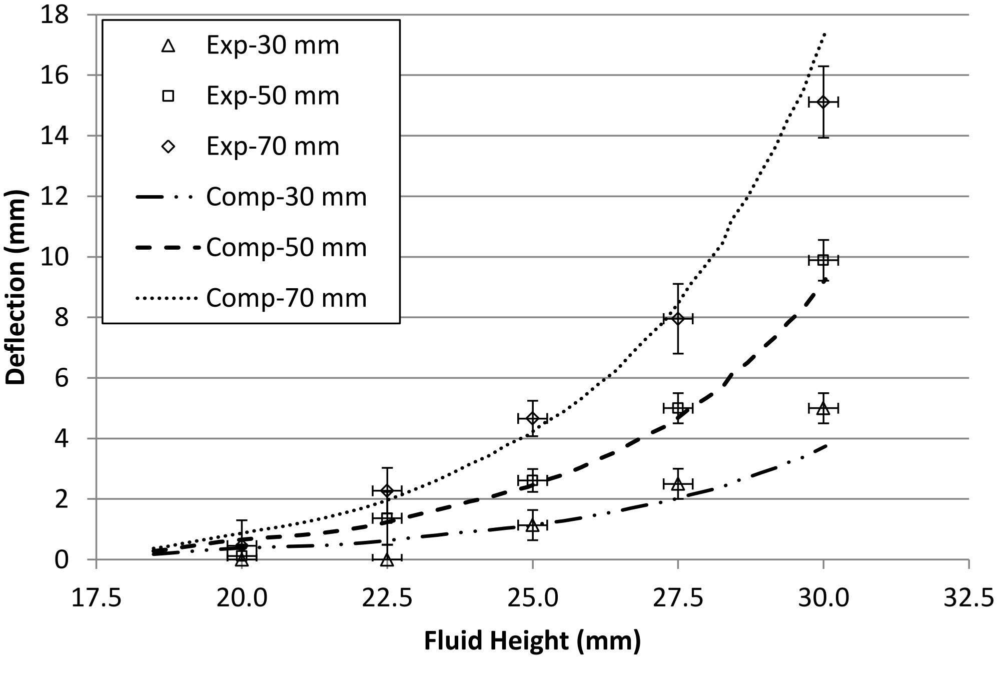


Figure 2.16: Computational results for deflection of three points as a function of oil height and experimental results for deflection of three points at five fluid levels.

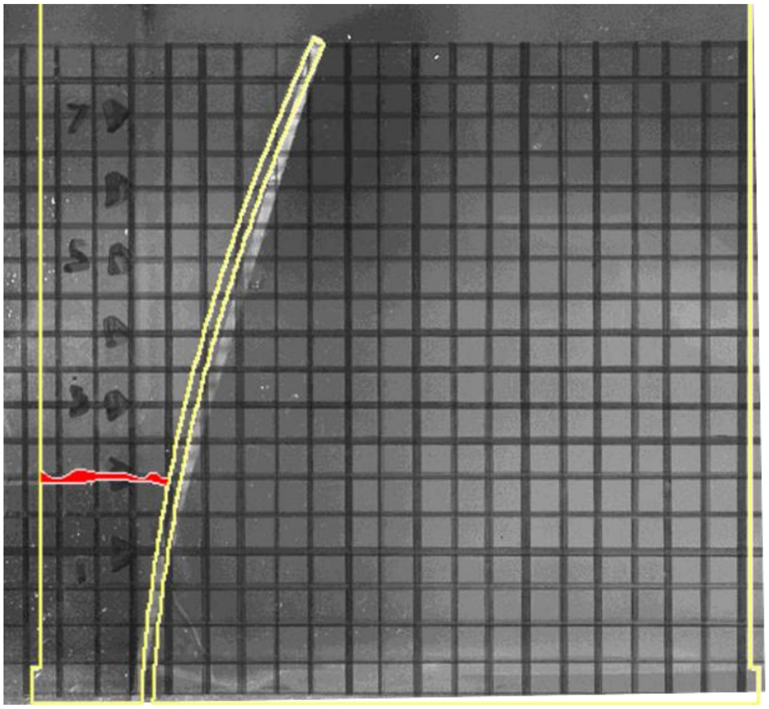


Figure 2.17: Image of experimental deflection and overlaid optimized computational deflection for oil at a fluid height (red) of 30 mm.

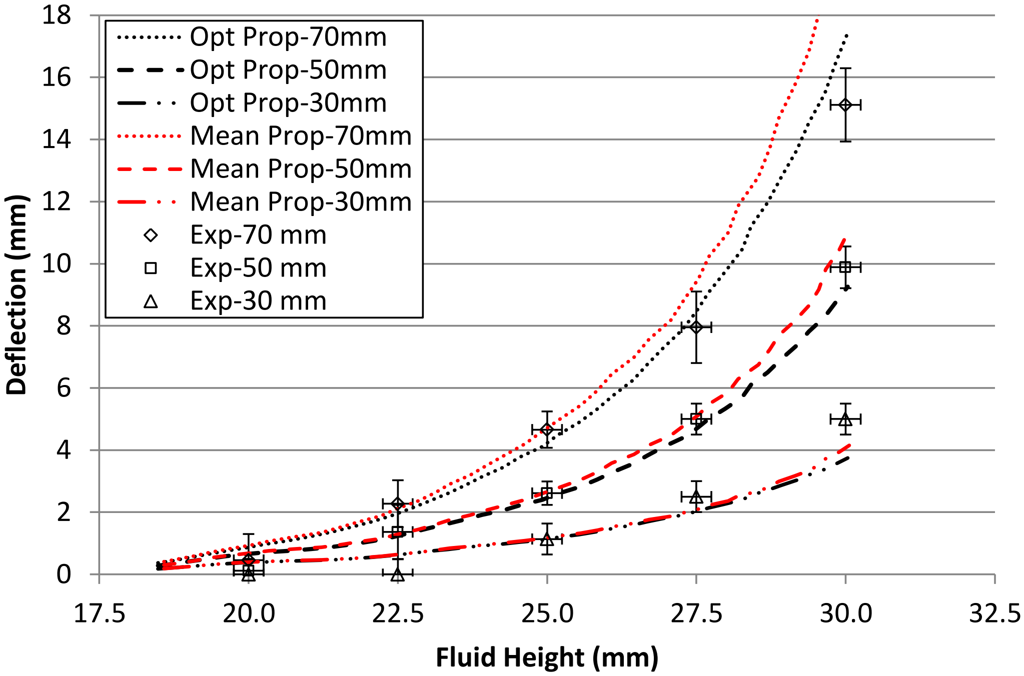


Figure 2.18: Computational results for the tuned material properties and the mean published values with experimental results.

2.5.6. Demonstration of Model Robustness with Water

Using the optimized material properties evaluated from oil, the computational model was reevaluated using water to create the hydrostatic forces. Figure 2.19 shows the computational and experimental results under these conditions. The computational results fall within the error bars of the experiment except at 27.5 mm, indicating the material properties calculated during the oil experiment match the physical system. At a fluid height of 27.5 mm the computational results fall near the edge of the experimental error bars. Again, this is due to the assumption that no fluid passes around the flap in the computational model, although this is not the case during the experiment. The reason this difference does not show up at lower fluid heights is the rate of fluid passing around the flap increases exponentially at higher deflections. The percentage error between the sums of the computationally measured distances over the sum of the experimentally measured distances is 1.8%. This error is only slightly larger than the error when oil was used. Figure 2.20 shows an image of the experiment at a water level of 27.5 mm with the computational results at 27.5 mm overlaid on top.

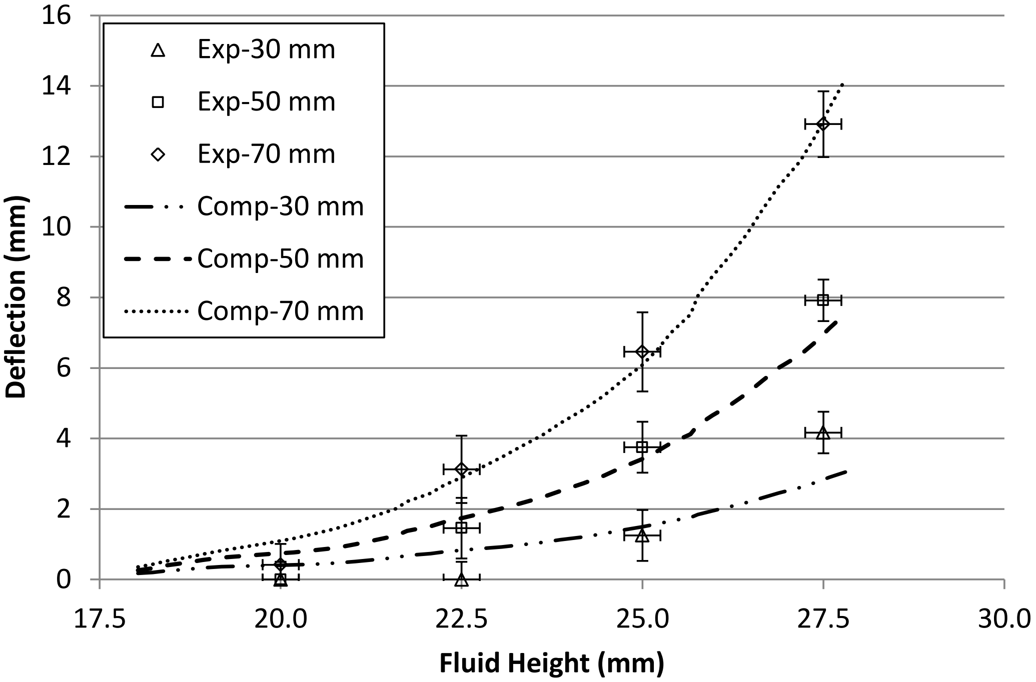


Figure 2.19: Computational and experimental data for the deformed flap caused by water.

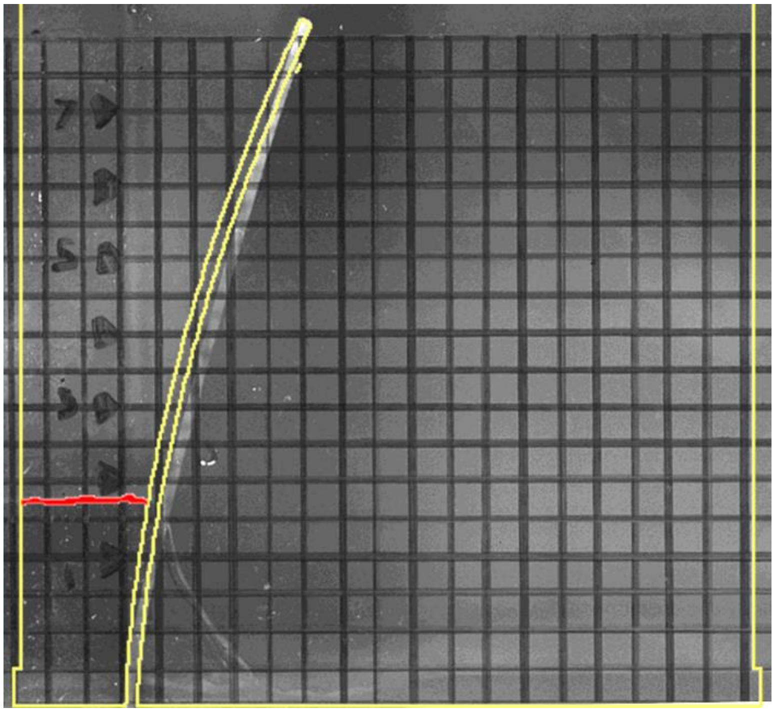


Figure 2.20: Image of experimental and overlaid computational deflection for water (red) at a height of 27.5 mm.

2.6. Conclusion

This study presented the setup and evaluation of a thin-walled FSI model with accompanying experimental data for quantitative validation. The geometry of this system allowed for a true three-dimensional interaction between the fluid and solid domains as the hyperelastic flap deflected as a function of the increasing hydrostatic pressure behind the flap. The hyperelastic material properties of the flap combined with its relatively thin geometry show that fluid–structure interaction modeling can be performed and validated for flexible thin-walled geometries. By tuning the hyperelastic flap material properties via an optimization, it was demonstrated that an FSI model can be created in a robust manner to accurately predict the results of related models. The results from these experiments and computational models show that with the proper implementation of instability mitigation techniques, a thin-walled FSI computational model can be evaluated, calibrated, validated, and used to accurately predict the results of related models. Step one of this study tuned the material properties to achieve accurate results when oil was used and a validation of the tuned model was performed by evaluating it with water to create the hydrostatic force. This demonstrated a quantitative validation of a FSI model with experimental results.

2.7. Lessons Learned

The methods described in this chapter lay the foundation for how future FSI models should be constructed, evaluated, and validated. Producing an accurate FSI model requires far more than the full multiphysics and final computational models alone.

Exploring and understanding the CFD and FEA models independently of each other was paramount to understanding possible sources for instabilities, evaluating the models in the most efficient time possible, and validating the mesh, boundary conditions, and material properties. For this reason each model should be evaluated independently before evaluating the coupled model.

Traditionally, sources for errors in CFD modeling originate from boundary conditions and mesh quality, while errors in FEA modeling originate from mesh quality and material properties. In FSI modeling the major sources for errors and discrepancies between experimental and computational results are material properties and mesh quality.

FSI models with stability numbers less than 1.0 can be accurately modeled by taking steps to add stability, including using compressible fluids, selecting appropriate time steps and data transferring methods, maintaining sufficient element quality, and avoiding material contact and separation in the FEA model.

The computational time required to evaluate an FSI model can be greatly reduced by selecting an appropriate time step, only remeshing regions with low quality elements, and maintaining a non-uniform mesh size with course and fine regions.

Dynamic meshing in a three-dimensional Fluent model requires an unstructured mesh and two dynamic meshing methods. Dynamic smoothing accounts for small displacements while dynamic remeshing allows for regions to be remeshed in order to maintain element quality. It is also important to know the proper location where this remeshing needs to occur and the preferable element size at these locations to produce an accurate answer in the shortest time possible.

CHAPTER 3: LIFETIME DESIGN FOR COUPLED FLUID–STRUCTURE INTERACTION MODEL UNDER THERMAL-CYCLIC LOADING

3.1. Abstract

Fluid–structure interaction modeling has become more available due to the increased computational power of modern computers and stability of algorithms employed. However, the a review of current literature found only limited instances of the use of fluid–structure interaction modeling to compute lifetime or design for performance at the end of life. This work utilizes a cyclic thermal load over a 1.0 s* time span to generate thermal expansion, material yielding, and temperature-dependent material properties to generate stress and strain fields in order to predict fatigue life. The transient computational modeling of this system was accomplished using computational fluid dynamics and finite element models linked with one-way coupling. A parametric study investigated material properties, geometric changes, and temperature profiles to determine the significance of various parameters on the life of the system. The parametric study demonstrated that the computational model is capable of capturing the effects of altered material properties, thermal boundary conditions, and geometry. The results of the parametric study indicated the coefficient of thermal expansion is the single most significant factor in lifetime performance by a considerable margin; therefore, it was unnecessary to perform an optimization because it would be dominated solely by the

coefficient of thermal expansion. This computational model and accompanying parametric study, in conjunction with evidence from the experimentally tested parts, demonstrated that a fluid–structure interaction model is capable of accurately predicting fatigue life and is robust enough to capture the effect of altered material properties, temperature profiles, and geometric variation.

3.2. Introduction

Thermal expansion is a common phenomenon where the atomic spacing in solid materials changes as a result of temperature fluctuation. The thermal expansion of a metal object does not always reduce the life of a part and may be desired to impart or elicit a desired function, e.g. biomedical switches and thermostats [80]. If a metal part is unconstrained and uniformly heated, the part does not experience stress as it thermally expands. However, if the part is constrained and/or temperature gradients are present, stress is produced in the part, potentially leading to material yielding and failure of the part. Fatigue due to thermal expansion is a significant problem in turbine blades, furnaces, heat exchangers, and large objects such as roads, bridges, airplanes, engines, and rigid tubing for fluid transport.

The idea of designing for lifetime is a well-established principle in Finite Element Analysis (FEA) modeling, but its application has not been used as extensively in Computational Fluid Dynamics (CFD) modeling. In CFD modeling, design optimization is well established, but instead of using optimization to design for system lifetime, it traditionally focuses on creating the optimal solution in terms of mixing efficiencies, optimal heat transfer, reduced drag forces, and increased coefficient of lift. Fluid systems are typically designed and built to operate at optimal conditions, but erosion, corrosion, oxidation, and fouling cause suboptimal conditions. Rarely are these conditions modeled and the overall performance optimized from beginning to end of life. Fluid–structure interaction (FSI) modeling results have not yet been coupled with lifetime performance optimization due to thermal fatigue failure even though commercially available codes are capable of such coupling [4]. FSI optimizations that have been performed have used one-way modeling with steady-state conditions [7,8,58]. This study demonstrates novel research by performing a one-way, transient evaluation over a 1.0 s* cycle.

3.3. Problem Normalization

The setup and results for this study were normalized as requested by our industry collaborator. Table 3.1 outlines the methods used to normalize the setup and results, where T represents the localized temperature, T_0 is the minimum temperature, T_∞ is the maximum temperature, t is the localized time, t_∞ is the final time, $life$ is the localized life, and $life_{base}$ is the life of the base condition.

Table 3.1: Normalization units and equations for presented data.

Quantity	Modeling Units	Normalization Method	Units Present
Temperature	Kelvin	$\theta = \frac{\mathrm{T} - T_0}{T_0 - T_\infty}$	Θ
Time	Second	$s^* = \frac{t}{t_\infty}$	s*
Life	Cycles	$\text{days} = \frac{life}{life_{base}}$	days

3.4. Problem Description

A burner containing a flame and cooled support structure was modeled and experimentally tested by our industry collaborator, Johns Manville, and at the University of Denver. The two-dimensional axisymmetric CFD model containing chemical kinetics, discrete flame structure, and internal cooling was evaluated by Johns Manville on a supercomputer taking over 30 days to evaluate a single 1.0 s* cycle. The resulting thermal load was then applied to a fatigue analysis to determine the lifetime of the burner. A parametric study was then performed to identify the affects various parameters had on the lifetime of the burner. The physical system is depicted in the top half of Figure 3.1.

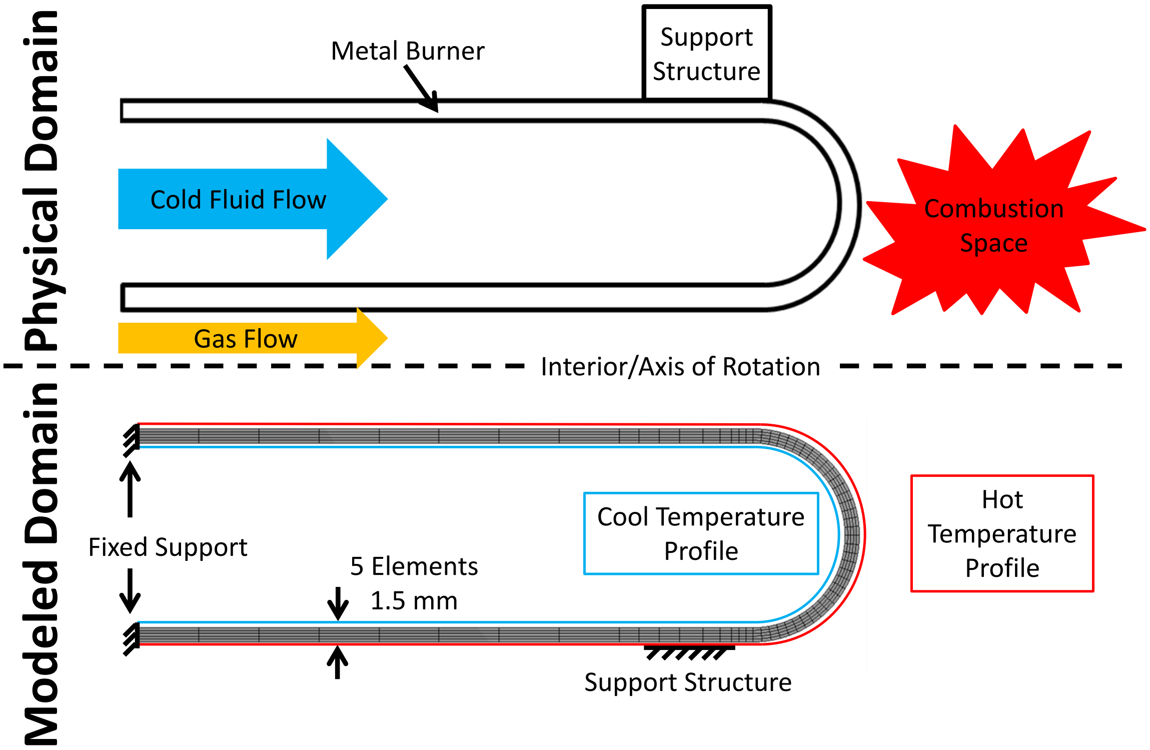


Figure 3.1: Diagram of the fluid and solid domains with boundary names.

3.5. Methods

An FSI model was evaluated over a 1.0 s* thermal cycle using CFD to apply the thermal conditions to an FEA model where thermal expansion and the stress–strain fields were calculated. From the results a fatigue life analysis was performed to determine the number of cycles until system failure. Finally, a parametric study involving material properties, geometric changes, and applied temperature profiles was conducted to compare the number of cycles before failure. It was desired to use a two-dimensional axisymmetric modeling domain, but ANSYS System Coupling does not allow for axisymmetric models to be coupled. For this reason, both models were evaluated as a quarter of the entire domain by utilizing two symmetry planes. The modeling domain is illustrated in the lower portion of Figure 3.1. This FSI model involves one-way data transfer using weak coupling from the fluid model to the solid model. Figure 3.2 shows the flow of data and the boundary dividing the one-way FSI model with results feeding into the strain-life analysis. Three key technical concerns were addressed in this model in order to obtain an accurate fatigue life: (1) thermal expansion as a function of temperature distribution, (2) temperature-dependent material properties, particularly fatigue properties, and (3) material yielding.

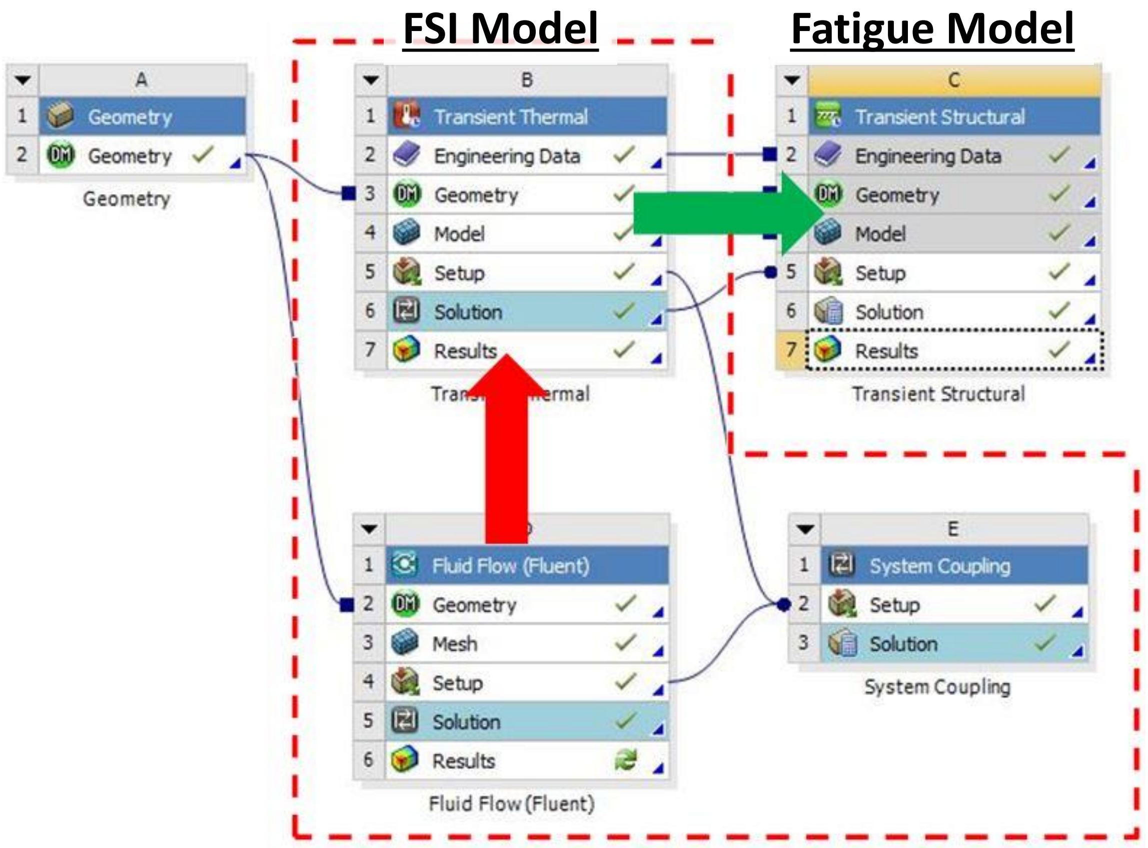


Figure 3.2: Flow of information for the one-way FSI model to the fatigue model containing the strain-life analysis.

3.5.1. CFD Numerical Methods

The fluid domain was evaluated using CFD modeling with ANSYS Fluent. The pre-processor used to generate the fluid geometry and mesh was ANSYS Design Modeler. Figure 3.1 shows a comparison between the physical system and computational modeling setup. Transient thermal boundary conditions were applied to the CFD model using data imported by a user-defined function attached in Appendix B. The fluid domain boundary conditions are presented in Table 3.2. Three different temperature profiles were applied at 0.0, 0.3, 0.5, and 0.75 s* during the evaluation of the 1.0 s* temperature cycle. Profile 1 was applied from 0.0-0.3 and 0.75-1.0 s*, Profile 2 was applied from 0.3-0.5 s*, and Profile 3 was applied from 0.5-0.75 s*. Figure 3.3 shows the three temperature profiles applied. The three profiles provided different temperature contours over the 1.0 s* cycle. However, the profiles did not capture the continually changing maximum temperature. Thus, the profiles were scaled by the CFD combustion model that contained 44,500 time steps in the 1.0 s* analysis, Figure 3.4. The continual scaling of data allowed for the application of a unique temperature boundary condition at each time step over the 1.0 s* cycle. Other data scaling and transfer methods were investigated, but this method was chosen because it allowed for the maximum data resolution while maintaining the steep time-dependent temperature gradients to be resolved. If these gradients were not resolved, material yielding due to the thermal gradients would cause reduced accuracy of the fatigue life prediction. This CFD model contained only a solid domain; thus, the energy equation found in Chapter 1, Equation 4 was evaluated with a relaxation factor of 1.0 until the scaled convergence value was below 1.0E-6.

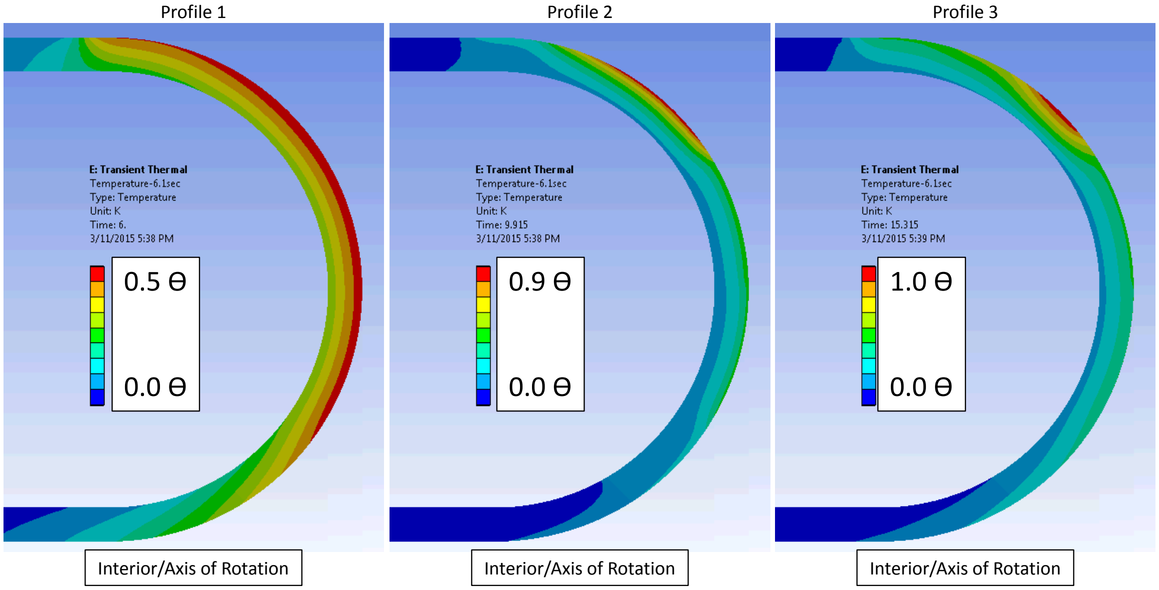


Figure 3.3: Three temperature profiles were scaled, so the maximum temperature at each time point reflected the maximum temperature as a function of time.

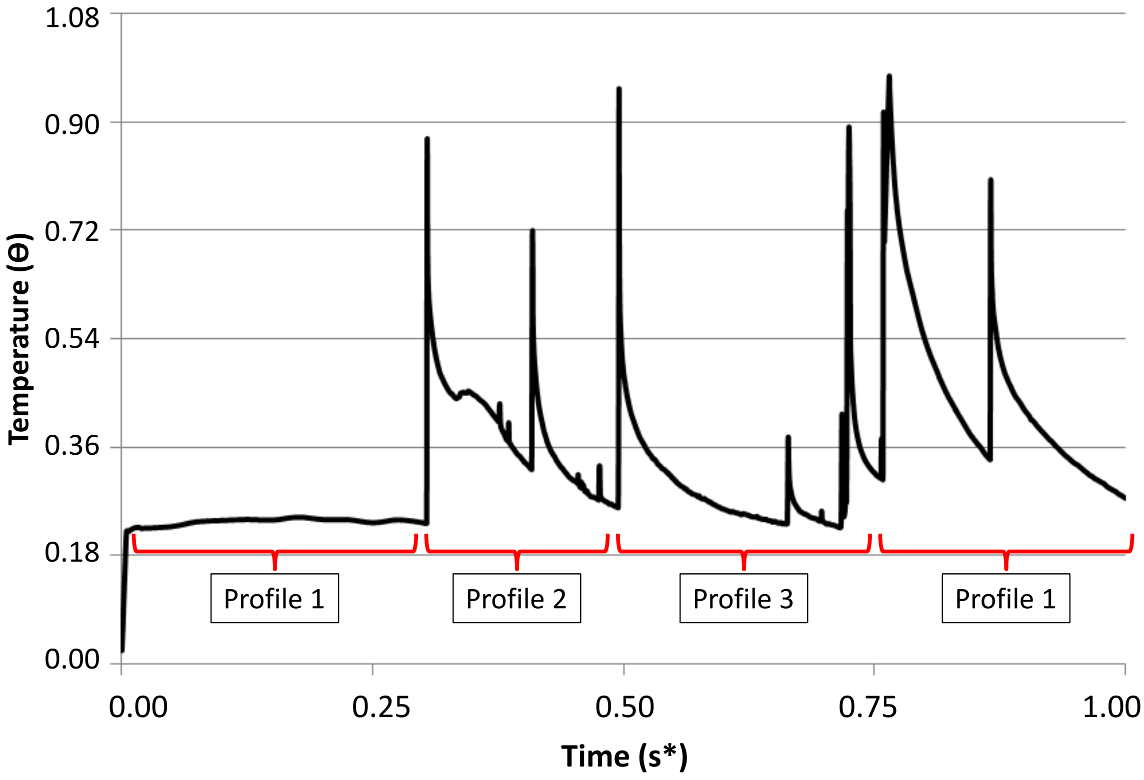


Figure 3.4: Maximum temperature along exterior of part as a function of time.

Table 3.2: Fluid and solid domain boundary conditions.

Boundary Condition Name	Domain	Type	Settings
Hot Profile	Fluid	Wall	User-defined temperature profile
Hot Profile	Solid	Wall	Temperature prescribed by fluid domain
Cool Profile	Fluid	Wall	User-defined temperature profile
Cool Profile	Solid	Wall	Temperature prescribed by fluid domain
Fixed Support	Solid	Wall	Fixed in all degrees of translation Fixed in all degrees of rotation
Support Structure	Solid	Wall	0.1 mm of free translation Fixed in all degrees of rotation

3.5.2. FEA Numerical Methods

The solid domain was evaluated using finite element modeling with ANSYS Mechanical and ANSYS Thermal, which when coupled together, are capable of computing transient thermal expansion and its associated deformation, stress, strain, and forces across each node in the model. The pre-processor used to generate the geometry and mesh was ANSYS Design Modeler.

The solid mesh contains approximately 4,400 HEX20 elements; it was determined from a mesh density investigation that a solid mesh, six elements thick, was sufficient for accurate results. The material used in the experiment is a nickel-based steel alloy, a heat-resistant stainless steel with high strength, oxidation resistivity, and minimal thermal expansion [81]. The modulus of elasticity, thermal conductivity, specific heat, and coefficient of thermal expansion was each characterized as a function of temperature, Table 3.3. Strain-life properties are not available for the nickel-based steel alloy, so the strain-life properties of the closest available material—stainless steel, Table 3.4—were used.

Table 3.3: Material properties used for computational modeling taken from a nickel-based steel alloy [81].

Temperature (K)	294	366	478	589	700	811	922	1033	1144	1255
Modulus of Elasticity (GPa)	199.9	-	184.8	-	168.2	-	149.6	139.3	128.9	121.3
Thermal Conductivity (W/m*K)	14.50	-	17.48	-	20.25	-	22.50	24.23	-	28.73
Specific Heat (J/kg*K)	440	-	490	-	544	-	595	624	-	687
Coefficient of thermal expansion (1/K * 10^-6)	-	16.31	16.81	17.26	17.66	17.95	18.25	18.5	18.9	19.4

Temperature (K)	323	373	473	573	673	773	873	973	1073	1123	1173
0.2% Yield Strength (MPa)	304.7	271.0	222.0	202.0	200.6	173.1	166.9	158.6	148.2	100.7	80.0

Table 3.4: Stainless steel fatigue life properties [82].

density (kg/m3)	7805
b	-0.139°
c	-0.415°
ε_f'	0.174°
σ_f' (GPa)	1.267°
K' (GPa)	2.275°
n'	0.334°

Due to cyclic loading and material yielding, material hardening was accounted for by applying a kinematic material hardening definition, Figure 3.5. This material property definition applies the stable hysteresis material properties after yielding occurs rather than the cyclic behavior from each cyclic load, Figure 3.6 [4,83,82]. Due to the yielding and resulting material hardening, the model must evaluate two full cycles in order to reach the long-term steady-state stress and strain fields. The first cycle generates the yielding and localized material hardening and the second cycle produces the stable hysteresis loop over which the fatigue life will be analyzed. If isentropic material hardening properties are used rather than kinematic material hardening properties, steady-state behavior is not observed, and if steady-state behavior is not observed, a cycle cannot be defined over which damage can be counted to generate fatigue life. Thus, steady-state behavior was observed by evaluating a model for four cycles. Material hardening data was not available for the nickel-based steel alloy, so the material hardening and plasticity properties of stainless steel were used, Table 3.4. Isentropic material hardening is important if the system experiences a relatively short number of cycles before failure, but the system evaluated here will experience tens of thousands of cycles before failure, thus use of the kinematic material hardening properties was appropriate. A complete list of the solid domain boundary conditions are presented in Table 3.2.

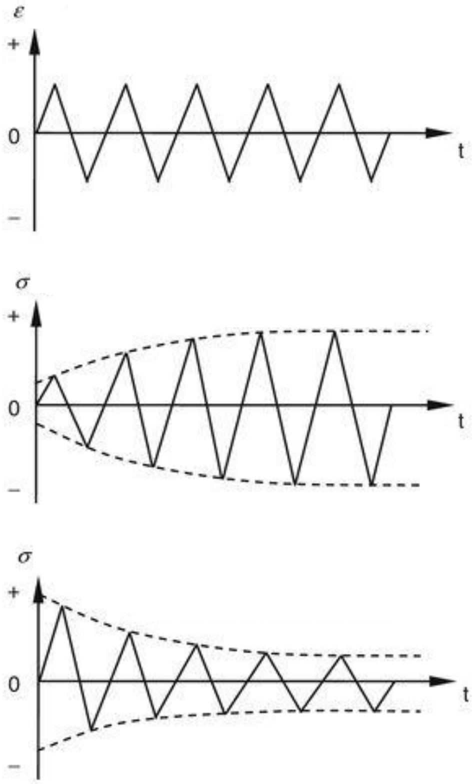


Figure 3.5: Transient material properties behavior for no cyclic material properties changing (top), cyclic hardening (middle), and cyclic softening [83].

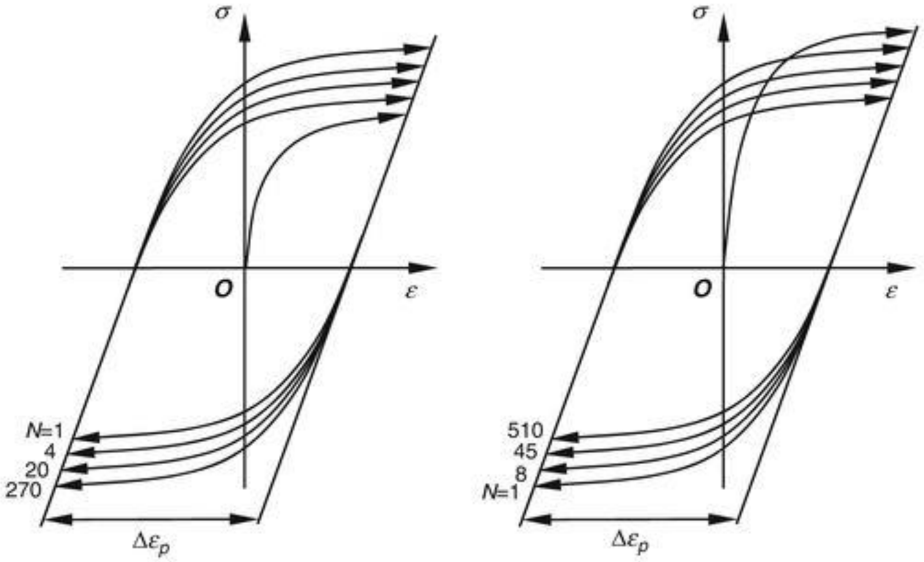


Figure 3.6: Stress–strain curve for cyclic loading until a stable hysteresis loop is reached, resulting in steady-state material properties [83].

3.5.3. Fatigue Life Numerical Methods

To determine the number of thermal cycles the part could endure, a fatigue life analysis was performed using a strain-life approach. The number of cycles the part could endure before failure was determined by the inverse of the damage from a single cycle. The strain-life approach was used because large strain values and yielding were present in the part. Mean stress was present in the part due to the yielding and long periods of elevated temperatures and can be accounted for using methods like Morrow and Smith-Watson-Topper. The Smith-Watson-Topper method, Equation 13, was utilized because it accounts for both compressive and tensile stresses, while Morrow only accounts for tensile residual stress, thus using the Morrow method only could result in an over prediction of fatigue life.

$$\boldsymbol{\varepsilon_a = \frac{(\sigma'_f)^2}{E\cdot\sigma_{max}}(N)^{2b} + \frac{\sigma'_f\cdot\varepsilon'_f}{\sigma_{max}}(N)^{b+c}} \quad \textbf{(13)}$$

Where ε_a is strain amplitude, E is modulus of elasticity, σ_{max} is the maximum stress in the model over the entire cycle, N is number of cycles, σ'_f, ε'_f, b, and c are experimentally found constants that are material-specific and generally always tested at room temperature. Due to the limited availability of strain-life constants at elevated temperatures, all constants used were measured at room temperature except for the modulus of elasticity.

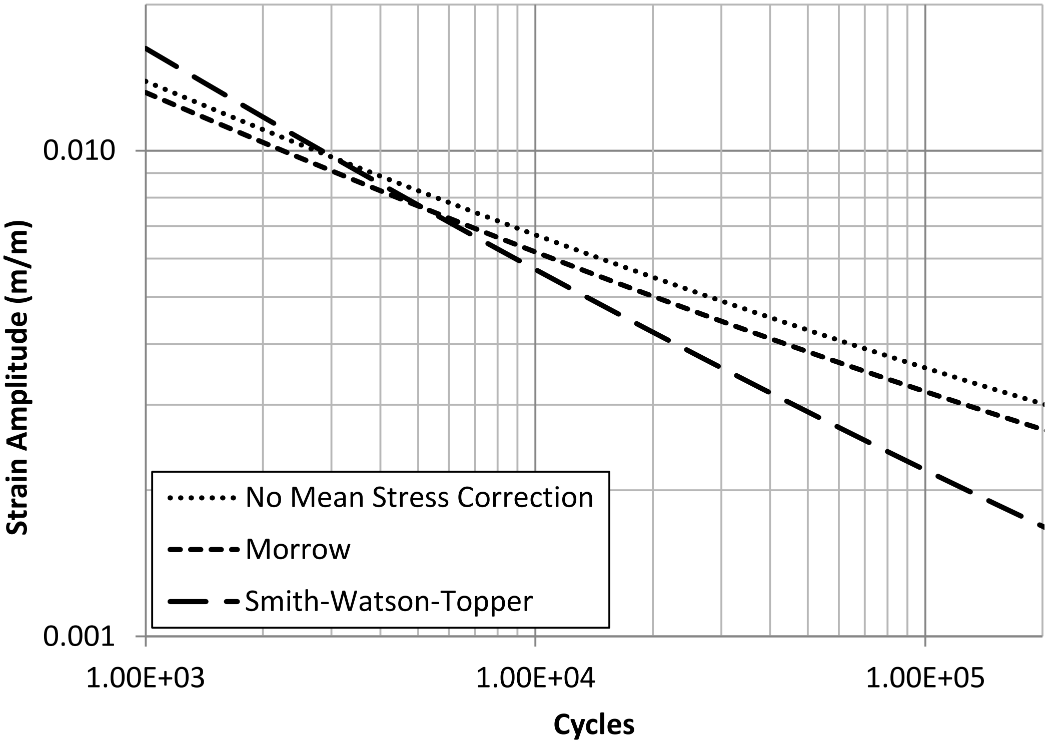


Figure 3.7: Strain-life curves for fatigue life prediction using no mean stress correction, Morrow, and Smith-Watson-Topper mean stress correction factors.

A cycle was quantified as a single 1.0 s* time span defined by the applied thermal boundary conditions. Because yielding was present, the cycle was evaluated twice and the fatigue life was analyzed based off the second cycle. To account for the effects of elevated temperatures, temperature-dependent material properties were not used in the fatigue life analysis; rather, the more conservative or worst case was used—the properties of stainless steel. Temperature-dependent material properties were used in the characterization of thermal expansion to produce the stress and strain fields. Analysis of the 1.0 s* cycle was broken into blocks so that a rainflow analysis could be performed and damage assessed for each cycle. Once the damage for one cycle was known, it was possible to compute the total number of cycles before failure. The fatigue life analysis was performed manually in Excel and again in an automated manner using ANSYS to ensure the rainflow analysis was properly performed accurately.

3.5.4. Interface between Fluid and Solid Domain

The pre-processor used for generating the geometry and mesh for both the fluid and solid domain was ANSYS Design Modeler. This commercially available software package has features that allow the fluid and solid domains to be created simultaneously and then separated for individual meshing and analysis techniques. This allows the faces to be shared for a more seamless data transfer and for interface compliance in the FSI coupling.

ANSYS System Coupling was used to transfer data from the fluid domain to the solid domain in a weak-coupled manner at each time step. Data was transferred one way because the changes in the solid model due to thermal expansion have negligible effects on the temperature profile applied by the fluid domain. A relaxation factor of 1.0 was used for transferring data between each domain; i.e. no ramping function or damping was introduced. The fluid domain provided temperature loads to the solid domain for thermal expansion. The FSI model utilized a 0.005 second time step over two 1.0 s* cycles.

3.5.5. Parametric Study Methods

Once the computational model base condition was completed, a parametric study was performed to determine what factors had the greatest effect on fatigue life. This parametric study and resulting discussion are intended to look at which designs perform better or worse relative to the base conditions—the exact number of cycles is not the focus. In addition to the base condition, nine other designs were evaluated, including cases that looked at geometry setup, applied temperature conditions, material properties, and a three-dimensional model with a temperature hot spot, Table 3.5. The thermal cap design with wall thickness of 1.5 mm and cap thickness of 5.0 mm is presented in Figure 3.8. The temperature contour for the three-dimensional temperature hot spot is presented in Figure 3.12. The modified thermal boundary conditions were achieved by using the existing temperature profiles, but altering the scaling factors used, either up 100K or down 50K, to match the maximum temperature values at each time step. In reality, it is difficult to alter a single material property without affecting other properties, so to minimize these consequences in this study only one property was altered at a time. The coefficient of thermal expansion was reduced from 16.31E-6 1/K to 5.00E-6 1/K, which corresponds to the coefficient of thermal expansion for the nickel-based steel alloy and tungsten, respectively [81,84]. The yield strength of the material was increased from

304.7 MPa to 600.0 MPa, which corresponds to the yield stress of the nickel-based steel alloy and high-end steel alloy respectively [75,81,82].

Table 3.5: Parametric study properties setup.

		Thickness (mm)	Temperature Profile	Coefficient of Thermal Expansion (1.0E-6/K)	Yield Strength (MPa)
Base Condition		1.5	Standard Profile	16.31 at 366K	304.7 at 323K
Geometry	1 mm thickness	1.0	Standard Profile	16.31 at 366K	304.7 at 323K
	2 mm thickness	2.0	Standard Profile	16.31 at 366K	304.7 at 323K
	Thermal Cap	1.5 wall thickness 5.0 cap thickness	Standard Profile	16.31 at 366K	304.7 at 323K
Temperature	50 K cooler	1.5	50K Cooler	16.31 at 366K	304.7 at 323K
	100 K warmer	1.5	100K Warmer	16.31 at 366K	304.7 at 323K
Material Properties	Reduced CTE	1.5	Standard Profile	5.00 at constant	304.7 at 323K
	Increased yield strength	1.5	Standard Profile	16.31 at 366K	600.0 at constant
	Tungsten	1.5	Standard Profile	5.00 at constant	900.0 at constant
Three-dimensional hot spot		1.5	Standard Profile	16.31 at 366K	304.7 at 323K

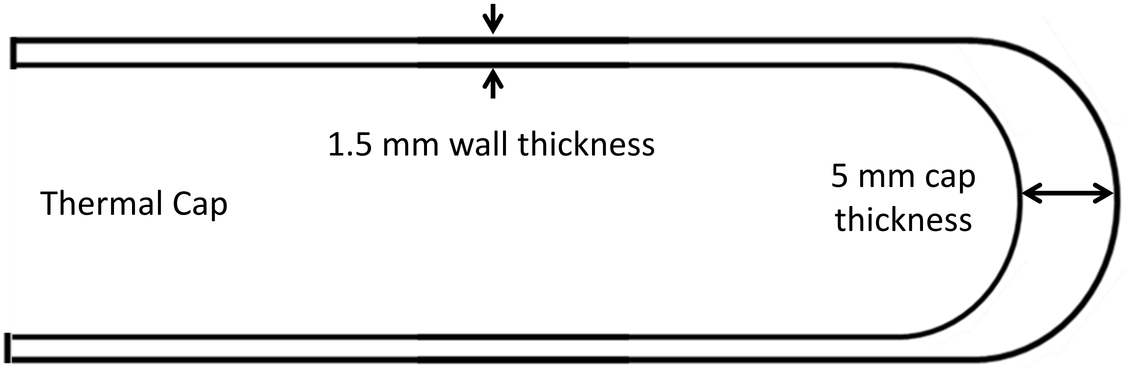


Figure 3.8: Design layout for the thermal cap geometry.

3.6. Results

3.6.1. Steady-State Material Properties

Initial evaluation and inspection of the strain for the first four cycles indicated the steady-state results were not reached using isotropic material hardening, Figure 3.9 top. Upon redefining the material properties using kinematic material hardening, the steady-state strain fields occurred on the second cycle, Figure 3.9 bottom.

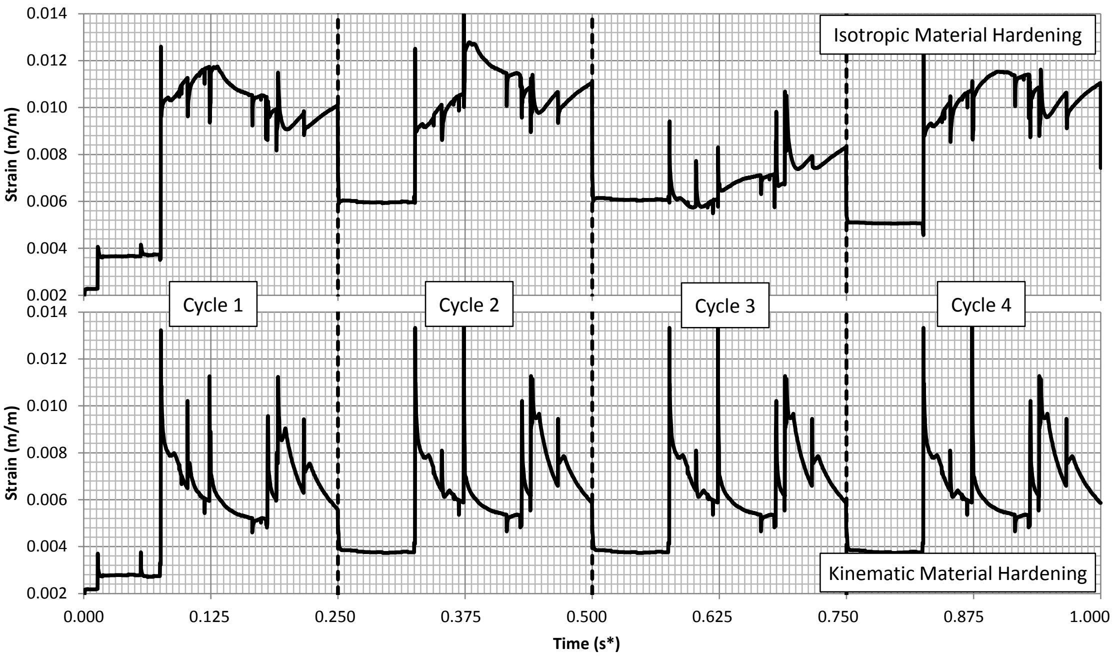


Figure 3.9: Comparison of material properties definitions for material hardening over four, 1.0 s* cycles.

3.6.2. Computational and Experimental Comparison

Experimental data is not available for a quantitative validation of the computational model, but experimental observations, trends, and failure modes are available. Additionally, the computational lifetime predictions from the fatigue life model cannot be used as a validation point because the CFD model used to apply the thermal boundary conditions is believed to be a worst-case condition and not the mean operating condition.

3.6.2.1. Yielding Location

The experimentally fatigued parts show residual tensile stress on the exterior surface of the parts. If the part was heated uniformly, no yielding would be present. However, if the exterior of the part is heated and the interior cooled, yielding can occur along the interior surface due to the large thermal expansion of the exterior. If the part is then allowed to cool to a uniform temperature, residual tensile stress will be present along the exterior surface due to permanently yielded (expanded) material along the inside surface. Experimentally, this residual tensile stress is present along the exterior of the part and is present in the computational results as well, Figure 3.10. This does not validate the computational model, but supports the conclusion that the model is capturing the proper yielding mechanisms due to thermal expansion.

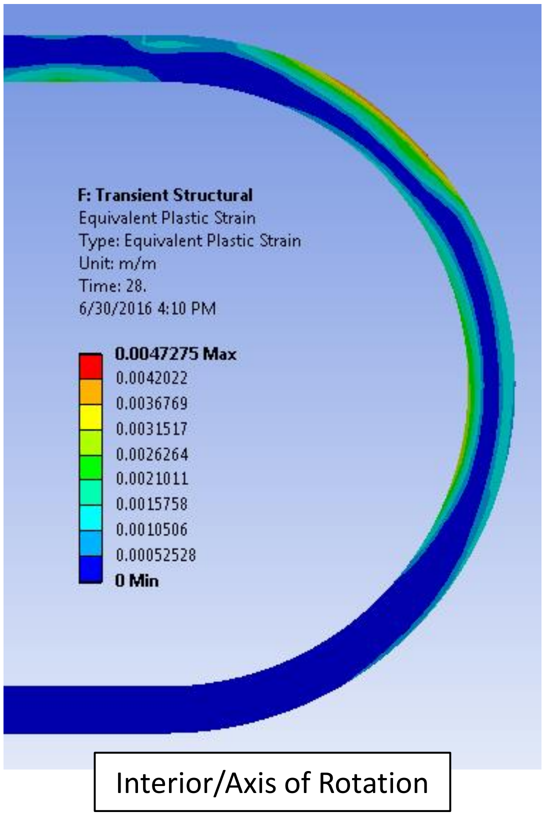


Figure 3.10: Contour plot of plastic strain indicating strain is positive along the exterior of the part in agreement with experimental data.

3.6.2.2. Failure Location

The failure location in the experimental data agreed with the computational data as occurring on the outside edge of the part halfway between the tip and the side wall, Figure 3.11 between the lines. This failure region is not centered in the area of highest observed temperature, but is one of the regions where the highest temperature gradient is present. The part experiences large regions with very high temperature where thermal expansion is significant. Large temperature changes alone are not capable of causing thermal stresses. A part that is unconstrained in space and experiences uniformly high temperatures experiences no stress due to thermal expansion. However, if a portion of the part is fixed or the entire part is not uniformly heated, thermal stresses can be significant. This part experiences large temperature spikes over short periods of time, but these spikes are not present over the entire part, which results in sharp temperature gradients in both space and time. These steep temperature gradients contribute to the thermal stress fields. Therefore, it would not be expected for this part to fail at the center of the high temperature location, but instead to fail where the steepest thermal gradients are present. In this setup, the steepest thermal gradients are present around the edge of the temperature hot spot, Figure 3.11 (left). Figure 3.11 (right) shows the predictive failure region of the part is in the area of steepest temperature gradient, demonstrating agreement with experimental failures.

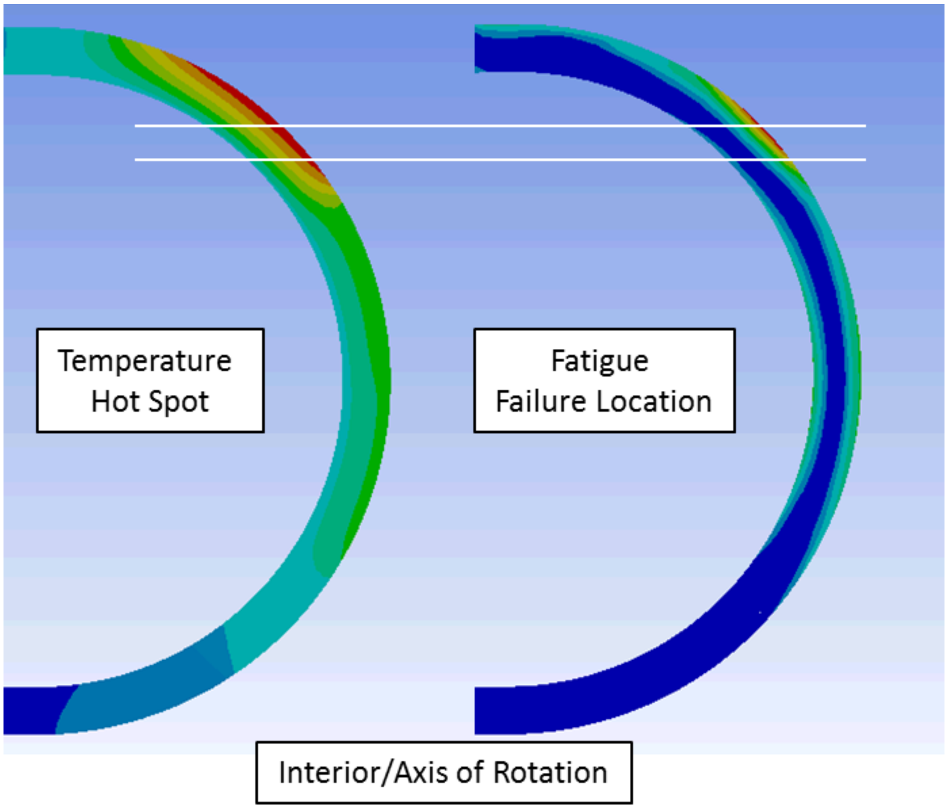


Figure 3.11: The fatigue failure location (right) lies on the edge of the temperature hot spot (left) where the temperature gradient is the greatest, as indicated by the white lines.

3.6.2.3. Failure Modes

The computational base condition assumed the surface finish of the part was void of flaws (polished), but the experimental base condition contained two variations with machined-surface and polished-surface finish. The tool markings in the machined surface part are indicated in Figure 3.12 and are in the hoop direction. In the experimental case of the machined surface part, cracks initiate and grow parallel to the tool markings in the hoop direction, while the part with the polished surface showed cracks growing radially, Figure 3.12. The parts with polished-surface finish also experience a longer service life than the machined surface part. The presence of tool markings creates stress concentrations that initiate crack growth. Therefore, it is not surprising that the polished-surface finish outperformed the machined-surface finish. Because this change in the failure mode was so distinctive in the experiments, it is hypothesized that the hoop forces and radial forces are on the same order of magnitude. The principal stresses in the computational model indicated the maximum principal stress is in the hoop direction, which would lead to radial cracking, Figure 3.13. However, the second largest principal stress is on the same order of magnitude and in the radial direction, which would lead to hoop cracking. This supports the cracks forming in the hoop direction when stress concentrations are present, while a longer service life occurs when cracks form in the radial direction with a polished-surface finish.

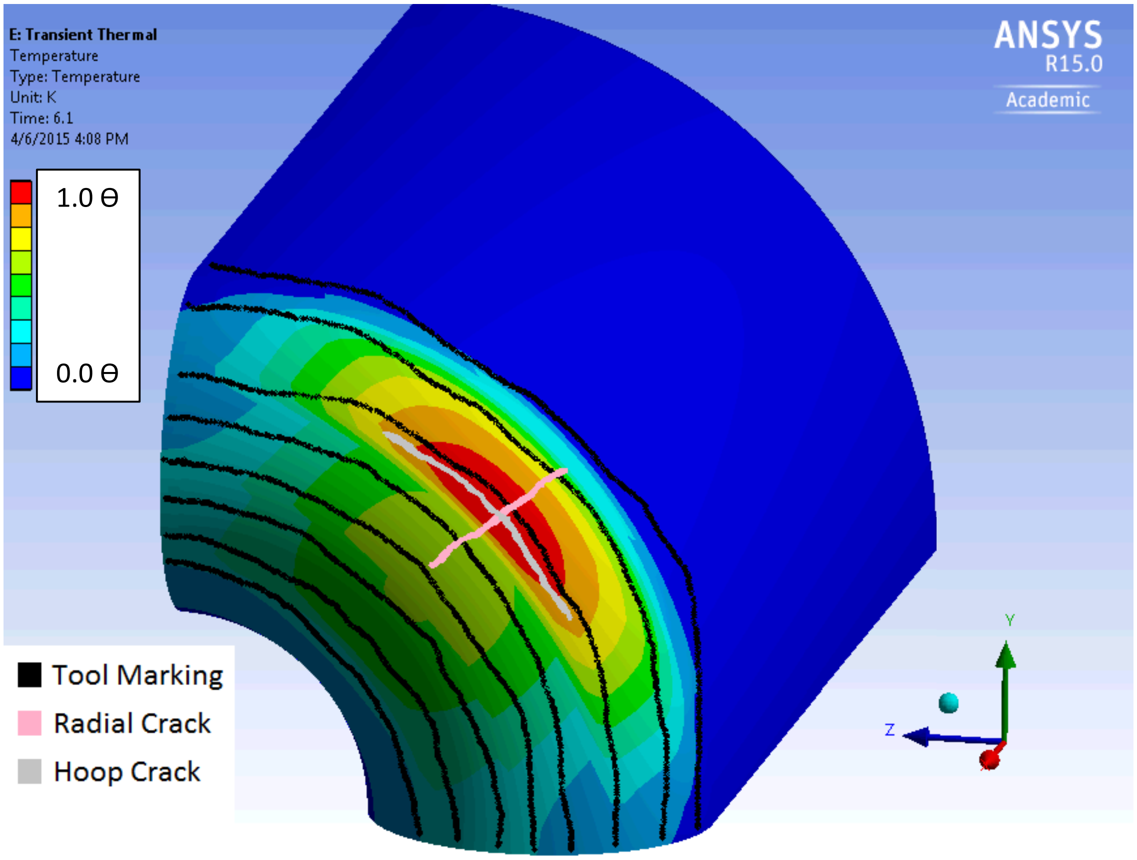


Figure 3.12: Contour lines of temperature with overlaid tool markings, radial crack, and hoop crack.

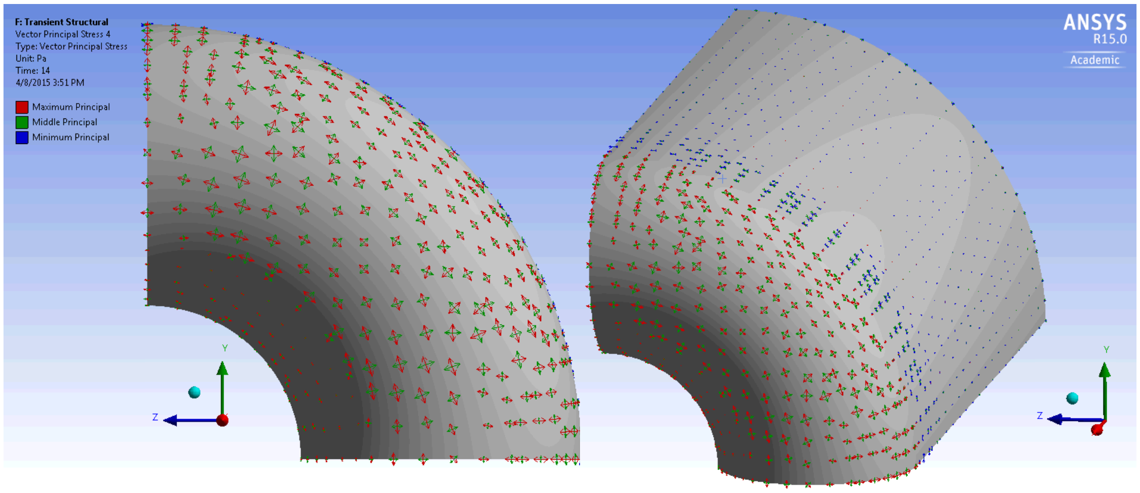


Figure 3.13: Principal stresses along the exterior of the part show the maximum and middle principal stresses are on the same order of magnitude, indicating the potential for cracks to form both radially and in the hoop direction.

3.6.3. Strain-Life Analysis

The fatigue life of the part was determined from the strain in the part (Figure 3.9) and the strain-life curve generated from the material properties (Figure 3.14). The damage from each load reversal in a single 1.0 s* cycle was summed to determine the damage imparted by a single cycle. Using the cumulative damage of each cycle, the total number of cycles until failure was calculated, Table 3.6. The base condition lasted 2.9 days and was used as a baseline for determining whether design changes improved or reduced the life of the part before failure occurred.

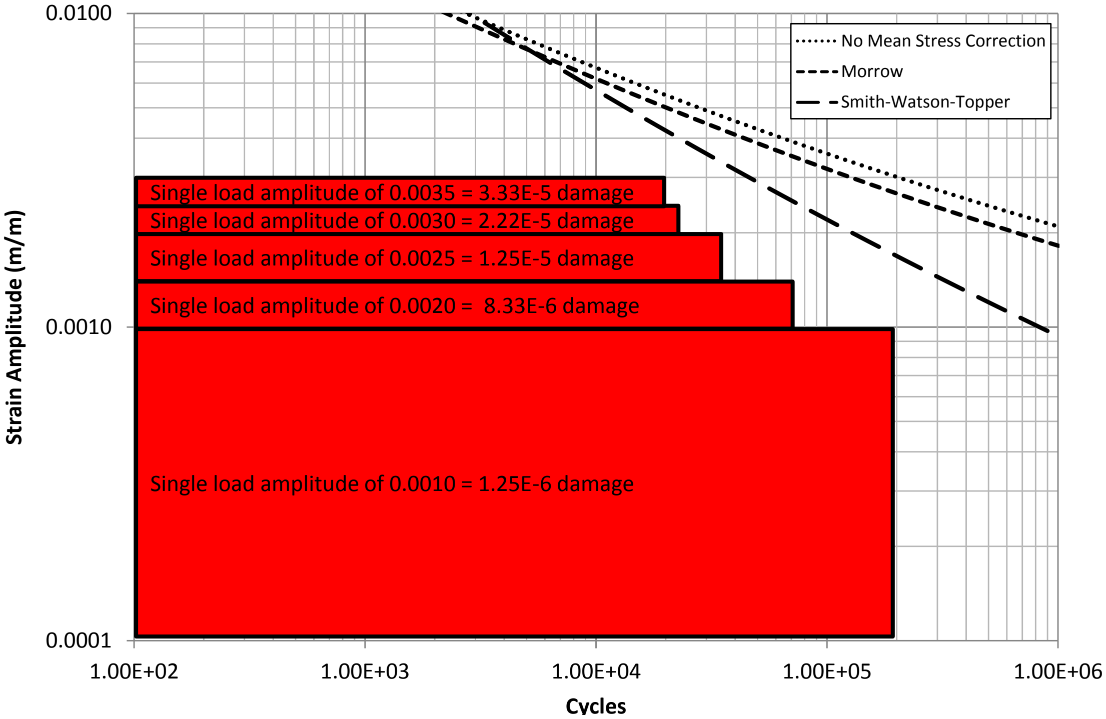


Figure 3.14: Damage produced from a single load amplitude of 0.001, 0.002, 0.0025, 0.003, and 0.0035 (m/m). With the known damage contribution of each amplitude, the damage of a single cycle can be calculated. With the damage per cycle known, the number of cycles until failure can be determined.

Table 3.6: Results from parametric study for alternative geometry changes, applied thermal conditions, material properties, and three-dimenstional temperature hot spots.

		Life in days
Base Conditions		1.00
Material Properties	Reduced CTE	4.98
	Increased yield strength	1.28
	Tungsten	7.01
Temperature	50 K cooler	1.11
	100 K warmer	0.74
Geometry	1 mm thickness	3.48
	2 mm thickness	0.40
	Thermal Cap	0.65
3 Dimensional		1.36

3.6.4. Parametric Study

After the base condition was completed, the parametric study determined which designs had a longer fatigue life. The geometry, material properties, and thermal boundary conditions all had an effect of the part fatigue life, Table 3.6. The computational models only perturbed a single variable at a time and left the other variables at the base condition.

3.6.4.1. Geometry Alteration

Three different geometric configurations were evaluated to determine the effects of geometric alterations to the fatigue life of the part. Reducing the thickness of the part from 1.5 mm to 1.0 mm increased the life from 1.00 days to 3.48 days, while increasing the thickness from 1.5 mm to 2.0 mm reduced the life from 1.00 days to 0.4 days. It was anticipated that added thermal mass near the hot spot by increasing the thickness of the part would broaden the temperature distribution resulting in shallower temperature gradients. This theory was tested with the thermal cap, but instead resulted in a reduction of life from 1.00 days to 0.65 days. In addition to computational modeling, each of the three geometric configurations were experimentally tested and matched with the trends seen in the computational results. By reducing the thickness of the part from 1.5 mm to 1.0 mm the Biot number within the metal is lowered, therefore reducing the thermal gradients resulting in decreased stress and strain, which increases the life.

3.6.4.2. Thermal Boundary Conditions

Two different thermal boundary conditions were evaluated to determine the sensitivity the thermal boundary conditions had with respect to fatigue life. Reducing the maximum temperature the part experienced by 50K compared to the base condition temperature resulted in an increased life from 1.00 days for the base condition to 1.11 days. This increase in life is believed to not be directly related to the applied temperature field, but more a result of the temperature-dependent material properties. Increasing the overall temperature by 100K reduced the overall life compared to the base condition from 1.00 days to 0.74 days.

3.6.4.3. Material Properties

Three different material conditions were evaluated to determine the effects material properties have on the lifetime of the part. Reducing the coefficient of thermal expansion resulted in significant reductions in stress magnitude and subsequently increased the life from 1.00 days to 4.98 days. This result was expected because the stress in this part is generated from thermal expansion and not from external loads. Increasing the yield strength of the material resulted in reduced yielding and stress, which subsequently increased the lifetime of the part from 1.00 days 1.28 days. Using tungsten increased the life from 1.00 days to 7.01 days.

3.6.4.4. Summary of Parametric Study

It was evident from the parametric study that the computational model is robust enough to capture the effects of changes in material properties, thermal boundary conditions, and geometry. Although the computational model can accurately capture the desired changes, the effect of the coefficient of thermal expansion was the most significant factor with respect to lifetime. It was therefore unnecessary to perform an optimization because the desired outcome was clear. It was desired to have a material with a low coefficient of thermal expansion and high yield strength and thermal boundary conditions with more uniform temperature profiles and the lowest maximum temperatures. The ideal geometry solutions are harder to predict because any changes in geometry will result in an altered thermal boundary condition, which is held constant under these conditions.

3.7. Conclusion

This qualitative validation and parametric study shows that an FSI model can be used as a tool to design a part for optimal lifetime performance. The analysis methods in this study are robust enough to capture thermal effects, material properties, and geometric alterations. These computational models were set up and evaluated using commercially available software and inexpensive computational resources. This study demonstrates that FSI modeling can be used as a design tool in an industry-applicable environment.

3.8. Lessons Learned

The methods described in this chapter lay the foundation for how thermal fatigue FSI models should be constructed, evaluated, and validated. Generating an accurate fatigue life prediction from the results of an FSI model requires a detailed understanding of the fluid model, solid model, and material properties used.

The introduction of elevated temperatures can cause changes in material properties, specifically altered material elasticity, yield stress, material plasticity, and fatigue life. The material properties of exotic materials may not be known, so the properties from known materials may need to be substituted. If substitutions are performed, it is important to understand what effects these substitutions will have on results and fatigue life predictions.

Accurately implementing material properties such as material hardening and yielding is a critical step to produce the most accurate answer. If accurate steady-state material properties and the resulting stress-strain fields cannot be achieved, a failure analysis will not yield accurate results.

The validation of FSI models can be challenging due to the limited data available for qualitative comparison. If this is the case, experimental observation and trends should be used to ensure fluid and solid mechanisms are accurately captured in the model.

CHAPTER 4: LIFETIME DESIGN FOR THERMAL-FLUID SYSTEM

4.1. Abstract

The modeling of thermal-fluid systems has become more available due to the increased computational power of modern computers and stability of algorithms employed. However, limited literature exists regarding the lifetime analysis and optimization of a thermal-fluid system. The objective of this work is to improve the current design of a passive thermal-fluid control system by maximizing the duration of applied external thermal loads, while minimizing the mass of a phase change material within the system. A computational model was used as a design tool to achieve optimal thermal life. The objective of the optimization was to maintain a uniform temperature distribution inside the vessel while staying within a desired temperature range and using the least amount of phase change material possible. The geometry consisted of a three-dimensional model with no symmetry and the model captured phase change, free convection, and radiation. Experimental data was used to tune the material properties in the model and validate the results. The detailed material properties testing was conducted at the University of Denver, while our industry collaborator conducted the experimental tests that required access to a thermal chamber.

4.2. Introduction

Thermal energy management is a significant concern in diverse industries such as spacecraft and satellite design, high-speed aircraft, electronics, medical organs and therapeutic protein transport, food transportation and storage, and wine storage [52,85–89]. In these varied industries, radiation, convection, and conduction each play an integral part in heat transfer that must be addressed to ensure designs, products, and systems do not fail as a function of exceeded temperature limits. These industries spend time and resources to manage thermal energy by means of open- and closed-loop control systems, active and passive systems, geometry, and material properties [52,85]. The resulting systems designed to manage thermal energy can be large, heavy, and bulky; for example, refrigeration units designed for food transport and storage. Other systems may be simple, requiring only insulation and ice blocks or specific properties, geometry features and/or fins [52,85]. Regardless of the method or complexity of the thermal management system, the goal is to manage the thermal energy so failure does not occur.

4.3. Problem Description

This study investigates a passive thermal-fluid system subjected to an external thermal load on all exterior surfaces. The system is designed as a passive cooling system using only ice blocks to maintain the internal contents of the vessel within a specified temperature range for a minimum of 60 hours. If the temperature is not maintained inside this range, the perishable contest of the vessel will become unusable. Prior to design optimization at the University of Denver, the vessel maintained the internal contents within the desired temperature range for only 40 hours. The goal of this study was to determine the minimal amount of ice required to maintain the temperature inside the vessel for 60 hours. Through this study, the number of ice blocks used and location of the ice blocks was altered to determine the configuration resulting in the smallest amount of ice necessary. This study used a computational model constructed in ANSYS Fluent to determine the performance of each configuration. The computational model was tuned using experimental test results conducted at the University of Denver and validated using experimental test results conducted by our industry collaborator.

4.3.1. Problem Normalization

The setup and results for this study were normalized as requested by our industry collaborator. Table 4.1 outlines the methods used to normalize the setup and results, where T represents the localized temperature, T_0 is the minimum temperature, T_∞ is the maximum temperature, l is length, l_{base} is the length of the vessel, m is mass, m_{base} is the mass of base condition, h is the convective heat transfer coefficient, K is thermal conductivity, ρ is density, and C_p is specific heat.

Table 4.1: Normalization units and equations for presented data.

Quantity	Modeling Units	Normalization Method	Units Present
Temperature	Kelvin	$\theta = \frac{\mathrm{T} - T_0}{T_0 - T_\infty}$	ϴ
Length	Meter	$m^* = \frac{l}{l_{base}}$	m*
Mass	Kilogram	$kg^* = \frac{m}{m_{base}}$	kg*
Time	Second	$s^* = \frac{t}{t_\infty}$	s*
Thermal Conductivity	W/m-K	$\text{Nusselt Number} = \frac{h * l}{K}$	Nu
Specific Heat	J/kg-K	$\text{Thermal Diffusivity} = \frac{K}{\rho * C_p}$	α

4.3.2. Thermal System Failure

This system is a closed thermal system where the temperature inside the vessel is governed by the initial thermal energy inside the system, insulation of the system, and applied boundary conditions. Failure of this system occurs when the temperature at any of the six measured locations is not with -0.192 to 0.397 ϴ for any length of time. The temperature measurements were collected in drawers 1, 3, and 5 with two locations measured per drawer represented by the *X*'s in Figure 4.5. The temperature measurements were taken in the same locations in both the computational model and experimental tests.

4.3.3. Design Constraints

The design requirements provided by our industry collaborator created several design limitations: (1) the cooling fluid inside the ice blocks must remain unchanged, (2) the ice blocks used must not be made smaller, (3) the dimensions of the vessel, drawers, and containers must remain the same, and (4) the flat ice blocks located on the top and bottom of the vessel cannot be changed in any way. The alterations to the closed thermal system allowed by our industry collaborator were to replace perishable items with more ice blocks, alter the number of ice blocks per drawer, and alter the spatial arrangement of the ice blocks in each drawer.

4.4. Methods

Preliminary experimental tests were required before the boundary conditions and material properties could be understood and implemented into the computational model. Once initial experimentation was completed, experimental tests and construction of the increasingly complex computational model were conducted in parallel. Experimental tests were conducted to provide a rough estimate of material properties before these properties were fine-tuned using the computational model and measured experimental results. Testing and tuning of material properties and boundary conditions was conducted in the following four steps:

1. Thermal Conductivity Experiment. Performed experimental tests to determine the range for composite thermal conductivity of the vessel.
2. Composite Thermal Conductivity Tuning. Tune thermal conductivity of the vessel using the empty vessel experimental tests and computational model.
3. Gel-Specific Heat Tuning. Tune the specific heat of gel using the computational model and the experimental tests with gel and no ice.
4. Boundary Conditions Tuning. Tune the boundary conditions applied to the outside surface of the vessel using experimental tests and computational model.

4.4.1. Experimental Methods

4.4.1.1. Thermal Conductivity Experiment

The performance of this system was driven by the initial internal energy, boundary conditions, and insulations (material properties) of the vessel layers. The vessel layers consisted of a paper product, two different thermoplastic liners, and insulating foam, and the layers were constructed in such a way to promote maximum thermal resistance between the ambient conditions and cool interior, Figure 4.1. The material properties of the foam can vary greatly as a function of foam cell density, gas inside the foam cells, and operating temperature of the vessel. Polyurethane foam is a commonly used insolating foam. Table 4.2 shows the possible range of thermal conductivities available for variations of polyurethane foam. The effect of placing the insulation layers next to each other also adds thermal contact resistance between the insulation layers. A closer inspection of a cross section of the foam revealed large voids and inconsistencies in foam pore size and density, Figure 4.2. Due to the unknown gas composition inside the potentially closed-cell foam, material inconsistencies in foam cell size, and contact resistance between layers, it was important to model the composite thermal conductivity of all the layers together as a single thermal resistance. An experimental test was set up and conducted at the University of Denver to determine the possible range for the composite thermal conductivity.

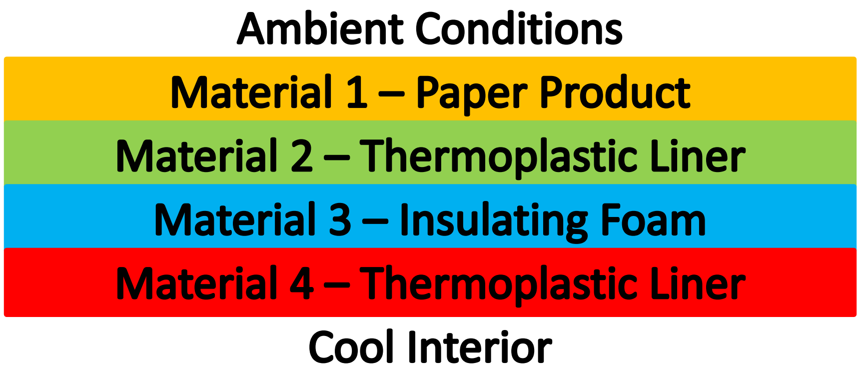


Figure 4.1: Layers inside the vessel wall to promote maximum thermal resistance between ambient conditions and the cool interior.

Table 4.2: Avalible material properties for polyurethane insulation and air [90–96].

Name	Data Source	Thermal Conductivity (W/m-K)		
		Mean	Min	Max
Polyurethane Insulation	Engineering ToolBox	0.03		
Micro-Cellular Polyurethane Foam	MatWeb	0.0865		
Polyurethane Foam - Unreinforced	MatWeb	0.129	0.076	0.83
Polyurethane	University Physics 7th Edition	0.02		
Common Insulation Material (Polyurethane)	Federation of the European Rigid Polyurethane Foam Associations		0.02	0.05
Polyurethane Foam	10th International Symposium on District Heating and Cooling	~0.024		
Air as a function of temperature (0-60°C)	Heat and Mass Transfer 7th Edition	0.0264	0.0243	0.0285

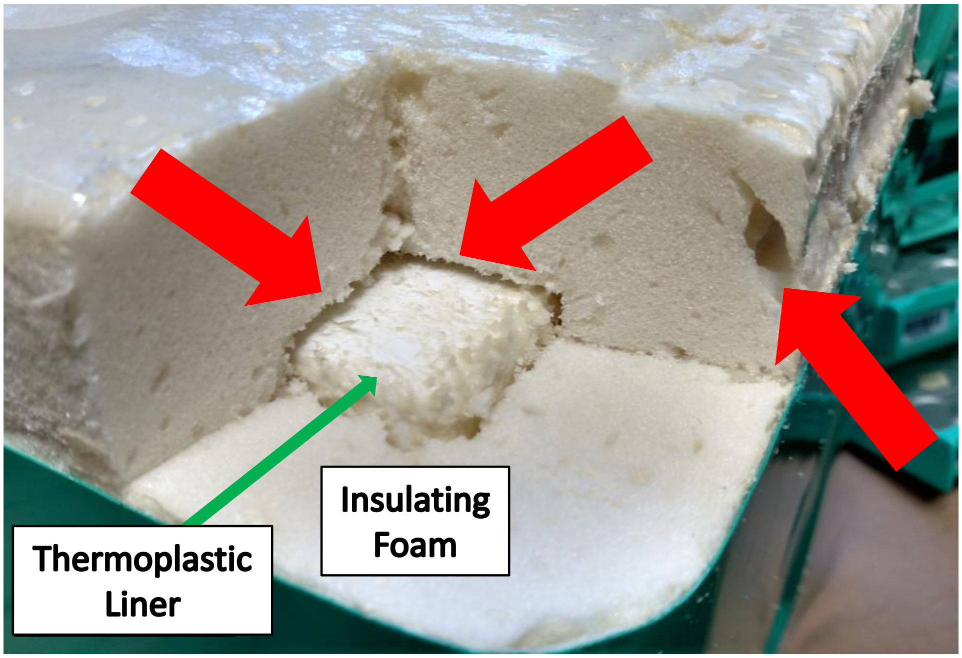


Figure 4.2: Cut cross section of the insulating foam interior showing the thermoplastic liner and material voids in the bulk of the insulating foam.

The thermal conductivity can be found by rearranging the basic conduction heat transfer equation, Equation 14, for a given material or a composite group of materials. All of the values on the right side of Equation 14 can be measured experimentally, but several assumptions must be made.

$$\boldsymbol{K} = \frac{\boldsymbol{Q}*\boldsymbol{L}}{\boldsymbol{A}*(\Delta \boldsymbol{T}_{\mathbf{1}}+\Delta \boldsymbol{T}_{\mathbf{1}})} \tag{14}$$

The experimental test performed to generate the composite thermal conductivity of the vessel and door was conducted by taking a constant four watt heater and placing it on a metal plate that fit inside the seal of the vessel cover. An additional Styrofoam seal was added to create a small gap between the cover and vessel to ensure the heater did not directly touch the surface of the cover. A 10 lb. load was then applied on top of the cover to ensure a consistent seal was present between the cover and vessel. Once the load was applied, the heater was turned on and allowed to run until steady-state conditions were reached, which took about 24 hours. Temperature measurements were taken along the top of the cover, between the cover and vessel, along the inside of the vessel, and on the outside of the seal, Figure 4.3. Once steady-state conditions were observed, the temperatures were recorded and the experiment was allowed to run an additional 24 hours before the measurements were retaken. If no significant difference was observed over the second 24 hours, it was determined the steady-state conditions had been reached and the measured temperature values were used to calculate the composite thermal conductivity.

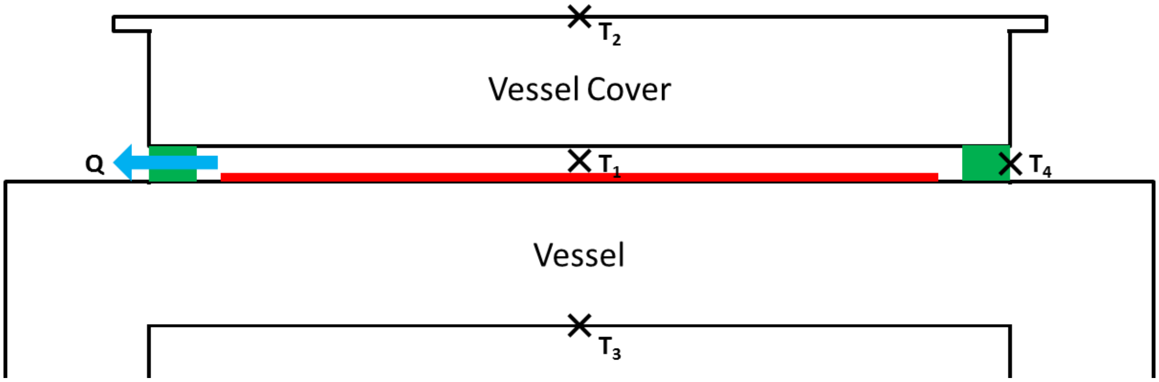


Figure 4.3: Diagram of experimental setup with location of temperature measurements and heat loss.

Due to the assumptions made during the test setup phase, there was a large range of experimental uncertainty for the measured composite thermal conductivity, but the experimental range fell within the range documented in published data. The assumptions that lead to this uncertainty were: (1) area of heat transfer, (2) temperature uniformity, (3) heat generation, and (4) heat transfer efficiencies.

(1) The area over which the heat transfer occurs has a large impact on the thermal conductivity calculation. Using the metal plate to distribute the heat over the entire area of the cover minimized the uncertainty of this experimental value. The area inside the seal is 0.871 m^{*2} and the area including the seal is 0.998 m^{*2}.

(2) The temperature uniformity along the heating surface, top of the cover, and inside of the vessel is important because the measurements are taken at a single point. If the surfaces have irregular temperature distributions, an average must be taken. By using the heating pad attached to the metal plate, the surface over which the heat is applied is uniform. Temperature uniformity was checked by taking temperature measurements at various locations along the top of the cover and inside of the vessel. The measurements from these locations never varied more than 0.066 Θ.

(3) The heat was generated using a constant four watt output heating pad. To verify the output of the heating pad, a power meter was used to measure the power, and indicated an average of 4.1 watts was used over the 48 hours. Given the slight inefficiencies inherent in the heater, 4.0 watts seemed to be an accurate value.

(4) The most significant source of uncertainty comes from the assumption of how much heat from the heating pad is actually transferred through the vessel and cover walls and how much escapes through the seal, displayed as Q in Figure 4.3. The losses through the seals were measured at 1.31 watts using Equation 15. Where K is the thermal

conductivity of Styrofoam, A is the area of the seal, ΔT is the temperature across the seal, and L is the length of the seal.

$$\boldsymbol{Q = \frac{K * A * \Delta T}{L}} \tag{15}$$

4.4.1.2. Thermal Contact at Drawer Interface

The ice and containers that hold perishable items inside the vessel each have four small round feet on the bottom, thus significantly reducing the contact area between the drawer and containers, Figure 4.4. To account for this reduced contact area and avoid modeling the small 1 mm gap, a small insulating material was modeled under the entire surface of each ice block and perishable item container. The thermal conductivity of this insulator was determined by a heat transfer relationship, Equation 16. By measuring the area of the feet and the area of the container bottoms, the resulting thermal resistance was determined. The thermal conductivity for the gap between the perishable items and drawer is 1.148 Nu, and the thermal conductivity for the gap between the ice and drawer is 0.575 Nu. The smaller thermal conductivity used for the ice and drawer occurred because the ice blocks have a smaller base area, but the same number and area of feet.

$$\boldsymbol{Q_{feet} = Q_{gap} = \frac{K * A * \Delta T}{L}} \tag{16}$$

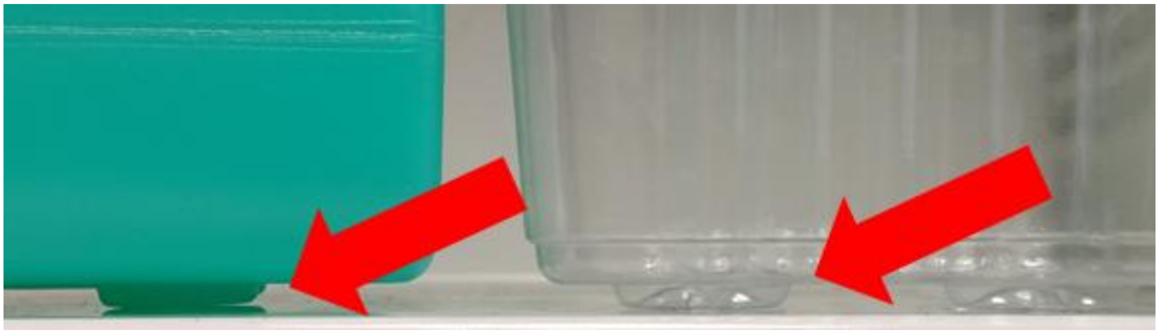


Figure 4.4: Image of feet on the ice block and perishable item containers.

4.4.1.3. Thermal Chamber Experimental Test

Transient experimental testing of the vessel was conducted by our industry collaborator using a controllable thermal chamber and ambient room conditions, while measuring the temperature inside and outside of the vessel at various time points. The thermal chamber experimental tests were performed with nine vessels in the chamber arranged in a 3x3 array and placed on wooden pallets. Multiple experiments were performed under a variety of conditions with multiple vessels under the same conditions. Two of these experiments were used for tuning the computational model and a third was used for validation. Gel was used as a substitute for the perishable items to reduce the experimental uncertainty originating from the variability in the thermal mass of the perishable items.

The first experiment contained an empty vessel with no gel and no ice. The empty vessel was placed inside a thermal chamber at 0.132 ϴ until thermal equilibrium was achieved. Then the vessel was removed and placed in a room with little air flow and ambient room temperature conditions for eight hours. The applied external temperature load and internal temperature measurement are available in Figure 4.7.

The second experiment contained a vessel with only gel and no ice. The vessel was placed inside a thermal chamber at 0.066 ϴ until thermal equilibrium was achieved. Once this equilibrium was achieved, an external temperature load was applied for 20 hours and the internal temperature was measured, Figure 4.8. The configuration of gel is depicted in Figure 4.5 (left) but without ice blocks.

The third experiment contained a vessel with gel and ice. The vessel was placed inside a thermal chamber at 0.000 ϴ until thermal equilibrium was achieved. Once this equilibrium was achieved, an external temperature profile was applied for 60 hours and the internal temperature was measured. The external temperature profile can be seen in Figure 4.9 and the configuration of the experiment is depicted in Figure 4.5 (left).

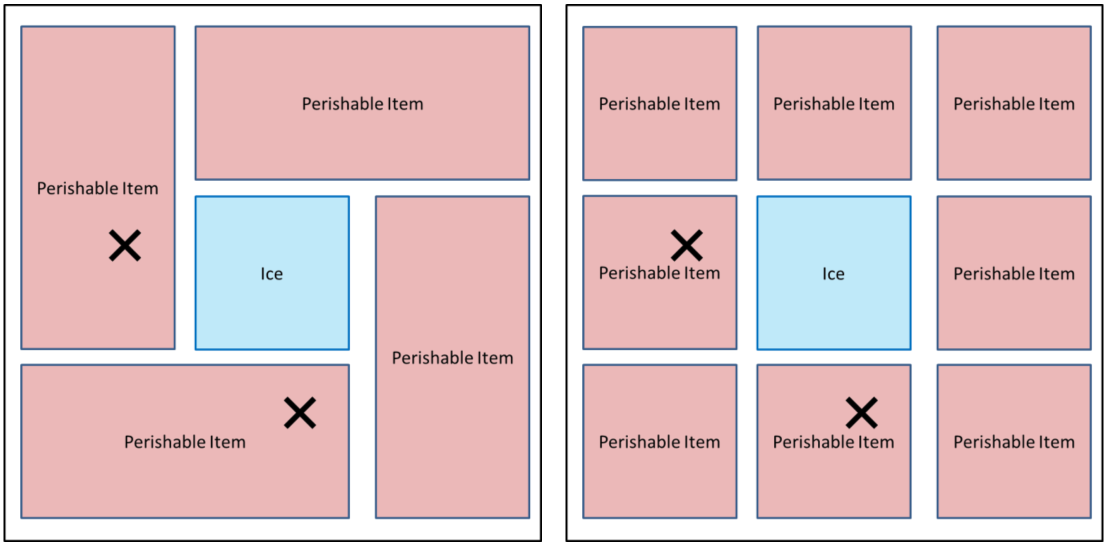


Figure 4.5: Gel and ice locations for the validation study (left) and design improvement study (right). The X represents the locations for temperature measurements both experimentally and computationally in drawers 1, 3, and 5.

4.4.2. CFD Numerical Methods

The fluid domain was evaluated using Computational Fluid Dynamics modeling in ANSYS Fluent 15. The pre-processor used for mesh generation was ANSYS Design Modeler. The mesh contained 2,230,000 elements with both tetrahedron and hexahedron elements. The mesh density was investigated to ensure proper element quality was maintained and a sufficient number of elements were present for accurate energy and mass transfer. The solid domain contained elements ranging in size from 1 mm to 10 mm in edge length and the fluid domain contained 1.8 mm elements at the edges of the drawers, surfaces of the ice, and surface of the vessel exterior, while the bulk fluid cells were 5 mm in length with a growth rate of 1.1.

The computational model containing the fluid domain utilized the three-dimensional Navier–Stokes equations in conjunction with the continuity and energy equations outlined in Chapter 1, Equations 1, 2, and 4 respectively, utilizing a pressure-based solver due to the subsonic incompressible flow with laminar fluid flow. In the pressure-based solver, the momentum and continuity equations, Chapter 1, Equations 1 and 2 respectively, were used in combination to calculate the pressure field. The model was evaluated using standard relaxation for pressure, density, body forces, momentum, turbulence kinetic energy, turbulent dissipation rate, turbulent viscosity, and energy of 0.3, 1.0, 1.0, 0.7, 0.8, 0.8, 1.0, and 1.0 respectively, until all scaled convergence values were below 1.0E-3 with the energy below 1.0E-6.

This model was a passive closed thermal system, thus the only defined boundary conditions were the external wall of the model, Appendix C. A more detailed discussion about the boundary conditions is presented in the section titled “Boundary Conditions Tuning.”

The fluid domain was initially evaluated using a computational model with two symmetry planes to reduce the required computational time, thus allowing the transient results to be generated in a timelier manner. This initial symmetric model provided a greater understanding of the mesh cell size, convergence criteria, and required convergence time as a function of time step and iterations. Information gathered from evaluating the symmetric model helped to reduce the amount of computational time required to evaluate the model and determine which time step allowed for a reliably stable fluid domain. The model indicated a time step up to 500 seconds could be used, but using a time step of 50 seconds resulted in the most efficient computational time. Symmetry could not be utilized over the final configuration of this model because the spatial orientation of the ice blocks was not symmetric in any direction.

4.4.2.1. Material Properties

The fluid and solid material properties are displayed in Table 4.3. The density of air was defined using the Boussinesq Model instead of the full ideal gas law, allowing for the density to be a function of temperature, thus allowing for natural convection. The Boussinesq Model treats the fluid as a constant density in all equations except for the buoyancy term in the momentum equation when it is represented as shown in Equation 17 [57,97]. This allows natural convection-driven flow without the need to solve the full set of compressibility equations. The Boussinesq Model is accurate for fluid domains without large temperature gradients and when chemical kinetics and species modeling are not present [57,97]. Ice was defined as a simplified effective heat capacity method, as defined in the section titled “Phase Change Modeling.” Detailed description for the solid material properties can be found in the sections titled “Composite Thermal Conductivity Tuning,” “Gel-Specific Heat Tuning,” and “Boundary Conditions Tuning.”

$$\boldsymbol{(\rho - \rho_0)g \approx \rho_0\beta(T - T_0)g} \tag{17}$$

Table 4.3: Material properties used in the computational model.

		Units	Values
Air	Density	kg/m^3	1.225
	Compressibility	1/K	0.0036
	Thermal Conductivity	W/m-K	0.0242
	Viscosity	kg/m-s	1.79E-05
	Specific Heat	J/kg-K	1006.43
Aluminum	Density	kg/m^3	2719
	Thermal Conductivity	W/m-K	202.4
	Specific Heat	J/kg-K	871
Cage	Alpha	α	1.17E-07
	Nusselt Number	Nu	0.286
Gap	Gel Gap Alpha	α	8.28E-07
	Ice Gap Alpha	α	1.65E-06
	Gel Gap Nusselt Number	Nu	0.0315
	Ice Gap Nusselt Number	Nu	0.0157
Gel	Alpha	α	9.38E-08
	Nusselt Number	Nu	0.02778
Ice	Density	kg/m^3	998
	Thermal Conductivity	W/m-K	0.58
	Specific Heat @ 200.00 K	J/kg-K	2090
	Specific Heat @ 272.65 K	J/kg-K	2090
	Specific Heat @ 273.15 K	J/kg-K	567800
	Specific Heat @ 273.65 K	J/kg-K	4210
	Specific Heat @ 400.00 K	J/kg-K	4210

4.4.2.2. Phase Change Modeling

The phase change that occurred due to melting of the ice was modeled using a simplified effective heat capacity method [98–100]. Farid et al. introduced a simplified phase change model in 1998, where a single material was used to represent both sides of the phase change and the latent heat from melting was captured in the specific heat of the material property. This simplified model is desirable because it only requires modeling a single phase, while also not requiring a third transitionary phase. In the model, the single phase was modeled continuously at all temperatures, while simultaneously capturing the latent heat of melting in the specific heat of the material. The spike in specific heat was defined at the phase transition temperature. The area under this spike corresponds to the latent heat of melting, Figure 4.6. The width of the spike determines the range over which the phase change occurs. Computational instabilities are introduced as the width of the spike is narrowed. The latent heat from melting ice is 334,000 J/kg and if the specific heat spike is defined over a 1 K width, the maximum specific heat is 668,000 J/kg-K.

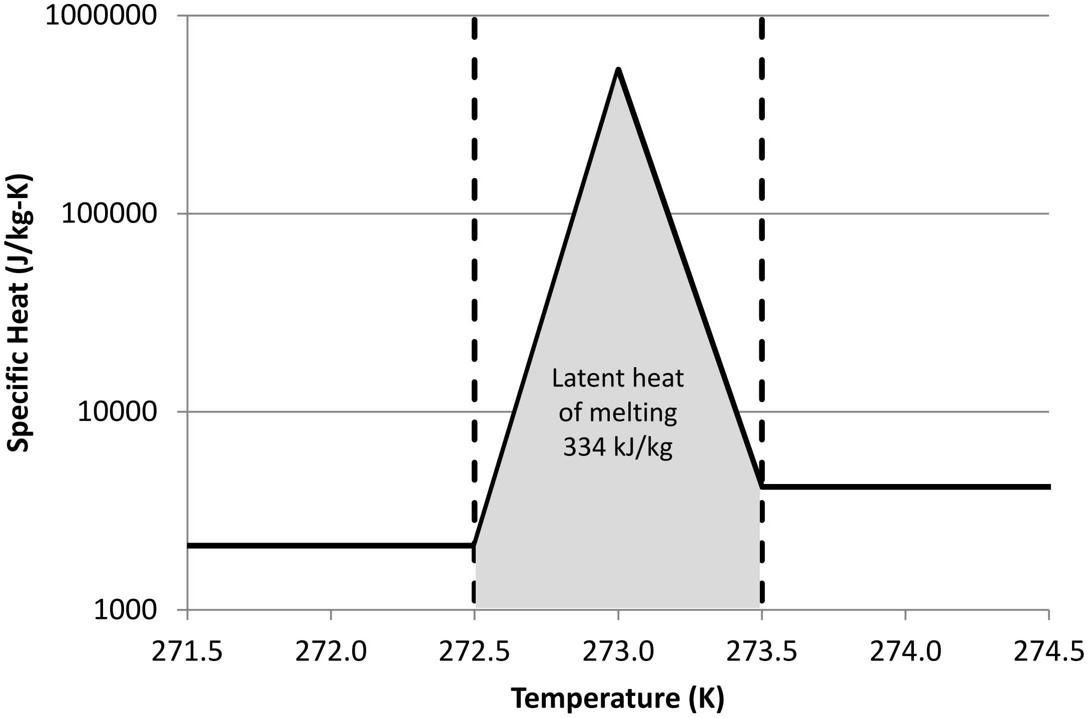


Figure 4.6: Specific heat spike to account for the latent heat of melting ice.

4.4.3. Tuning the Computational Model

4.4.3.1. *Composite Thermal Conductivity Tuning*

A computational model of the vessel containing no gel and no ice was created and the composite thermal conductivity range determined experimentally was used as a starting point to tune the composite thermal conductivity of the computational vessel. The results from this computational model were compared with experimental results under the same empty vessel configuration. The thermal conductivity was then tuned to match the experimental results.

4.4.3.2. *Gel-Specific Heat Tuning*

Gel was used as a substitute for perishable items in order to provide more repeatable and consistent experimental results. The exact composition of the gel was not known, which resulted in uncertainty in the material properties, particularly specific heat. A literature review indicated the specific heat of the gel was between 1.90E-7 to 4.78E-7 α, Table 4.4 [84]. A computational model and experiment containing gel with no ice was created, and the range of specific heat provided by literature was used to tune the specific heat of the gel. The resulting tuned value fell within the published literature range.

Table 4.4: Published specific heat bounds for 20% gelatin gel compared to computationally tuned specific heat value [84].

Upper Bound (α)	1.90E-07
Lower Bound (α)	4.78E-07
Tuned Value (α)	2.00E-07

4.4.3.3. *Boundary Conditions Tuning*

Our industry collaborator performed initial experiments on the vessel under ambient conditions and in a thermal chamber. The experimentally applied thermal loads and measured results were provided. Based upon this information, different boundary conditions needed to be applied to represent the experimental conditions in the thermal chamber. During the experiments when the vessel was at ambient conditions, a constant temperature boundary condition was uniformly applied to all sides of the vessel because the air flow was negligible. However, the air flow in the thermal chamber was significant enough that it could not be ignored. Accounting for the air flow in the thermal chamber required the application of a conjugate heat transfer boundary condition on the top, sides, and bottom of the vessel. The thermal chamber experimental tests were performed with nine vessels in the chamber arranged in a 3x3 array and placed on wooden pallets. This thermal chamber had air entering through the floor and exiting through the ceiling. The walls of the thermal chamber were highly polished metal, resulting in sufficient levels of thermal radiation that could not be ignored. Table 4.5 shows the applied boundary conditions for the experimental test conducted in the thermal chamber.

Table 4.5: External boundary conditions for the top, side, and bottom exterior surfaces of the vessel when testing occurred in the thermal chamber.

	Top	Side	Bottom
Heat Transfer	Convection	Convection	Convection
	Radiation	Radiation	Radiation
Velocity (m/s)	1.0	1.0	1.0
Free Stream Temperature	Measured temperature profile		
Additional Heat Transfer Area	0	0	16x
Convective Coefficient (W/m^2-K)	6.02	6.02	48.16
External Radiation Temperature	Measured temperature profile		
Emissivity	0.09	0.81	0.90

4.4.4. Design Study

The base condition design provided by our industry collaborator failed because it did not maintain the temperature in the desired range for 60 hours. The following steps were taken to determine how much ice and what configuration of ice would produce a passing result while using the smallest amount of ice possible:

1. Determine how much ice mass was needed in each drawer to pass
2. Determine how many full ice blocks were needed in each drawer to pass
3. Determine the significance of the ice spatial arrangement in each drawer

Step 1 of the design study utilized a simplified ice configuration as presented in Figure 4.5 (right). The mass of all ice was increased uniformly in all ice blocks and drawers until the temperatures in all the drawers were maintained in the desired range for 60 hours. This design study resulted in the mass of each ice block being greater than physically possible. However, this simplified ice mass alteration model allowed results to be generated rapidly in an automated method. The results from this study guided the subsequent steps in the design study. Upon completion of this simplified ice mass alteration model, the inaccuracies made by the assumptions of this method were quantified and are discussed in the section "Step 1: Ice Mass per Drawer."

Step 2 of the design study used the information gained in step 1 as a rough estimate for how many full ice blocks were needed in each drawer to maintain a passing result. Only full ice blocks were used in this step and for each ice block added a gel block was removed, resulting in drawers completely filled with ice or gel.

Step 3 of the design study used the number of full ice blocks required in each drawer found in step 2 and arranged the ice blocks spatially in each drawer to determine the optimal configuration. Only full ice blocks were used and for each ice block added, the perishable item in the corresponding location was removed. This resulted in all drawers being completely filled with ice or gel at all times. From these three steps, the optimal configuration was determined.

4.5. Results

4.5.1. Experimental Thermal Conductivity

A total of five different experimental tests were performed to measure the thermal conductivity of the vessel. The results for the experiments are summarized in Table 4.6, where minimum and maximum values are shown. Through the duration of the experimental tests, fluctuations were observed and quantified, resulting in the high and low thermal conductivity measurements. Values that resulted in a lower thermal conductivity were placed in the minimum values column and values that resulted in a higher thermal conductivity were placed in the maximum values column. This provided the most conservative estimate for the lower and upper bounds for possible composite thermal conductivity values for the vessel walls and door. Published data shows the thermal conductivity of polyurethane ranges from 0.02 to 0.90 W/m–K [90–96]. The experimental range lies within this same range and provided a guide for tuning the insulation in the computational model.

Table 4.6: Values used to calculate the composite thermal conductivity of the vessel.

	Minimum Alpha	Maximum Alpha
Temperature difference between T_1 and T_2 (ϴ)	2.572	2.396
Temperature difference between T_1 and T_3 (ϴ)	2.572	2.455
Temperature difference between T_1 and T_4 (ϴ)	2.278	2.071
Thickness of door and vessel wall (m*)	0.123	0.137
Energy generated from heater (W)	4.0	4.1
Heat transfer area (m^{*2})	0.998	0.871
Heat lost through the seal (W)	1.51	0.56
Thermal Conductivity (Nu)	**0.556**	**0.286**

4.5.2. Computational Model Tuning

The results from tuning the composite thermal conductivity using the empty vessel experiments produced a thermal conductivity value of 0.286 Nu. Figure 4.7 shows the external temperature profile and results from the experimental tests and computational models with various thermal conductivity values. These values fall within both the bounds provided by literature and the experimental tests. The results from tuning the specific heat using the gel-only experiments produced a specific heat value of 2.0E-7 α. Figure 4.8 shows the external temperature profile and results from the experimental tests and computational model with various gel-only specific heat values. This specific heat value also falls within the bounds provided by literature.

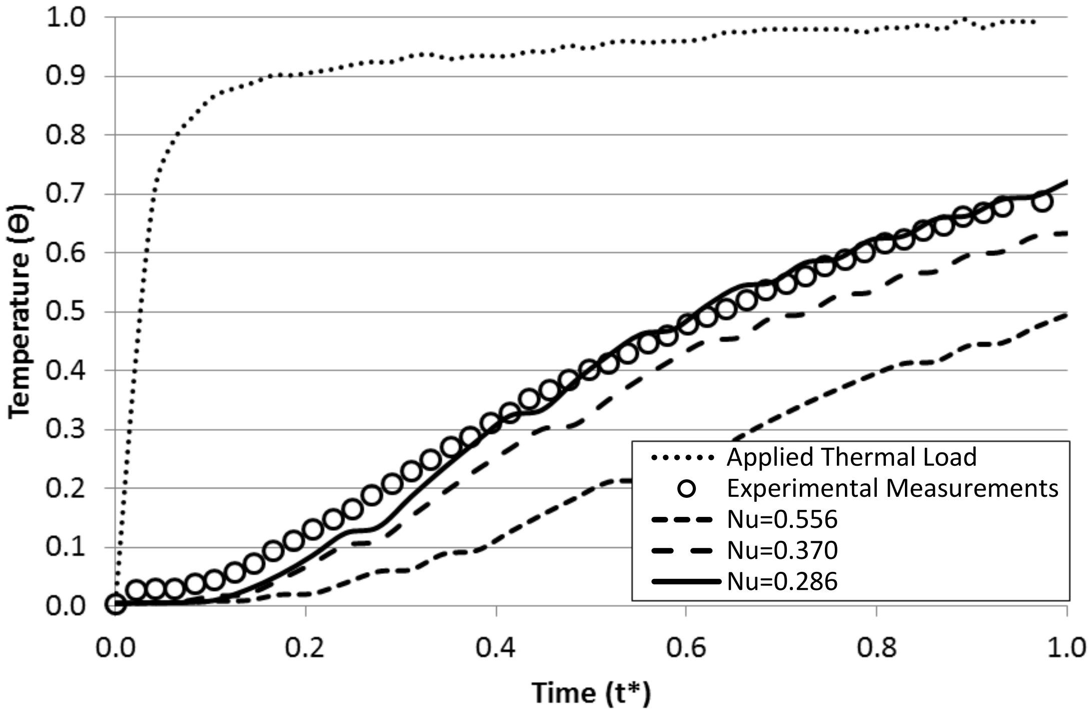


Figure 4.7: Comparison of computational and experimental results for an eight hour empty vessel test with various thermal conductivity values.

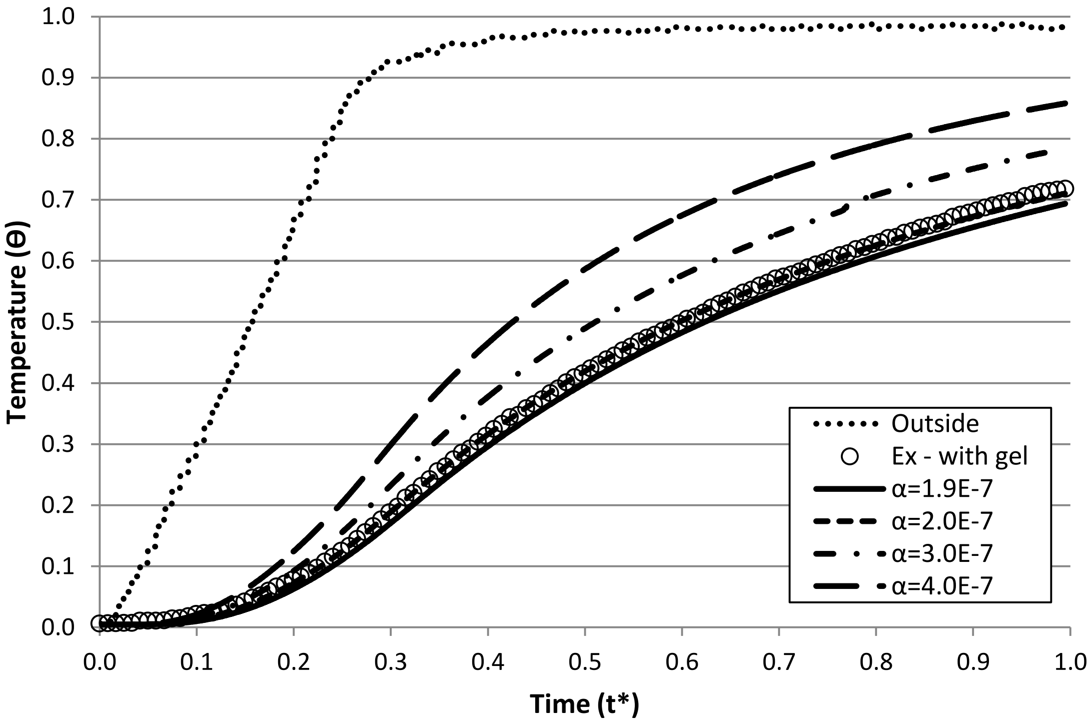
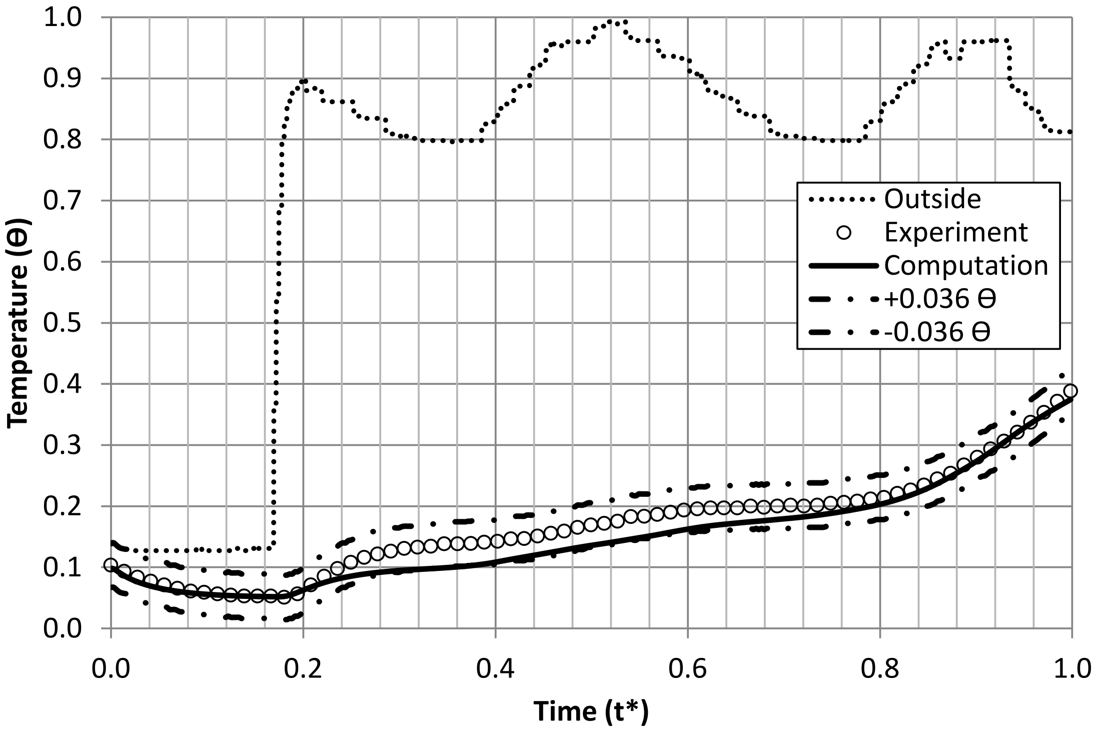


Figure 4.8: Results of computational model with various specific heat values for the gels compared to experimental results.

4.5.3. Validation

The final computational model was validated against experimental results measured in a thermal chamber over 60 hours. The validation was sufficient because the computationally determined temperature in drawer 3 of the vessel did not deviate more than ±0.036 Ɵ from the temperature in the experimental results throughout the duration of the 60 hour test, which was within the experimental uncertainty. Figure 4.9 shows the computational average, experimental average, and ±0.036 Ɵ experimental temperatures in drawer 3 of the vessel throughout the 60 hour test.



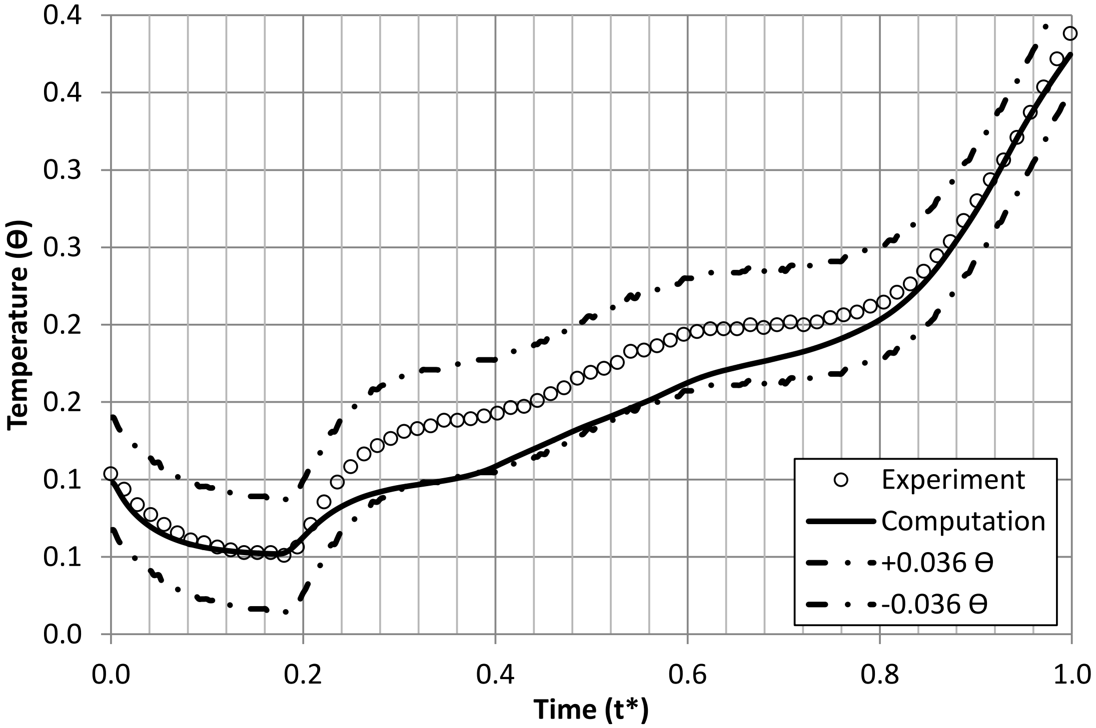


Figure 4.9: Experimental and computational data averages for drawer 3 with temperature bounds at ± 0.036 Θ. The top plot contains the experimental and computational boundary conditions labeled outside and the bottom plot contains a closer view of the experiment and computational results.

4.5.4. Design Study

The design improvement study evaluated 39 design perturbations. Step 1 of the design study evaluated nine designs, step 2 evaluated 21 designs, and step 3 evaluated nine designs. Hundreds of additional designs were evaluated in the tuning and validation process.

4.5.4.1. Step 1: Ice Mass per Drawer

The results from the simplified ice mass alteration model (Step 1) provided the required ice mass in drawers 1, 3, and 5 to maintain the model in the passing region, Figure 4.10. The results indicated that significantly more ice was needed to keep drawer 5 passing, while drawer 3 required the least additional ice mass to pass. This study did result in an unphysical amount of ice in each ice block, i.e. the density of the ice is 2700 kg/m^3. Figure 4.11 shows a direct comparison between the simplified ice mass alteration model (Step 1) and an ice replacement model (Step 2) where gel is replaced when ice blocks are added. In this comparison, both models contain the same total ice mass and the same ice mass per drawer. Drawer 5 for each model contains the equivalent of four ice blocks and drawer 1 contains two ice blocks, with the remaining drawers each containing one ice block each. Significantly lower temperatures were observed in the ice replacement model (Step 2) compared to the ice mass alteration model (Step 1). This discrepancy is not concerning because the extra ice mass comparison study was only used as a guide to determine approximately how many full ice blocks were needed in each drawer to pass.

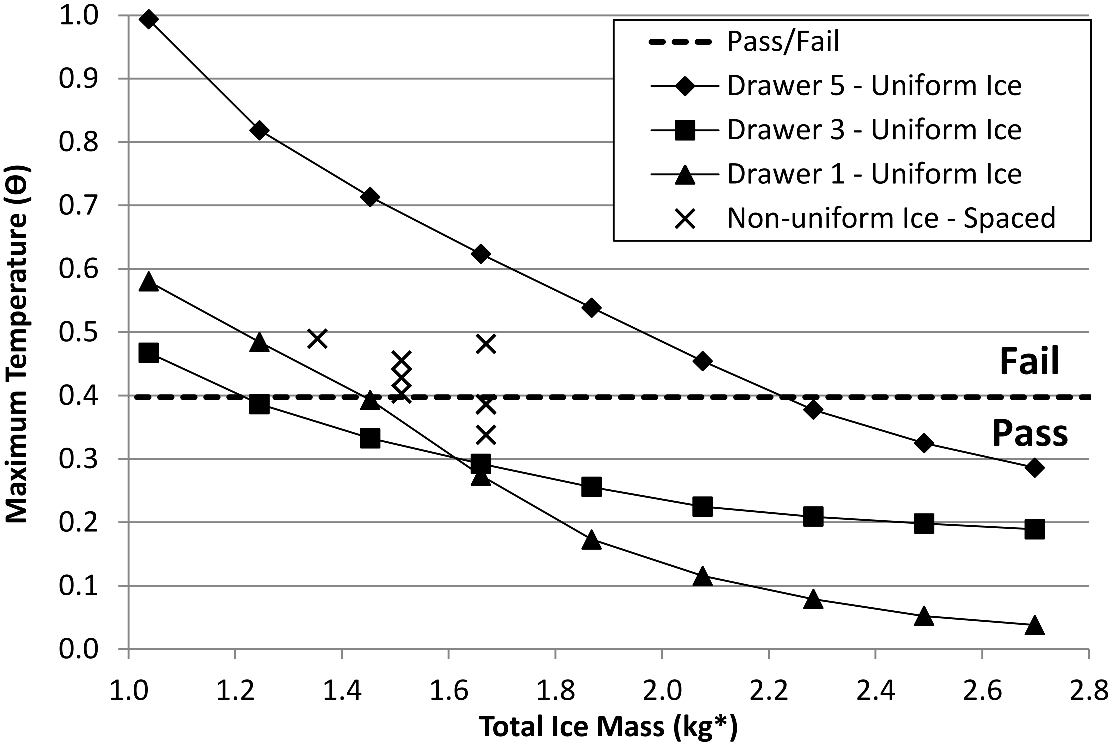


Figure 4.10: Computational results for maximum temperature after 60 hour test as a function of ice mass with uniformly and non-uniformly distributed ice.

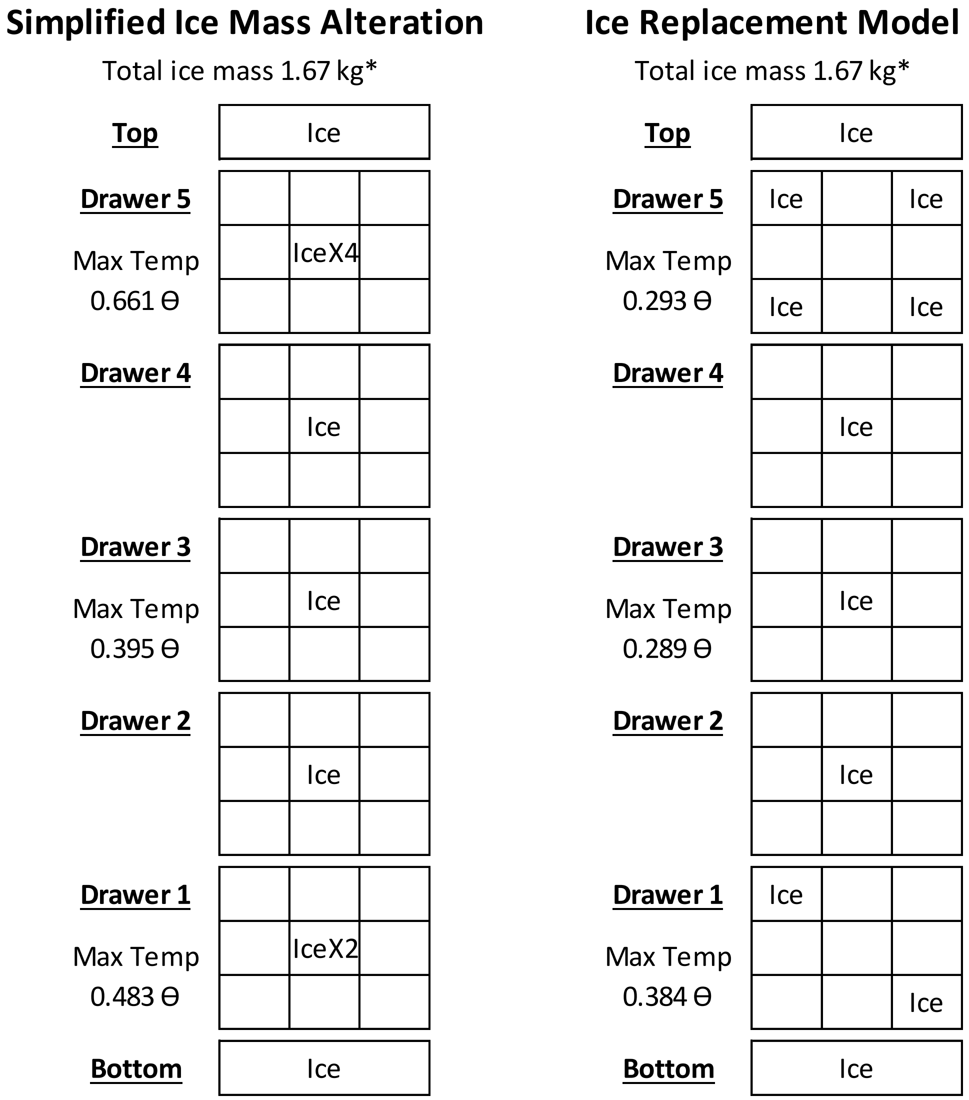


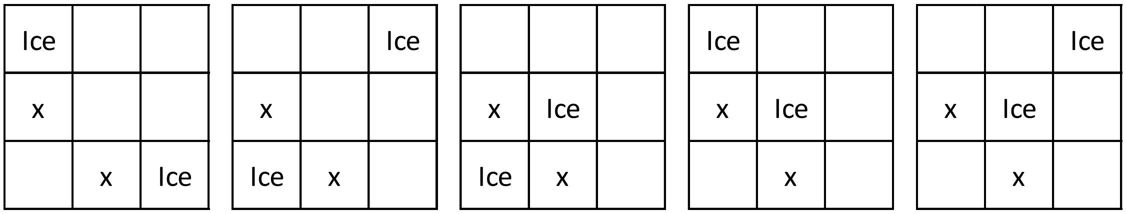
Figure 4.11: Comparison of maximum temperature in each drawer with the simplified ice mass alteration model (left) and ice replacement model (right).

4.5.4.2. Step 2: Full Ice Blocks Required

The results from the full ice block study (Step 2) indicated the minimum ice block configuration required flat ice on the top and bottom of the vessel, four ice blocks in the top drawer, two ice blocks in the bottom drawer, and a single block in each of the remaining drawers. Figure 4.11 (right) shows the configuration for how the nine ice blocks must be distributed as a function of drawers to achieve a passing result. To demonstrate the importance of ice mass per drawer, nine ice blocks were placed in drawer 3 with no ice anywhere else, resulting in a failing test. This step in the design improvement only assessed the ice required in each drawer and did not investigate the spatial arrangement of ice in each drawer.

4.5.4.3. Step 3: Spatial Ice Block Distribution per Drawer

The results from the spatial study (Step 3) indicated the arrangement of ice blocks in each drawer does affect the maximum temperature in each drawer, but this difference was negligible due to experimental uncertainty. Computational models were constructed with five configurations, resulting in the final maximum temperature ranging from 0.338 to 0.415 Θ, as displayed in Figure 4.12. This range of 0.077 Θ is significantly smaller than the temperature difference as a function of vertical ice placement and is within the experimental uncertainty error. The spatial location of the ice in each drawer does affect the maximum temperature, but it was not significant enough for additional investigation.



Drawer 1 Maximum Temperature 0.385 Θ

Drawer 1 Maximum Temperature 0.386 Θ

Drawer 1 Maximum Temperature 0.338 Θ

Drawer 1 Maximum Temperature 0.385 Θ

Drawer 1 Maximum Temperature 0.415 Θ

Figure 4.12: Comparison of ice location and maximum recorded temperature in drawer 1 under various configurations. Location of ice block is indicated by “ice” and temperature measurements indicated by *X*.

4.6. Conclusion

The model presented in this study demonstrated the setup, validation, and lifetime prediction of a thermal-fluid system over long periods of computational time, and that lifetime predictions can be used to increase the life of a system. One key finding is that the vertical arrangement of ice blocks in this passive system was significantly more important than the horizontal arrangement inside each drawer. This model also demonstrated that the exact thermal conductivity value of the system must be known, otherwise the energy balance will not be accurate, thus resulting in incorrect lifetime prediction.

4.7. Lessons Learned

The methods described in this chapter lay the foundation for how a thermal-fluid system should be constructed, validated, and evaluated. The accuracy of computational results pertaining to the evaluation of thermal lifetime over extended periods of time depend heavily on well-characterized and understood fluid material properties, solid material properties, material inconsistencies, and applied boundary conditions.

The evaluation of thermal systems over long time periods involving material properties with low thermal conductivity requires implementing the thermal conductivity as a function of temperature. Even a 10% change in thermal conductivity has a significant effect on the final results for a model evaluated for a 60 hour period.

Improving the defined computational boundary conditions and material properties has a significant effect on the accuracy of matching experimental results; however, the time spent tuning these parameters has diminishing returns. If the goal of a computational model is to explore a design space, generic boundary conditions and broad material properties should be used that capture the physics of the entire design space.

The energy released or absorbed due to phase change can be implemented using a single material and phase by implementation of an effective heat capacity method. This method accurately captures energy transport due to phase change, but does not require the time needed to directly model a transition phase or two independent phases.

The computational time required to evaluate a model can be greatly reduced by selecting an appropriate time step, modeling a single phase before and after a phase change, and indirectly modeling phase change by capturing it in an effective heat capacity method.

CHAPTER 5: LIFETIME DESIGN OF A PERISTALTIC PUMP USING FLUID–STRUCTURE INTERACTION MODELING

5.1. Abstract

Engineers must be able to accurately predict the life of components as they fatigue, even if the fatigue is a result of complex interactions between fluid and solid systems operating with continuous heat transfer. Therefore, coupling fluid–structure interaction models as a function of temperature with fatigue life analysis is a valuable tool for engineers. As one key example, the pharmaceutical industry needs the ability to design better-performing peristaltic pumps with longer fatigue life and thus reduced particle generation and subsequent contamination of the fluid. Currently, after completion of the final filtration process, peristaltic pumps are widely used to fill vials and syringes with individual drug doses. Because the pumps are used to measure drug doses after final filtration, any contaminants introduced by the pump itself will ultimately be injected into the patient. To reduce the level of contaminants present in the drug vials due to tubing wear, the tubing is replaced frequently—a necessary process that requires shutting down the entire pharmaceutical manufacturing line while the tubing is replaced—causing revenue loss for the drug manufacturer and higher drug prices for the patient. Reducing or eliminating the need to close down the manufacturing line is particularly important in the biologic industry where the cost of drugs to the patient may already be more than

$1,000 per month. Therefore, it is desired to determine exactly how long the peristaltic pump tubing can last and what measures can be taken to increase the length of time between tubing replacement. This work demonstrates the ability to model a three-dimensional fluid–structure interaction model and then utilize the model to design for system lifetime. The model incorporated fluid flow, solid mechanics, thermal heat transfer, nonlinear material properties, and fatigue life into a fully coupled model. A parametric study was performed to investigate the effects of geometric changes, fluid properties, solid properties, and operating temperature. This work presents a preliminary study that demonstrates it is possible to determine whether computational optimization using fluid–structure interaction modeling can identify design parameters with the potential to improve current performance. Because most of the tubing material properties used by pharmaceutical manufacturers are proprietary, this work focused instead on the amount of improvement that can be generated from a base case scenario and proves that a detailed design study could yield valuable results if the proprietary tubing material properties are known.

5.2. Peristaltic Pump Background

Peristaltic pumps operate on the principle of positive fluid displacement generated by means of compressing a flexible tube followed by its subsequent release and return to its original shape. The means by which this positive displacement is generated is an inherently multiphysics problem where the fluid flow is generated by large deformation of the pump tubing due to the motion of the pump cams. The performance and fluid flow of the pump are determined by the tubing geometry, tubing material, environmental effects, and fluid properties. A wide variety of tubing is available from various manufacturers, with each manufacturer utilizing a unique chemical compound for its tubing. The diverse material properties of the tubing used are highly dependent upon temperature, age, and fatigue life [101].

5.2.1. Peristaltic Pump Operation

The nature of their positive displacement and resulting fluid motion lends peristaltic pumps to be widely used for fluid transport in pharmaceutical manufacturing, medical devices, and mining [87,102–104]. Some of the reasons peristaltic pumps are preferred include lower fluid shear forces relative to piston pumps, the fluid being pumped does not come into contact with metal surfaces, and the internal mechanisms of the pump can be easily replaced rather than require extensive cleaning [103,104]. There are two methods by which the tubing can be deformed—linear pumping or rotary pumping—but each method results in large tubing deformation, resulting in cracking, fatigue, abrasive wear, particle shedding, frictional heating, and diminishing fluid flow rates. If the tubing is not replaced at appropriate intervals, fluid flow rates can be greatly reduced, particles from the tubing wall can contaminate the fluid, and tubing rupture may occur [87,105,106]. Fluid contamination due to particle generation and the time required for tube replacement is a multimillion dollar problem in the pharmaceutical manufacturing industry [105]. Depending on the pump and fluid being transported, a wide variety of tubing materials can be used, including silicone, rubber, and thermoplastics, with an equally diverse set of material properties for each material type [104,107,108].

5.2.2. Peristaltic Pump Tubing

Failure of peristaltic pump tubing is characterized by rupture or when the flow rate is reduced by 50%, whichever comes first [104]. Rupture is defined as when fluid is no longer fully contained in the tubing due to cracks or abrasive inclusions [104,107]. Balancing these two failure modes requires understanding how the tubing properties perform over time. The flow rate in the tubing is determined by the tube geometry, pump geometry, pumping frequency, operating temperature, and how quickly the tube returns to its original shape after deformation occurs [102,104]. The long-term performance of the pump is determined by how well the tubing retains its original shape over prolonged periods of operation when experiencing cyclic loading at high strain level, temperature gradient, and potentially damaging chemicals passing through the tubing. To maximize the length of time before a 50% flow rate reduction occurs and prevent rupture, tubing manufacturers use proprietary chemical compounds to make the tubing more durable.

5.2.2.1. Tubing Material Properties

The chemical compounds used in peristaltic pump tubing can vary greatly, but can include silicone, rubber, and thermoplastics, to name a few [104,108,101]. These uniquely nonlinear materials, when tested independently, are highly temperature- and time-dependent, and potentially directionally dependent [109,110]. Furthermore, the testing method, clamping technique, and load-dependent aspects of testing these materials can greatly affect the experimental results [109,110]. Due to these challenges present when testing pure compounds, the testing and determination of material properties for mixed compounds can be an even more daunting task. The material properties published by tubing manufacturers, Table 5.1, are insufficient for accurate representation of material properties in a Finite Element Analysis (FEA) model. The only material properties provided by all tubing manufacturers are tensile strength and ultimate elongation, and oftentimes these values are published as a range of values. Furthermore, the values in Table 5.1 represent the final strength of the material before failure, and do not provide any insight into the material behavior between the unstressed state and failure point. This lack of information leads to an infinite number of possible material properties characterizations.

Table 5.1: Material properties for tubing used in peristaltic pumps from a variety of sources, including journal publications, Master's theses, and manufacturing specifications.

Material Name	Source Location	Published Material Properties
Plasticized PVC	Practical Guide to Polyvinyl Chloride [101]	Flexural Modulus from 0-100% Elongation at 25°C is 30 kPa Flexural Modulus from 100-300% Elongation at 25°C is 8 kPa Flexural Modulus from >300% Elongation at 25°C is 4 kPa
Elastomeric Material	Fluid–Structure Interaction Analysis of a Peristaltic Pump [10]	Not Provided
Nylon - linear	The Fluid Structure Interaction Analysis of a Peristaltic Pump [12]	Young's Modulus = 3.0E9 Pa Poisons Ratio = 0.33
Nylon - nonlinear	The Fluid Structure Interaction Analysis of a Peristaltic Pump [12]	Not Provided
64 Grade Bioprene	Watson-Marlow Tubing [107]	Stress at 100% Elongation = 1.9-3.0 MPa Tensile Strength = 5.5-687 MPa Ultimate Elongation = 340-600%
73 Grade Bioprene	Watson-Marlow Tubing [107]	Stress at 100% Elongation = 2.8-4.4 MPa Tensile Strength = 7.2687 MPa Ultimate Elongation = 380-99999%
87 Grade Bioprene	Watson-Marlow Tubing [107]	Stress at 100% Elongation = 6.1-7.80 MPa Tensile Strength = 13.8-687 MPa Ultimate Elongation = 500-99999%

Polytetrafluoroethylene	Dow Corning, Life Sciences [108]	Tensile Strength = 21-35 MPa Elongation at break = 200-400%
Silicone	Dow Corning, Life Sciences [108]	Tensile Strength = 6.8-8.7 MPa Elongation at break = 570-795%
PVC	Dow Corning, Life Sciences [108]	Tensile Strength = 14 MPa Elongation at break = 400%
Polyurethane	Dow Corning, Life Sciences [108]	Tensile Strength = 56 MPa Elongation at break = 550%
Tygon® S3™ E-3603	Masterflex® Tubing [104]	Tensile Strength = 11.4 MPa Ultimate Elongation 450%
Tygon® S3™ Silver	Masterflex® Tubing [104]	Tensile Strength = 15.8 MPa Ultimate Elongation 240%
Tygon® 2001	Masterflex® Tubing [104]	Tensile Strength = 5.51 MPa Ultimate Elongation 500%
Tygothane R C-210-A	Masterflex® Tubing [104]	Tensile Strength = 41.7 MPa Ultimate Elongation 500%
Tygothane R C-544-A	Masterflex® Tubing [104]	Tensile Strength = 34.5 MPa Ultimate Elongation 400%

5.2.2.2. Temperature-Dependent Material Properties

The material and fatigue properties of silicone, rubber, and thermoplastics are temperature-dependent, with the potential to change properties by an order of magnitude over tens of degrees Celsius [101]. At lower temperatures, the tubing is more rigid and brittle, while at higher temperatures it is more flexible and ductile [101]. Many of the published material properties presented in Table 5.1 do not specify the temperature conditions under which the material property testing was performed. Furthermore, if a temperature is provided with respect to the material properties, only a single temperature point is provided; therefore, temperature-dependent properties cannot be defined. The operation of peristaltic pumps in the manufacturing process requires pumps to operate at high rpm, resulting in significant amounts of frictional heating with the heat being dissipated through natural convection and heat transfer to the pumped fluid. Thus, the frictional heating and heat dissipation will result in temperature gradients within the tubing, resulting in non-uniform material properties. Currently, to the author's knowledge there is no published dataset for peristaltic pump tubing material and/or fatigue properties as a function of temperature.

5.2.2.3. Fatigue Life Properties

The failure of peristaltic pump tubing is characterized by rupture or when the flow rate is reduced by 50%, whichever comes first [104]. Rupture is defined as when fluid is no longer fully contained in the tubing due to cracks or abrasive inclusions [104,107]. To determine the time before failure, it is best practice to use a strain-life analysis because of the large deformation experienced by the tubing. Thus, strain-life material properties are required. Many tubing manufacturers publish data pertaining to the life of the tubing produced, but these datasets lack all of the necessary information needed for a proper strain-life analysis. Generally, the data published is presented in number of hours until failure, with some manufacturers providing the pump rpm and number of cams on the pump head, Table 5.2. This provides the number of cycles until failure, but does not provide the stress, strain, force, pressure or displacement the pump exerts on the tube. As discussed previously, temperature has a significant effect on material behavior and although temperatures values are provided with the lifetime data, it is unclear if the temperatures are an average temperature for the entire system or represent the ambient air, fluid inlet or fluid outlet temperatures. Additionally, the tests were only performed at a single temperature; thus, temperature-dependent fatigue trends cannot be gathered.

ASTM testing standards do exist for testing rubber and thermoplastic elastomers, but the tests are highly specimen-specific and results can vary greatly depending on clamping techniques, specimen preparation, loading rates, and temperature conditions [109,110].

Table 5.2: Published fatigue life properties for tubing used in peristaltic pumps from a variety of sources, including published data and manufacturing specifications.

Material Name	Failure Classified By	Operating Setup	Fatigue Life
Tygon® S3™ E-3603	Hours prior to rupture	3-roller pump head at 600 RPM at 73°F	30 hours (10 PSI back pressure) 35 hours (0 PSI back pressure) [91]
Tygon® 2001	Hours prior to rupture	3-roller pump head at 600 RPM at 73°F	70 hours (10 PSI back pressure) 100 hours (0 PSI back pressure) [91]
Tygon® LFL	Hours prior to rupture	3-roller pump head at 600 RPM at 73°F	650-700 hours (10 PSI back pressure) 800 hours (0 PSI back pressure) [91]
Generic Masterflex® Tubing	Failure at rupture or time to 50% of original flow	Variable RPM, unknown number of rollers	Tubing life as a function of RPM [104]
Generic Masterflex® Tubing	Crack growth rates increase by a factor of 5,000 under cyclic loading [101]		

5.3. Problem Description

The problem investigated in this study evaluated the fluid flow, heat transfer, stress–strain fields, and fatigue of peristaltic pump tubing by means of an FSI model combining fluid mechanics, solid mechanics, and heat transfer, Figure 5.1. An 8 mm inner diameter and 11 mm outer diameter polypropylene-based tube was deformed by a 25 mm cam in a linear peristaltic pump. The cam translated in a circular motion at a rate of 60 rpm, depicted in Figure 5.2, to induce tubing deformation and resulting fluid flow. Upon completion of the FSI model, a parametric study was performed to investigate the influence of various parameters on the tubing lifetime.

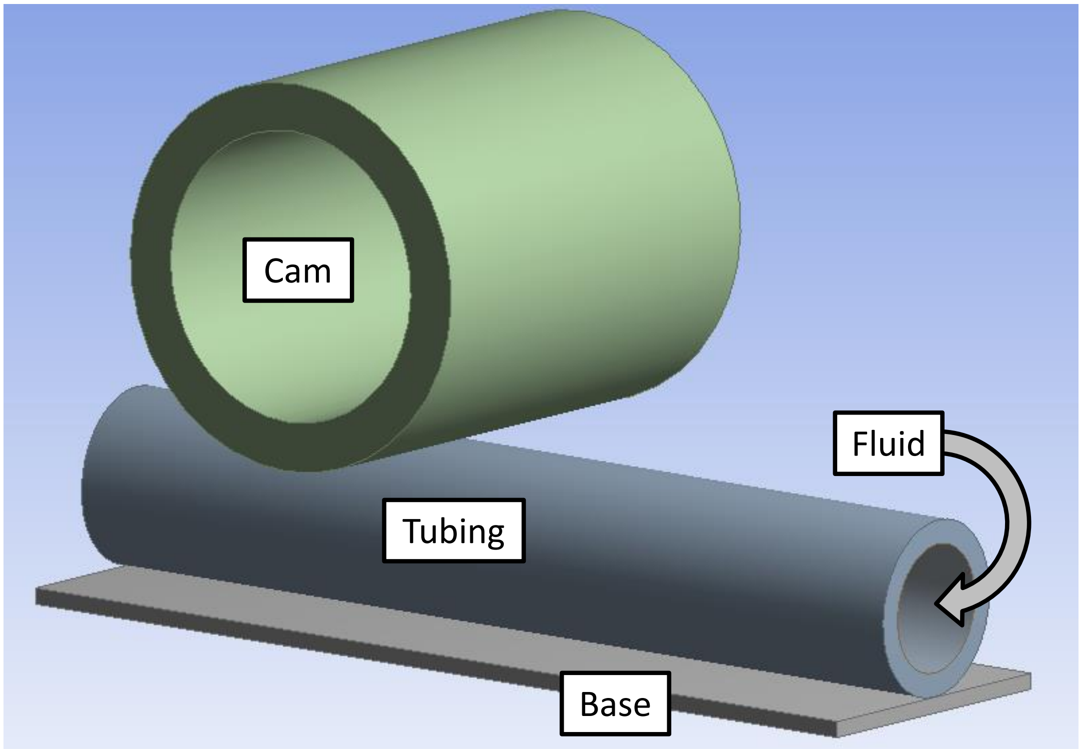


Figure 5.1: Model overview for the FSI model with named components in the peristaltic pump.

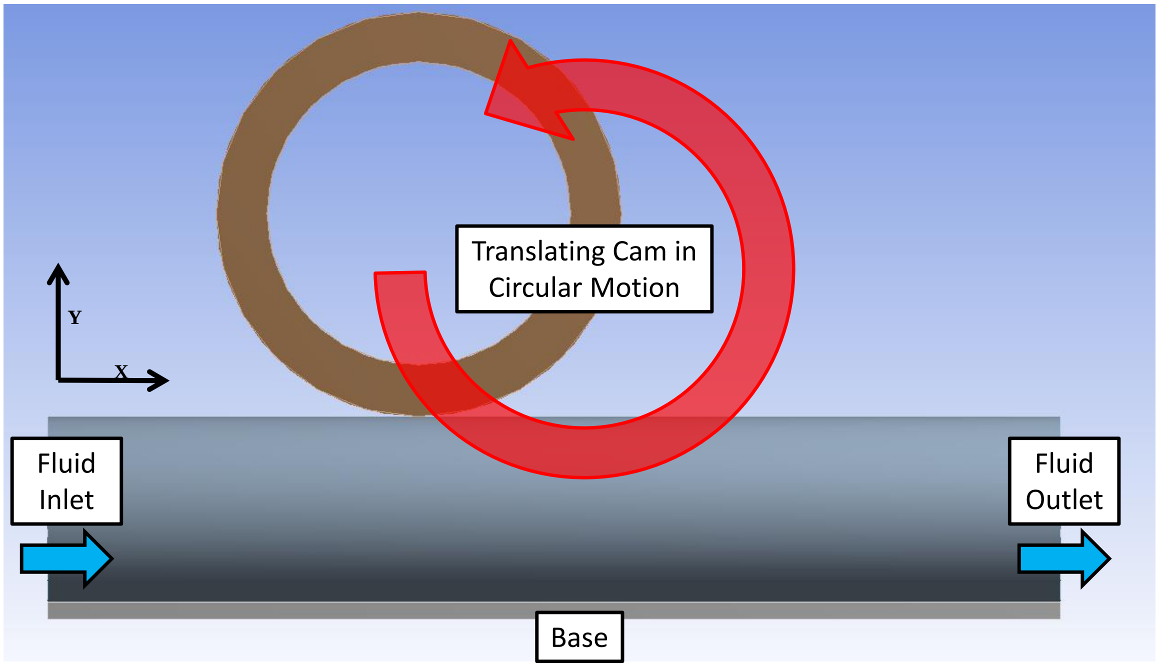


Figure 5.2: Diagram of the solid motion and resulting fluid flow directions.

5.4. Methods

This study utilized a two-way FSI model in which the effects of solid mechanics, fluid mechanics, and heat transfer were coupled between the fluid and solid domains. Computational Fluid Dynamics (CFD) modeling was used to evaluate the fluid mechanics and heat transfer within the fluid flow. FEA was used to evaluate the solid mechanics and heat transfer within the solid domain. The fluid and solid domains were linked, allowing heat transfer between both models. Once the model containing the base condition was complete, a parametric study evaluated several additional variables, including tubing geometry, tubing material properties, fluid properties, and operating temperatures.

5.4.1. CFD Numerical Methods

The fluid domain was evaluated using CFD modeling with the commercially available software ANSYS Fluent 17. The pre-processor used for generating the geometry and meshing of the fluid domains was ANSYS Design Modeler, which will be discussed in detail in the section titled “FSI Numerical Methods.”

The computational model containing the fluid domain utilized the three-dimensional Navier–Stokes equations in conjunction with the continuity and energy equations outlined in Chapter 1, Equations 1, 2, and 4 respectively, utilizing a pressure-based solver due to the subsonic incompressible flow, along with a laminar turbulence model. In the pressure-based solver, the momentum and continuity equations, Chapter 1, Equations 1 and 2 respectively, were used in combination to calculate the pressure field.

The fluid domain contained incompressible water under laminar conditions passing through the pump tubing, Table 5.3. The maximum Reynolds number at any point in the flow field is 1600. The fluid inlet was defined as a pressure inlet with 10 Pa at 20°C, the outlet was defined as a pressure outlet with 0 Pa, and the walls were defined as smooth walls with temperature data transfer coupled with the solid domain, Figure 5.3 and Table 5.4. These pressure values correspond to published data from Masterflex® and other tubing manufacturers as outlined in their protocol for fatigue testing of peristaltic pump tubing [104]. A more detailed discussion about the thermal boundary conditions along the fluid wall is covered in the section titled "FSI Numerical Methods." Dynamic smoothing and remeshing was used to maintain adequate element quality as the tubing was clamped closed. Initially, the fluid domain contained an unstructured 200,000 element tetrahedron mesh with an edge element size of 0.4 mm along the walls growing to 1 mm in the bulk of the fluid, resulting in approximately 20 elements through the diameter of the tube before dynamic meshing occurred, Figure 5.3. Dynamic meshing was performed to maintain sufficient element quality through the thickness of the fluid flow, and the meshing parameters are presented in Table 5.6. The model was evaluated using standard relaxation for pressure, density, body forces, momentum, turbulence kinetic energy, turbulent dissipation rate, turbulent viscosity, and energy of 0.3, 1.0, 1.0, 0.7, 0.8, 0.8, 1.0, and 1.0 respectively, until all scaled convergence values were below 1.0E-3 with energy below 1.0E-6.

Table 5.3: Material properties of water used in the CFD model.

Density (kg/m3)	998.2
Viscosity (kg/m-s)	1.003E-03
Specific Heat (J/kg-K)	4182
Thermal Conductivity (W/m-K)	0.6

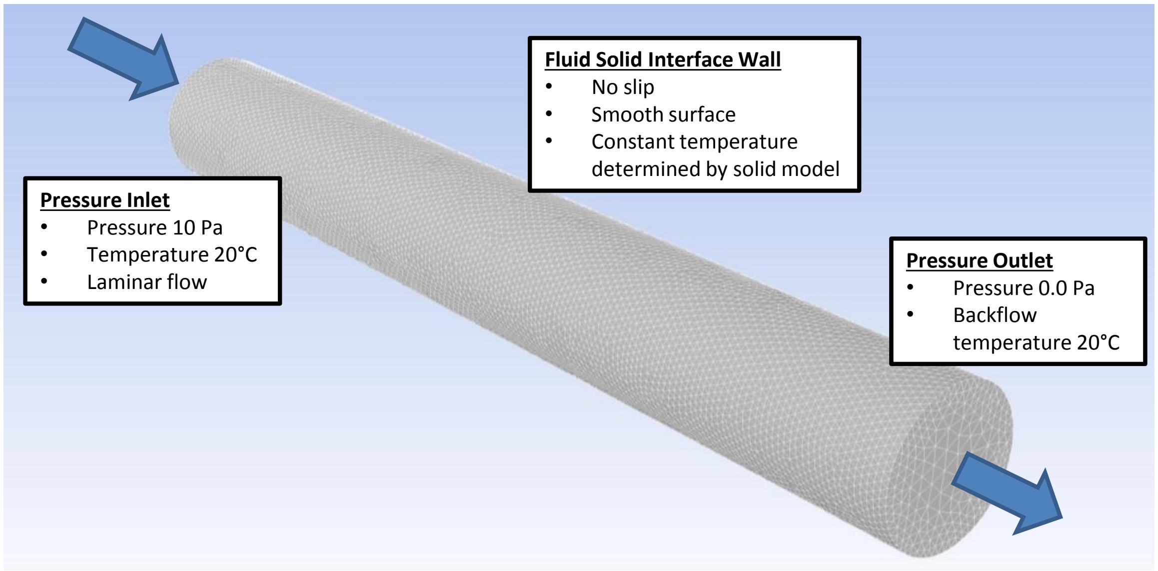


Figure 5.3: Fluid tetrahedron mesh with temperature and flow boundary conditions.

Table 5.4: Boundary conditions for fluid and solid domains.

Boundary Condition Name	Domain	Type	Settings
Fluid inlet	Fluid	Pressure inlet	Pressure 10 Pa Temperature 20°C Laminar flow
Fluid outlet	Fluid	Pressure outlet	Pressure 0.0 Pa Backflow temperature 20°C
Fluid solid interface wall	Fluid	Wall	No slip Smooth surface Temperature defined from solid domain Coupled with dynamic meshing to allow for motion
Tube-fixed	Solid	Fixed support	Fixed in all degrees of translation Fixed in all degrees of rotation
Base	Solid	Fixed support	Fixed in all degrees of translation Fixed in all degrees of rotation
Cam	Solid	Joint Movement	X displacement defined in Equation 14 Y displacement defined in Equation 15

The fluid domain was initially evaluated independently of the solid domain and without dynamic meshing. This simplified, uncoupled CFD model allowed for greater understanding of the mesh cell size sensitivity, convergence criteria as a function of fluid flow rate, and required convergence time as a function of time step size and number of iterations. This information gathered from evaluating just the CFD model without dynamic meshing or the solid domain provided valuable insight into what time step size and flow rate produced the optimal balance of a reliably stable fluid solution while reducing the amount of computational time required to evaluate the model.

5.4.2. FEA Numerical Methods

The solid domain, including the temperature distribution in the solid, was evaluated using FEA with the commercially available software ANSYS Mechanical 17. The pre-processor used for generating the geometry and mesh was ANSYS Design Modeler, which is discussed in the section titled “FSI Numerical Methods.”

The computational model utilized the three-dimensional strain displacement, nodal displacement, and stress equations, Chapter 1, Equations 8 through 10 respectively, to solve for the force, deformation, stress, strain, and contact across each node in the model. The rigid cams and base surface mesh contained 3,450 QUAD8 elements and the deformable tubing volume mesh contained approximately 13,000 HEX20 elements. A mesh density study was performed to determine the minimum number of elements required for the length, thickness, and circumference of the tubing. Furthermore, the elements used were defined as thermal–structural elements allowing for thermal degrees of freedom as well as structural degrees of freedom.

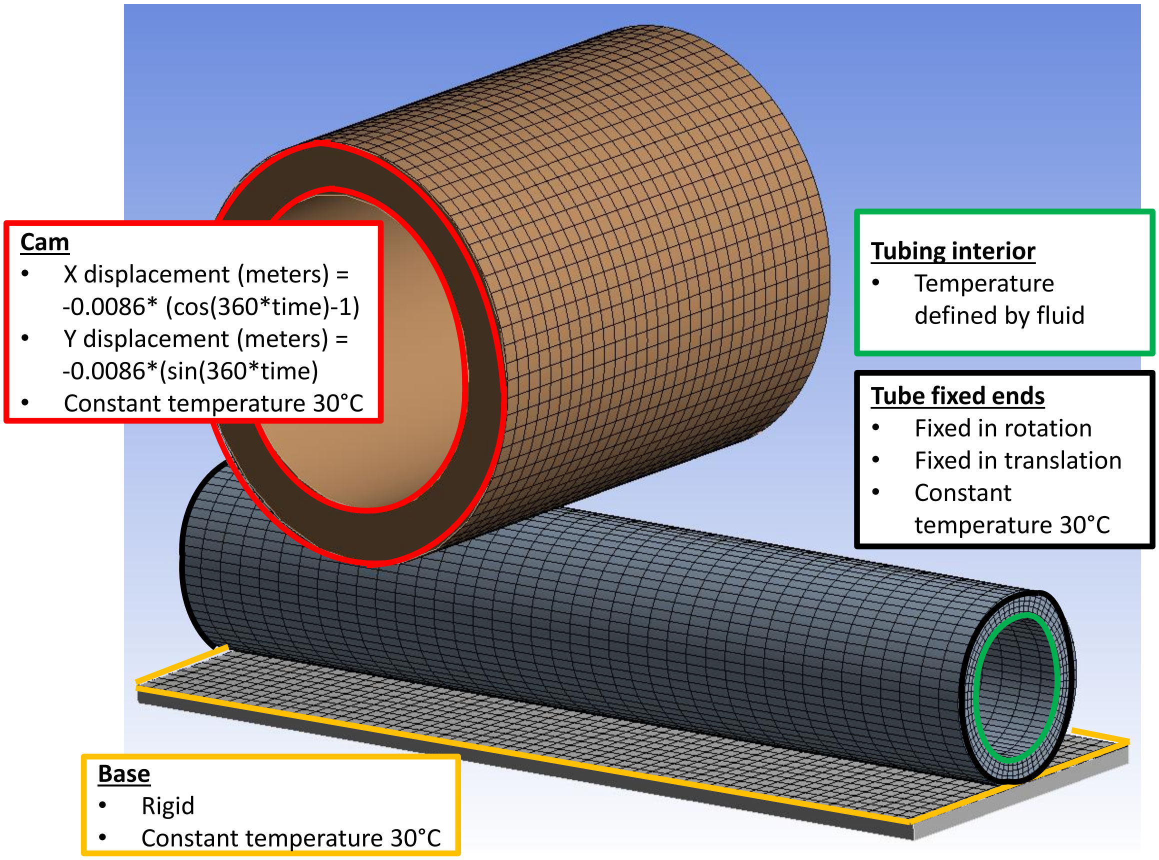


Figure 5.4: Solid domain with QUAD8 elements along the surface of the cam and base with HEX20 elements through the volume of the tubing with boundary conditions defined.

The circular motion of the cam is defined by the X and Y displacement with respect to time using Equations 14 and 15 respectively. These equations represent a pump operating at 60 rpm. This is the lower end of operational pump speed, but demonstrates the operational feasibility of the FSI model. A time step of up to 0.001 seconds was utilized, but the model could be evaluated more efficiently if variable time steps were used. A more detailed discussion of the time step setup is contained in the section titled “FSI Numerical Methods.” Contact between the cam and tube was defined as frictional with a coefficient of static friction of 0.2 [111]. Contact between base and tube was defined as frictional with a coefficient of static friction of 0.1 [111]. Self-contact within the tube was defined as frictional with a coefficient of static friction of 0.2, and a gap size of 0.5 mm was defined to ensure a small gap was present so that convergence could still be achieved in the fluid domain. Energy conservation was achieved by applying a heat source term equal to the energy absorbed through friction [4]. The frictional heat source term was applied at a nodal basis resulting in heat generation directly at the source of the friction. This heat source was applied at all three contact pairs along the exterior and interior of the tubing. The exterior surfaces were treated as warm ambient temperature at 30°C. The interior tubing surface in contact with the fluid was defined as a convective boundary condition and will be discussed in more detail in the section titled "FSI Numerical Methods."

$$\mathbf{X\ Displacement\ in\ meters} = -\mathbf{0.008} * \left(\mathbf{cos}(\mathbf{360} * \mathbf{time}) - \mathbf{1}\right) \tag{14}$$

$$\mathbf{Y\ Displacement\ in\ meters} = -\mathbf{0.008} * \mathbf{sin}(\mathbf{360} * \mathbf{time}) \tag{15}$$

Although ANSYS Mechanical is a robust FEA software capable of solving a wide range of complex solid mechanics problems, at its core it assumes the model is under uniform temperature distribution in both time and space. In order to implement temperature-dependent degrees of freedom, the element type must be redefined as a multiphysics element through text commands located in Appendix D. Once temperature degrees of freedom were enabled, thermal material properties and boundary conditions were implemented through text commands, Appendix D. Heat generation as a function of frictional heating was governed by Equation 16, while the frictional heating factor of 10 W/m^2–K and even heat distribution into each contact surface were defined using text commands, Appendix D [4].

$$\textbf{Heat generated} = (\textbf{Frictional Heating Factor}) * (\textbf{Contact Pressure}) \qquad \textbf{(16)}$$

Due to the complex features implemented in the solid domain, it was initially evaluated uncoupled from the fluid domain to ensure the complex intricacies of the solid domain were properly implemented. The features evaluated in the uncoupled solid domain were (1) heat transfer in the solid domain, (2) nonlinear material properties, (3) temperature-dependent material properties, (4) multiple contact regions, and (5) frictional heat generation. Additional detail about each evaluation is discussed below.

(1) Testing the multiphysics elements and heat transfer in the solid domain was performed by applying constant temperature and convective boundary conditions, while observing the temperature distribution and heat flux between the domains. This was done using various combinations of boundary conditions to ensure the thermal boundary conditions were applied properly and the heat was being distributed correctly in both space and time.

(2) The three material properties investigated, Figure 5.5, were initially implemented at constant temperatures to isolate the effects of the nonlinearities of the

material properties. This nonlinear material testing resulted in nine separate evaluations of the model to test the entire range of material properties independently from temperature.

(3) The material properties that were previously tested were then applied to the model with thermal boundary conditions that resulted in temperature gradients across the model. The resulting deformation, stress, and strain was then observed to ensure the material properties were changing as a function of temperature. This was conducted for each of the three material properties curves.

(4) The three contact regions were tested to ensure element penetration was not achieved and element quality was maintained within the solid model. Testing the self-contact region along the tubing interior was of particular importance, because poor element quality along the pinched portion of the tube was likely. Both visual and numerical methods were used to ensure element quality was maintained when self-contact was achieved.

(5) Frictional heat generation was tested while using perfectly insulated external boundary conditions. This setup was evaluated both with and without friction. The results were compared to ensure heat generation was present in the frictional model.

Once the desired features were tested and properly implemented, the overall stability of the model was evaluated to determine the limits of the hyperelastic material and ensure the most efficient time step and mesh were utilized.

5.4.3. Peristaltic Pump Tubing Properties

The materials properties published in S.G. Patrick's *Practical Guide for Polyvinyl Chloride* were used to represent the nonlinear and temperature-dependent materials of the peristaltic pump tubing [101]. Because of the proprietary nature of the exact tubing used in the pharmaceutical industry, operational material properties bounds were created from the properties defined by Patrick [101]. Patrick characterizes the material properties of various types of PVC as a function of chemical composition, temperature, and strain level. Patrick's characterization of plasticized PVC most closely relates to the material used for peristaltic pump tubing. From this dataset, material properties were created at three temperatures and a linear interpolation method was used to determine the properties at other temperature between the defined values. This dataset does not provide details for material behavior between unstressed and 100% strain measurement. Therefore, three material assumptions were made to demonstrate the completeness of the computational model by evaluating the extremes that encompass the actual material properties of the peristaltic pump tubing. Figure 5.5 and Table 5.5 present the material properties for the linear, upper bound, and lower bound materials at 0, 25, and 45°C. The author believes this range of material properties encompasses the accurate nonlinear temperature-dependent material properties if the actual material properties could be obtained. Additionally, the author believes that once the accurate material properties are obtained, the properties can be implemented into the computational model and the model will achieve stable results.

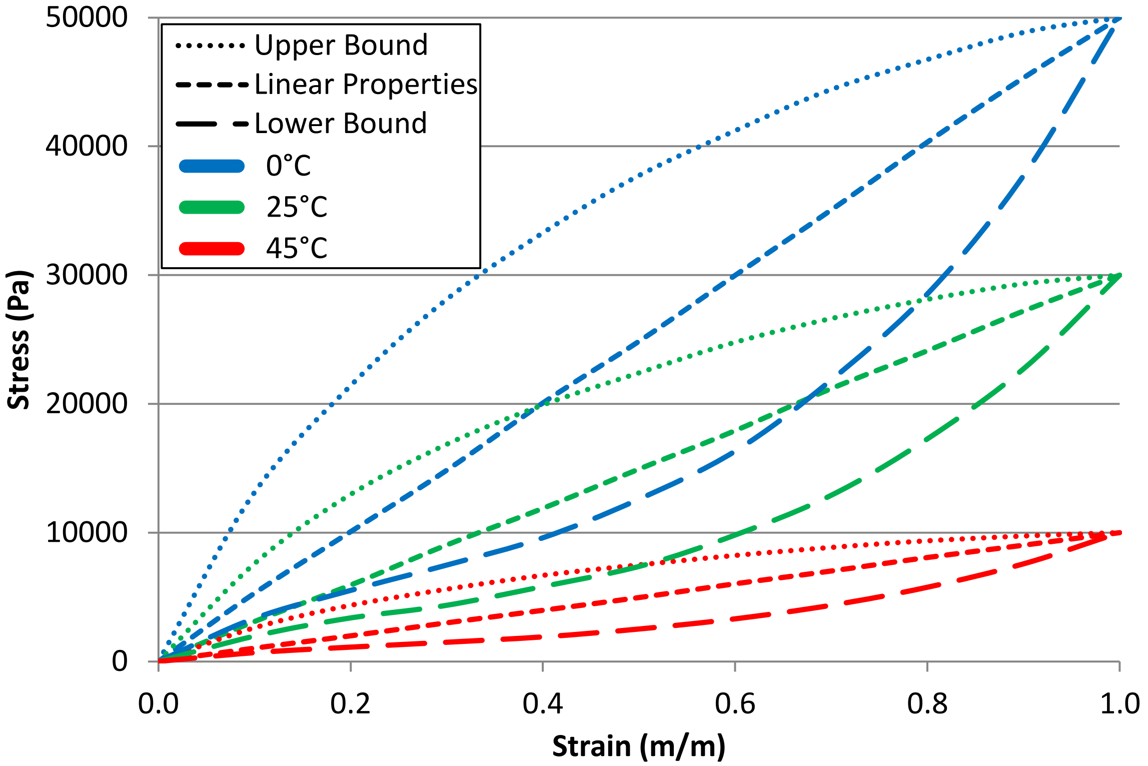


Figure 5.5: Possible material properties paths for stress–strain material properties behavior between the unstressed and 100% strain measurement.

Table 5.5: The material properties for the linear, upper, and lower bounds were defined using a three parameter Mooney–Rivlin hyperelastic model.

Material Name	Operating Temperature (°C)	Stress at 100% Strain (Pa)	C01 (Pa)	C10 (Pa)	C11 (Pa)	D (1/Pa)
Linear Plasticised PVC	0	50,000	-183.8	4627	-167.3	1.0
	25	30,000	-110.3	2776	-100.4	1.0
	45	10,000	-36.76	925.4	-33.45	1.0
Upper Bound Plasticised PVC	0	50,000	12170	986.0	-101.2	1.0
	25	30,000	7302	591.6	-60.71	1.0
	45	10,000	2434	197.2	-20.24	1.0
Lower Bound Plasticised PVC	0	50,000	5221	-1487	397.9	1.0
	25	30,000	3132	-892.5	238.7	1.0
	45	10,000	1044	-297.5	79.57	1.0

5.4.4. FSI Numerical Methods

ANSYS Design Modeler was used as the pre-processor for generating the fluid and solid geometries in a single software. This allowed for the geometries to be created simultaneously before being split into separate domains for meshing and analysis. ANSYS System Coupling was used to transfer data between the fluid and solid domains in a strong-coupled manner at each time step. A relaxation factor of 1.0 was used for transferring data between each domain, i.e. no ramping function or damping was introduced and the full loads were applied at each data transfer. The fluid domain provided pressure loads, near wall temperature, and the convective heat transfer coefficient, while the solid domain passed the displacement and wall temperature, Figure 5.5. Other configurations of the thermal energy transfer were implemented, but were significantly less stable. It is hypothesized that this thermal energy transfer is more stable because the convection coefficient and near wall temperature values are calculated within Fluent as opposed to the ANSYS Mechanical solver. Fluent is well known in the CFD industry for having a robust and stable solver; therefore, allowing it to effectively handle discrete changes in the transferred thermal boundary conditions passed by ANSYS Mechanical. The model was evaluated until each CFD and FEA model converged independently and the root mean square of the data transfer error was less than 0.01 for each data transfer.

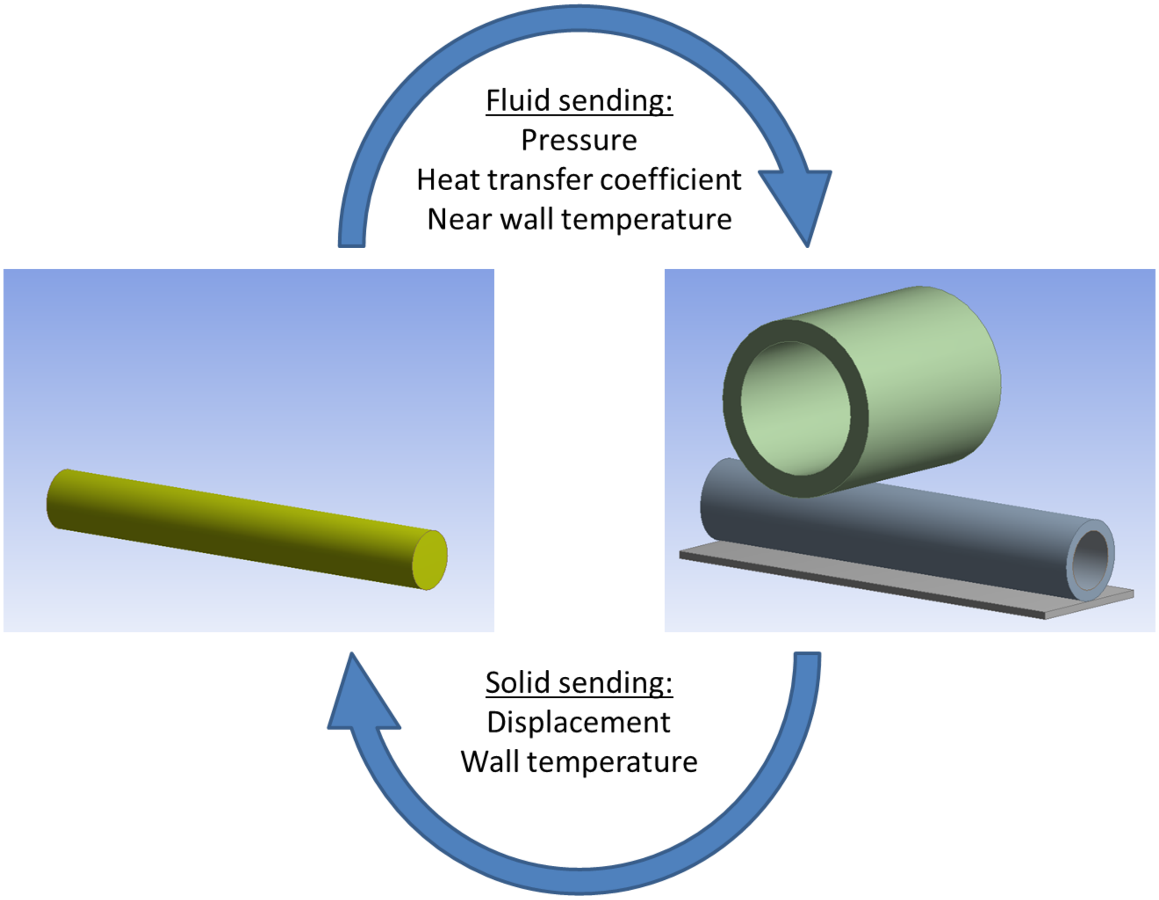


Figure 5.6: Diagram of the system coupling data transfer methods between the fluid and solid models.

Each domain was evaluated independently prior to evaluating them as a single coupled model. This process of initially evaluating each model independently allowed for a greater understanding of the instabilities contributed from time steps, contact, data transfer, and dynamic meshing. The maximum allowable time step for the fluid side was 0.1 second when the flow channel was not restricted and 0.005 seconds when the flow channel was closed. The allowable time step for the solid model was 0.05 seconds when minor deformations were present and 0.001 second with large deformations and tubing self-contact. One of the limitations with ANSYS System Coupling is the same time step must be used for the CFD and FEA models and the time step cannot be changed as a function of time [4]. This means the same time step must be used when there is little deformation with no flow constriction and when there is large deformation with significant flow constriction. Therefore, a time step of 0.001 second was used for the coupled CFD and FEA modeling over the full 0.5 seconds of the evaluation.

Once the FSI numerical model was set up, the dynamic meshing and data transfer methods were evaluated to ensure they were implemented properly. The only region where dynamic meshing was performed was in the bulk fluid, where a minimum cell size of 0.1 mm, maximum cell size of 1.0 mm, maximum skewness of 0.9, and dynamic smoothing/diffusion parameter of 0.5 was used. The fluid–solid interface defined in the fluid model maintained at least a 0.1 mm cell thickness along the wall. The solid domain contact surface containing the fluid domain was defined to maintain a gap size of 0.5 mm to ensure a negative volume was not generated in the fluid domain. This 0.5 mm gap allowed for at least four elements to be maintained across the thickness of the fluid channel at maximum deformation. After the dynamic meshing was set up, the heat transfer between the fluid and solid domains was tested as defined in Figure 5.6. Other thermal data transfer configurations were explored, but either did not produce results as efficiently or failed to converge.

Table 5.6: Dynamic meshing conditions.

Location	Minimum cell size (mm)	Maximum cell size (mm)	Maximum skewness	Dynamic smoothing - diffusion parameter
Fluid Solid Interface Wall	0.1	1.5	0.9	0.5
Bulk Fluid	0.1	1.5	0.9	0.5

Only after the fluid domain, solid domain, and coupling models were independently tested were the results of the multiphysics model processed. This process of evaluating each domain independently and then confirming the success of individual components together allowed for a greater understanding of the instabilities contributed by each portion of the model.

5.4.5. Failure Analysis

Upon completion of the FSI model, the stress and strain field were input into a strain-life analysis to determine the number of cycles until failure. Due to the unpublished fatigue life properties, arbitrary strain-life data was used and the results were nondimentionalized by Equation 17 where $Cycles\ until\ failure_{base}$ is the life of the base conditions. This step in the analysis process demonstrates how this modeling tool can be implemented if appropriate material and fatigue properties are obtained. A general comparison was be made regarding the life of the tubing in the various cases in the parametric study. However, this was only a qualitative analysis to determine which parameters in the parametric study increase the tubing lifetime.

$$\boldsymbol{Life^{*}} = \frac{\boldsymbol{Cycles\ until\ failure}}{\boldsymbol{Cycles\ until\ failure_{base}}} \qquad \textbf{(17)}$$

5.4.6. Parametric Study

Once the computational model base condition was completed, a parametric study was performed to determine what factors had the greatest effect on tubing life. As previously discussed in the section titled “Peristaltic Pump Tubing,” tubing life was classified by two modes of failure. Because strain hardening material properties are not available, the tubing degradation and reduced fluid flow cannot be characterized. Thus, only the fatigue failure mode was investigated. Although the lifetime of peristaltic pump tubing is important in many industries and applications, this parametric study and resulting discussion were specifically tailored for application by the pharmaceutical manufacturing industry. In addition to the base condition, four additional design parameters were explored, Table 5.7: (1) tubing geometry, (2) tubing material properties, (3) fluid properties, and (4) operating temperature.

(1) Tubing geometry is potentially the easiest and most cost effective design change and was investigated by changing the tubing wall thickness from 1.5 to 2.0 mm, while maintaining the same 11 mm tubing outer diameter.

(2) The tubing material properties have the potential for the most drastic improvement in life, but pose challenges such as the difficulty in accurately defining material properties, changing one property without affecting another, and balancing the two failure modes to achieve the best match for life. For this reason, the generalized tubing material properties were utilized. The base condition was evaluated using linear material properties and the altered material properties were evaluated using the upper and lower limit of the nonlinear material bounds.

(3) The pharmaceutical industry has the desire to pump solutions at higher drug concentrations, which results in higher fluid viscosities [106]. The pumping efficiencies of peristaltic pumps are directly related to the fluid viscosity and the relationship between

drug concentration and fluid viscosities is exponential [102,104]. Therefore, two cases were evaluated with fluid viscosities at 0.01 and 0.1 kg/m-s (SAE 10W-40 at 100°C is 0.0148 kg/m-s and 0.104 kg/m-s at 40°C) [112].

(4) Due to the highly temperature-dependent material properties of the pump tubing, the operating temperature of the system were adjusted by means of fluid inlet temperature from in the base condition at 20°C to a chilled temperature at 10°C.

Table 5.7: Cases investigated during the parametric study.

Design Name	Tubing Wall Thickness (mm)	Solid Material Property Behavior	Viscosity (kg/m-s)	Fluid Inlet Temperature (°C)
Base Case	1.5	Linear	0.001003	20
Case 1	2.0	Linear	0.001003	20
Case 2a	1.5	Upper bound	0.001003	20
Case 2b	1.5	Lower bound	0.010000	20
Case 3a	1.5	Linear	0.010000	20
Case 3b	1.5	Linear	0.100000	20
Case 4	1.5	Linear	0.001003	10

5.5. Results

5.5.1. Base Condition Results

5.5.1.1. Fluid Results

The computational results for the FSI model indicate the total flow over one cycle to be 2.32 ml at a pump speed of 60 rpm. At maximal flow constriction the fluid channel is 0.5 mm tall as specified by the contact methods. Figure 5.7 shows a cross sectional view of the tetrahedron mesh at 0, 0.25, and 0.5 seconds. The dynamic meshing and solid contact settings allow for between four and five elements in the thickness of the tubing at maximum deflection. The average flow rate of the exit over two cycles is seen in Figure 5.8. The flow rate reaches steady-state in the tubing in the second cycle after the maximum flow rate is reach and the cam is nearly fully clamped. The velocity contour plot at 0, 0.125, 0.25, 0.375, and 0.5 seconds is presented in Figure 5.9.

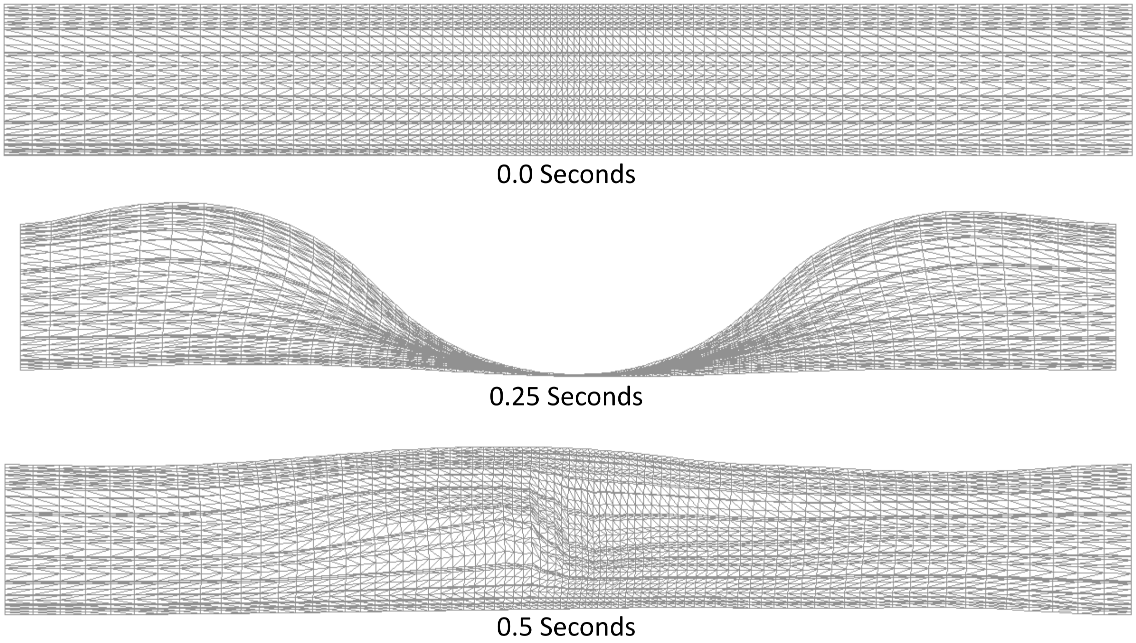


Figure 5.7: Cross section of three-dimensional tetrahedron mesh at 0, 0.25, and 0.5 seconds.

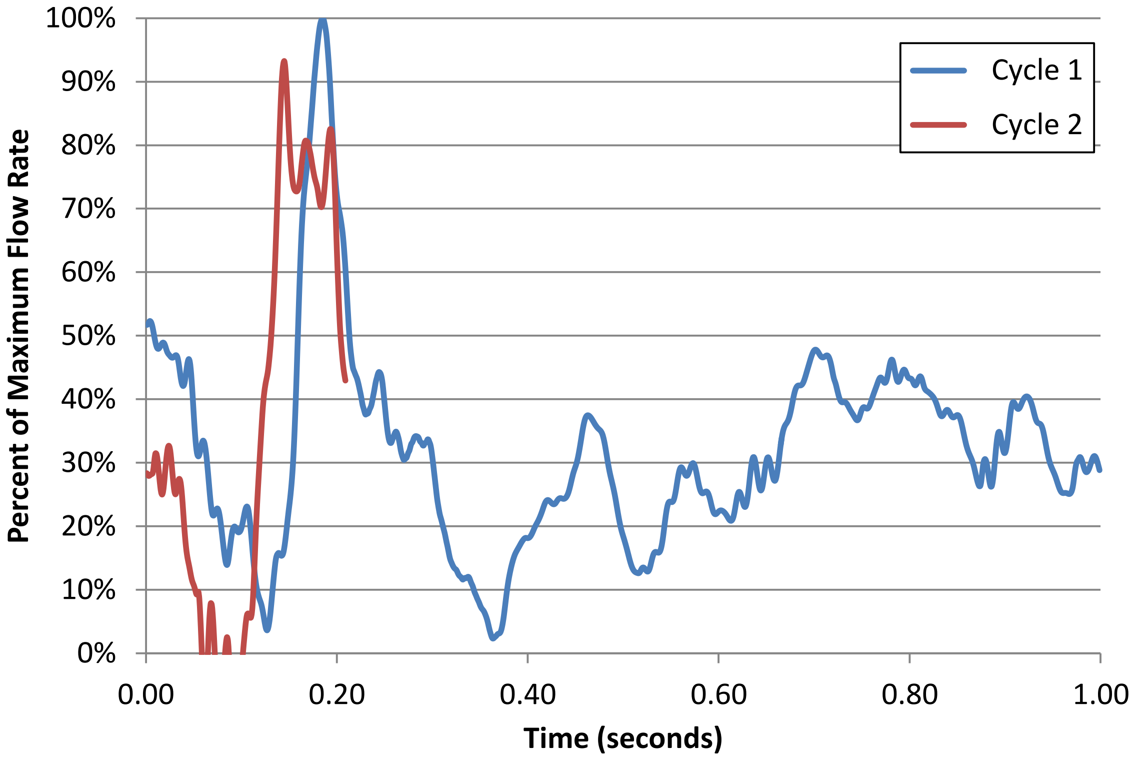


Figure 5.8: The flow rate as a function of time over two pumping cycles.

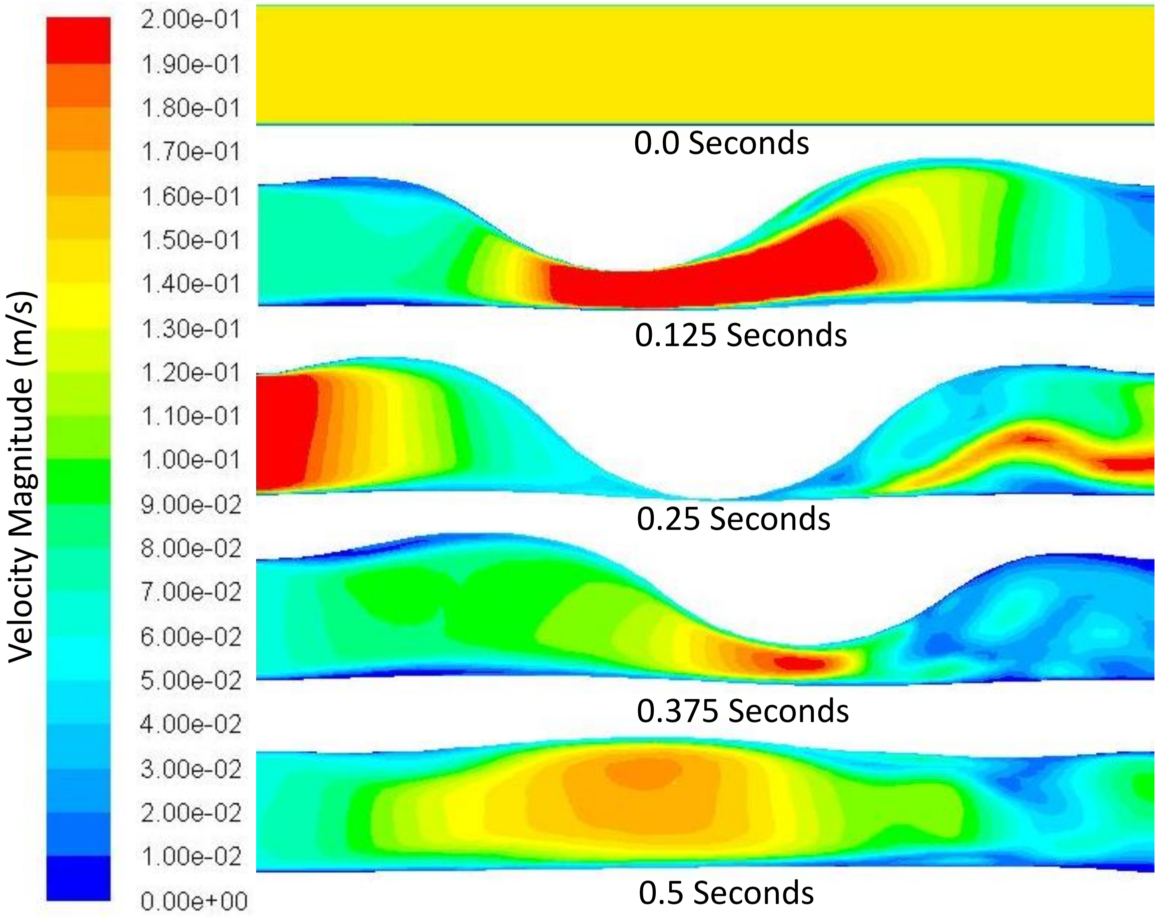


Figure 5.9: Fluid velocity contour plot at 0, 0.125, 0.25, 0.375, and 0.5 seconds.

The mass average fluid inlet and outlet temperature as a function of time was insignificant and never exceeded a temperature difference of 0.5ºC. The temperature contours of the base condition at 0, 0.125, 0.25, 0.375, and 0.5 seconds are presented in Figure 5.10. The temperature difference between the fluid inlet and outlet was small, and the reasoning for this is discussed in the section titled “Solid Results.”

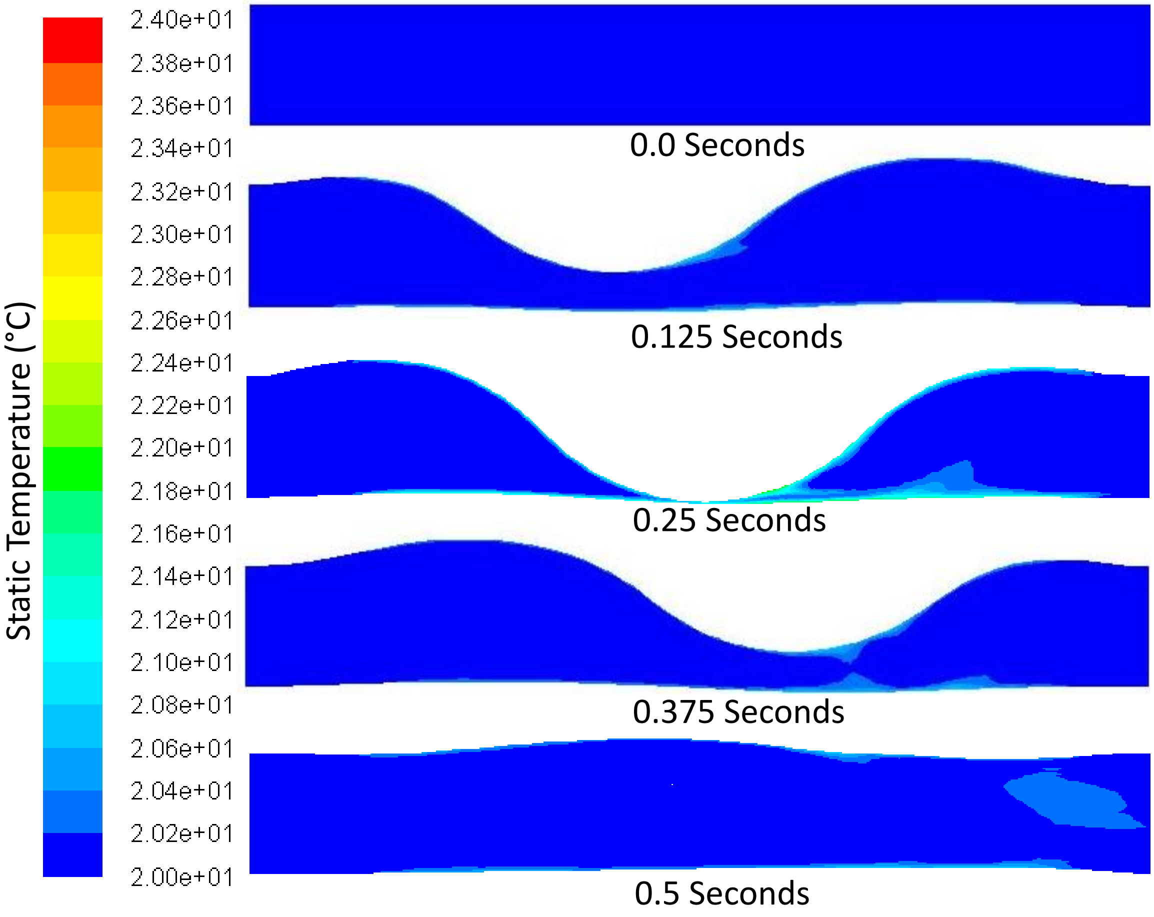


Figure 5.10: Static temperature contour in the fluid at 0, 0.125, 0.25, 0.375, and 0.5 seconds.

5.5.1.2. Solid Results

The mesh investigation for the solid domain of the FSI model indicated 60 elements were necessary along the circumference of the tubing with five elements through the tubing wall thickness. Increasing the number of cells in the circumference and thickness resulted in higher aspect ratios, while reducing the number of cells resulted in failed elements when maximum flow constriction was applied. The von Mises stress and strain is plotted as a function of time in Figure 5.11, while the maximum and minimum normal stress and strain in the constriction direction as a function of time are presented in Figure 5.12 and Figure 5.13 respectively. The maximum normal stress and strain values in the constriction direction as a function of time and contour plots are presented in Figure 5.14 and Figure 5.15. The maximum normal stress and strain as a function of time occur at 0.25 seconds. The material properties were defined using the 100% strain values stated in Patrick’s *Practical Guide for Polyvinyl Chloride* [101]. The material properties behavior between the unstressed and 100% strain data was defined linearly in the base condition. The greatest maximum normal strain experienced by the material is 0.55 m/m, indicating the material properties have been sufficiently defined over the operating range of the material. The lack of stress and strain concentrations at the end of the tube indicates the length of the tube is sufficient to not cause end effects from the boundary conditions.

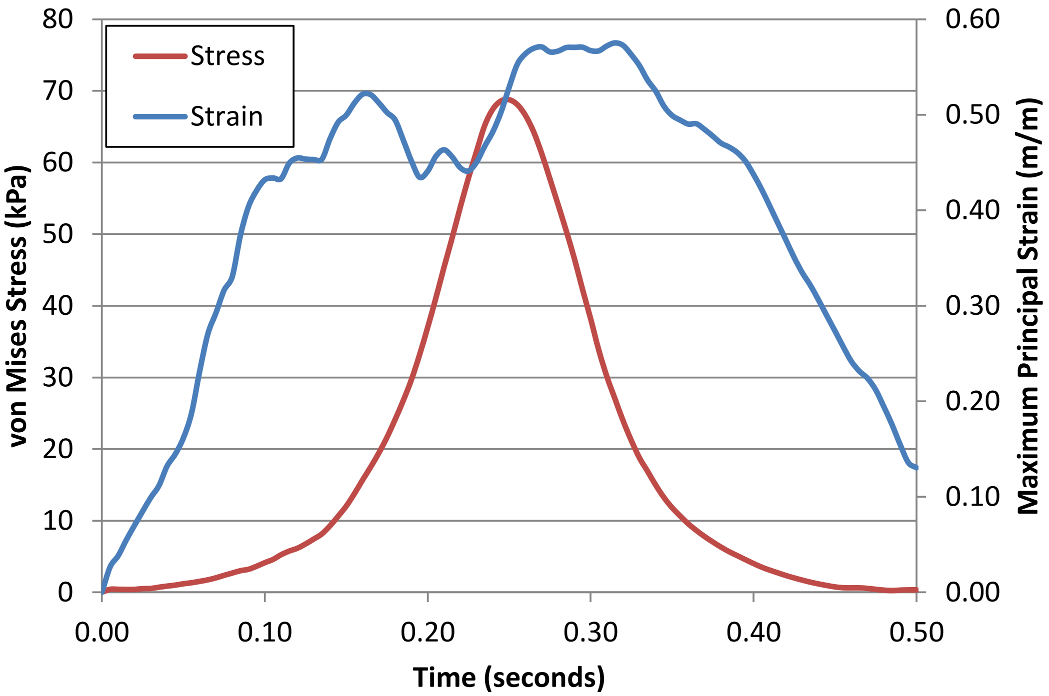


Figure 5.11: Maximum von Mises stress and maximum principal strain in the tubing as a function of time.

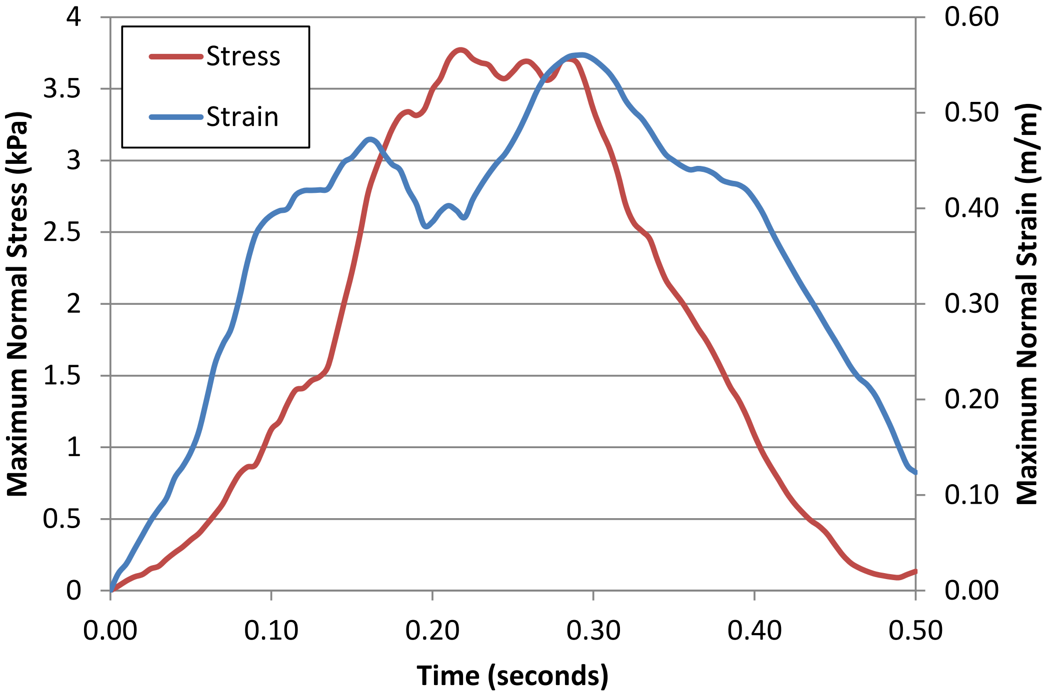


Figure 5.12: Maximum stress and strain in the constriction direction as a function of time.

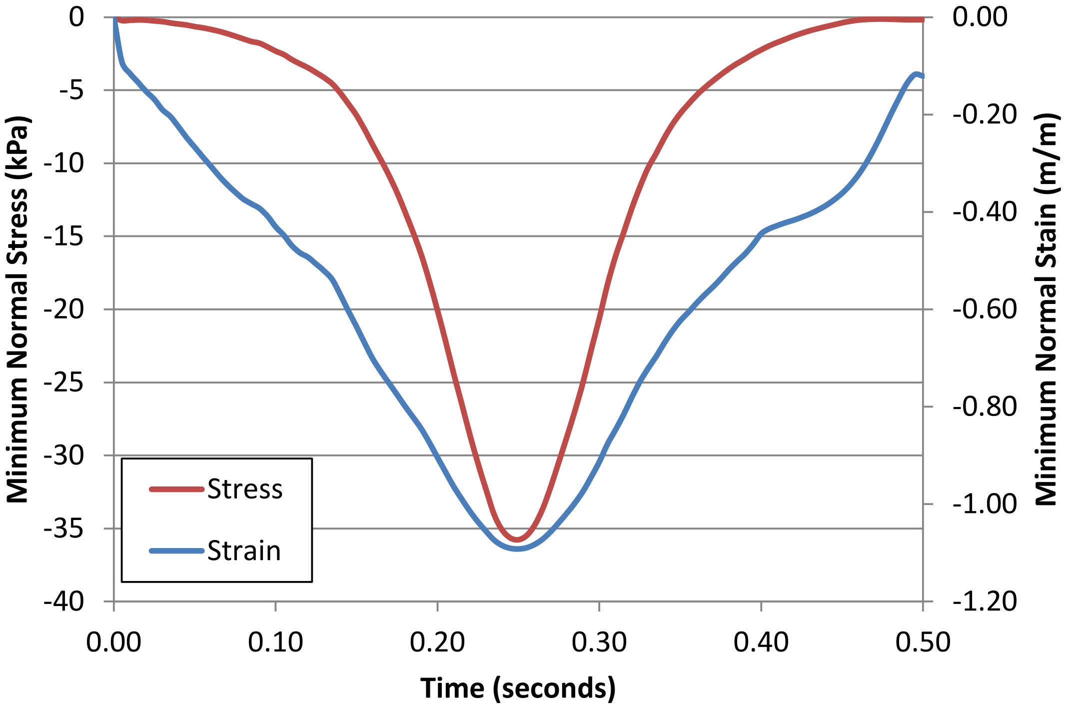


Figure 5.13: Minimum stress and strain in the constriction direction as a function of time

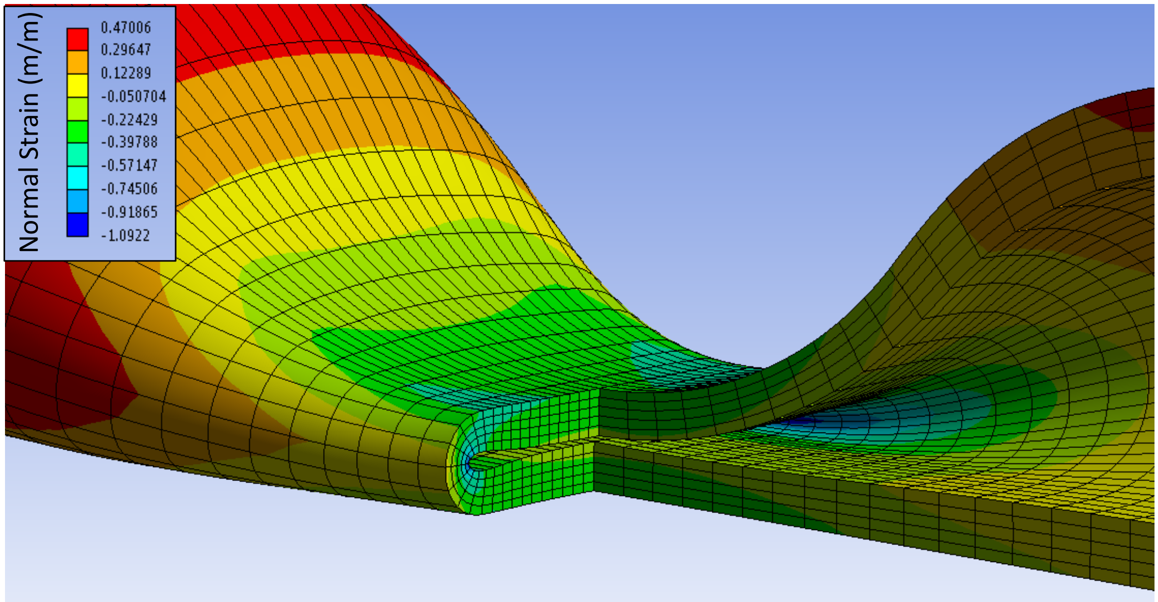


Figure 5.14: Normal strain in the constriction direction at 0.25 seconds with maximum strain of 0.47 and minimum strain of -1.09.

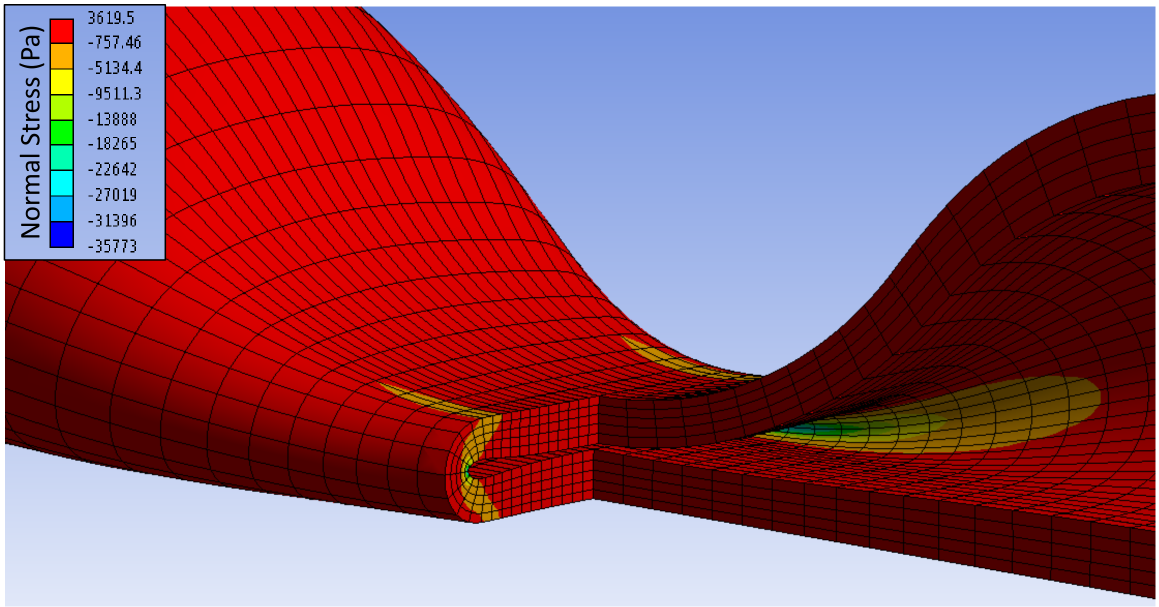


Figure 5.15: Normal stress in the constriction direction as 0.25 seconds with maximum strain of 3.62 kPa and minimum strain of -35.7 kPa.

The tube-to-base, tube-to-cam, and tube-to-tube contact regions defined in the solid domain all experienced touching between surfaces and are depicted at 0.25 seconds in Figure 5.16, Figure 5.17, and Figure 5.18, respectively. The reason the tube-to-tube contact region does not experience sliding is due to the gap that has been defined between the contact surfaces.

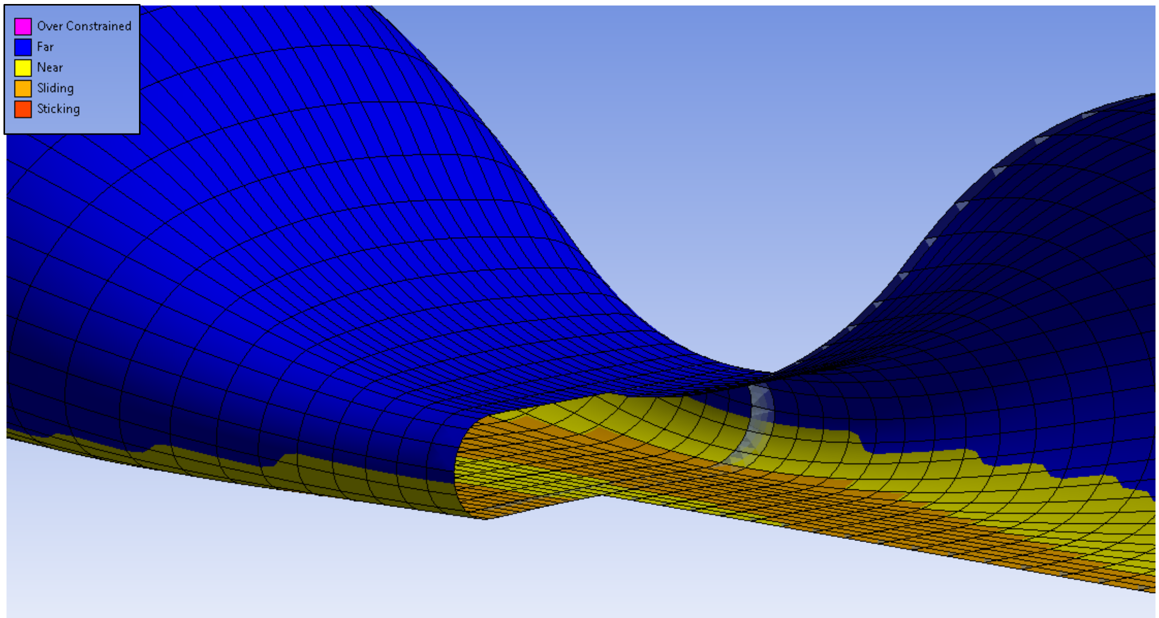


Figure 5.16: Contact status between tubing and base at 0.25 seconds.

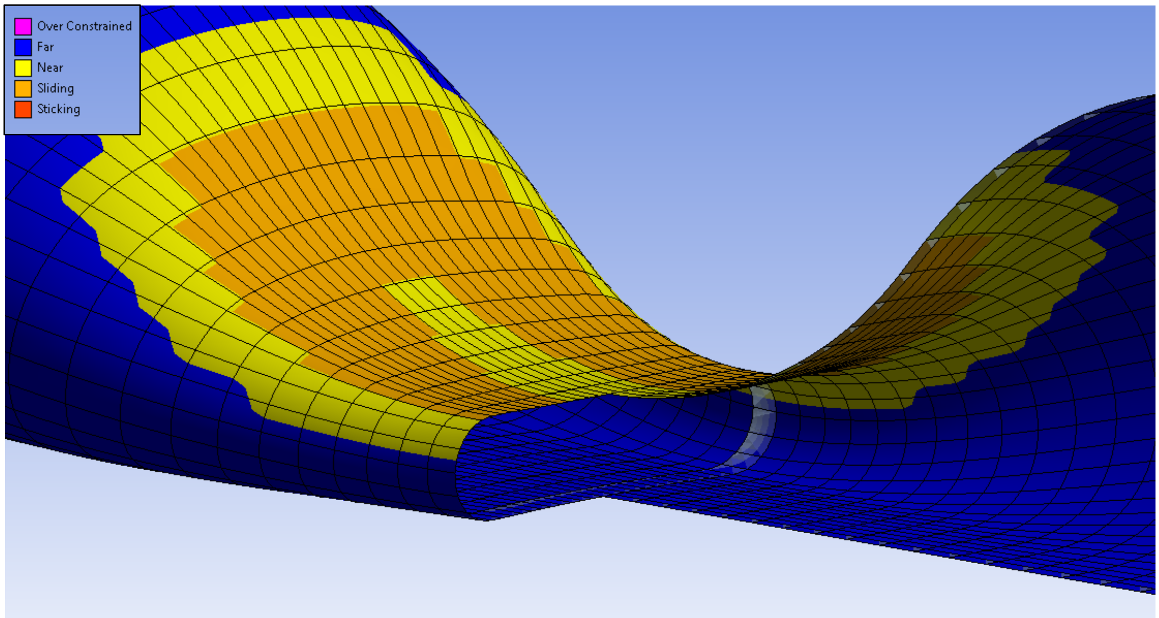


Figure 5.17: Contact status between tubing and cam at 0.25 seconds.

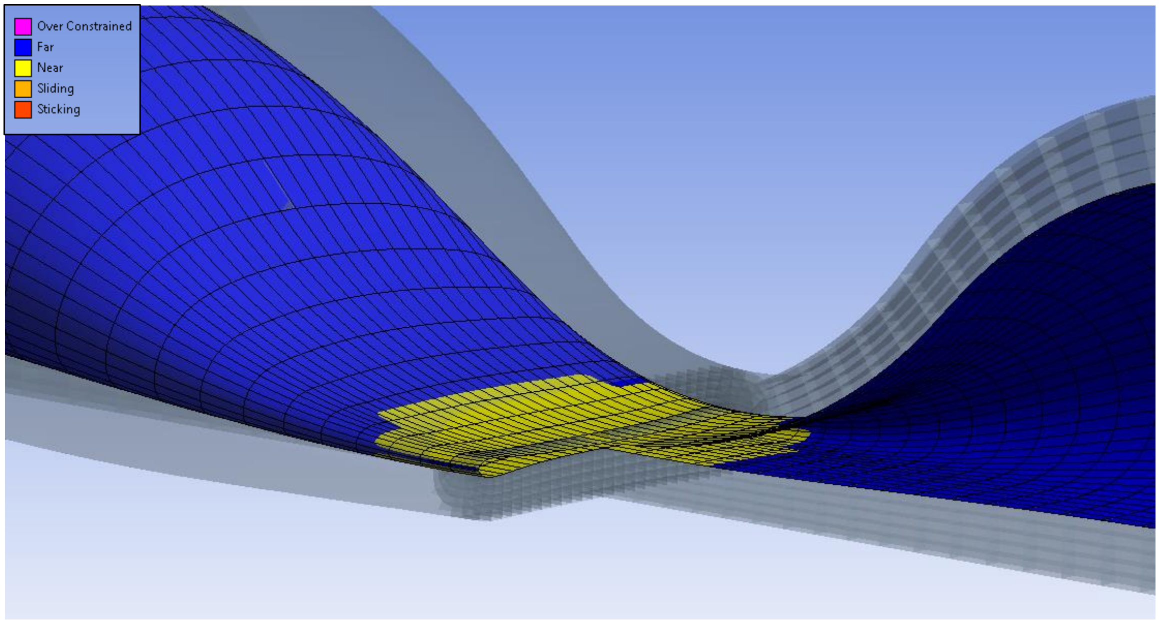


Figure 5.18: Contact status for tubing self-contact at 0.25 seconds. Status in near because 0.5 mm gap has been specified to ensure fluid domain is present.

The temperature through the thickness of the tubing is depicted in Figure 5.19. The amount of frictional heating at the contact interface is directly related to the magnitude of the contact pressure at the surface of the contact. Due to the poorly defined material properties, the contact pressure is very low, resulting in minimal frictional heating and low temperature gradients. These shallow temperature gradients are then translated to the fluid, resulting in minimal fluid heating.

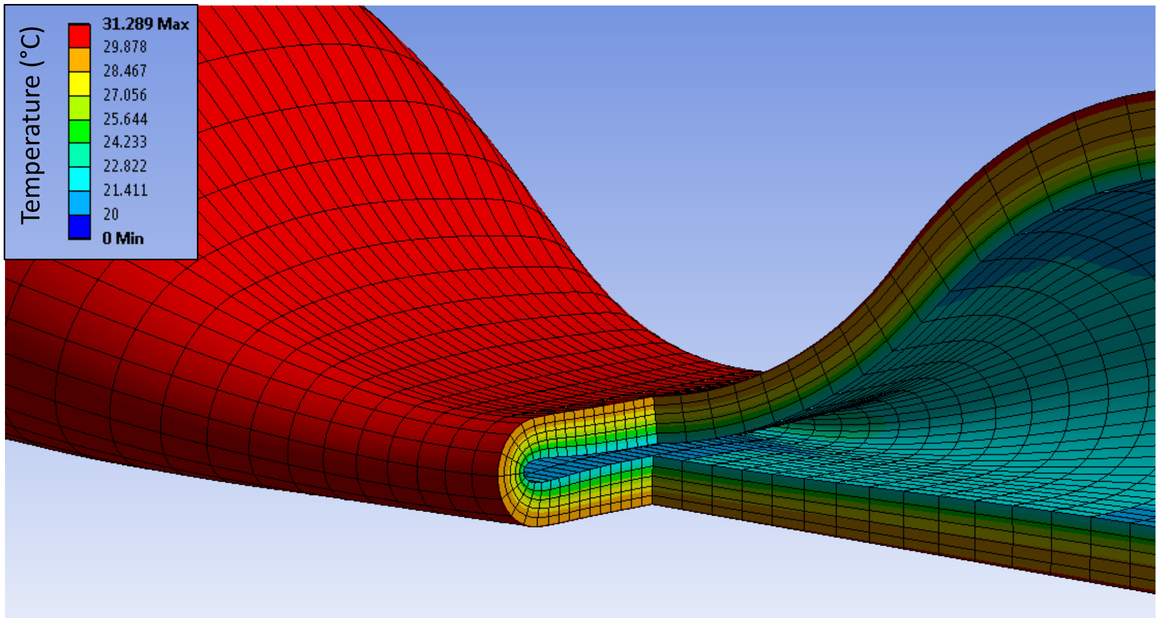


Figure 5.19: Contour of temperature at 0.25 seconds in the solid domain.

5.5.1.3. Fatigue Life Results

The location of failure occurred along the inside of the tubing where the tubing is pinched during constriction, Figure 5.20. The lifetime of the base condition and resulting designs from the parametric study have been normalized by dividing the number of cycles until failure by the minimum number of cycles until failure of the base condition. Thus the lifetime of the base condition is 1.00.

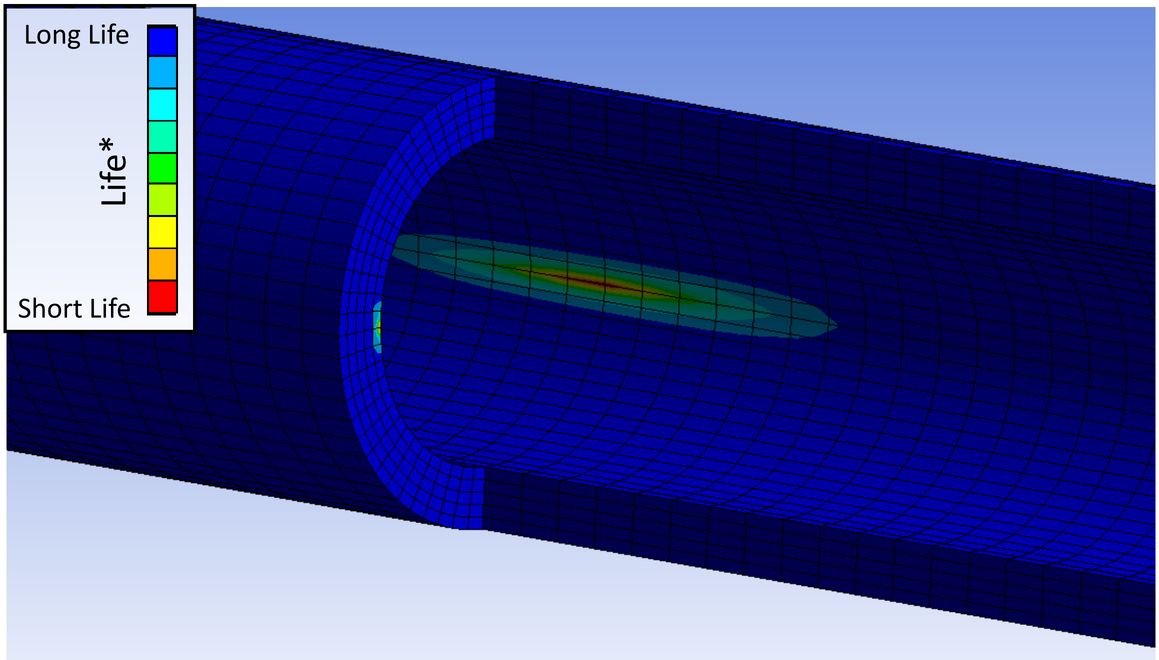


Figure 5.20: Contour of tubing life* for the base condition with shorter lifetime in red and longer lifetime in blue.

5.5.2. Parametric Study

The results from the parametric study indicated that altering the tubing geometry, tubing material properties, fluid properties, and operating temperature does have an effect on fluid flow rate, stress, strain, and/or lifetime, Table 5.8. Altering the linear material properties to the upper material bound significantly reduced the lifetime of the design, while altering the linear material properties to the lower material bound increased the lifetime by three times. Increasing the fluid viscosity had no effect of the stress and strain levels in the solid model, but did reduce the flow rate significantly. Changing the operating temperature of the fluid inside the pump had a small effect on the stress and strain levels due to the temperature-dependent solid material properties. Thus, based on the parametric study, it is desired to have tubing properties similar to the lower material bound because fluid flow rate remains the same while tubing lifetime can be improved.

Table 5.8: Results for the parametric study including flow rate, stress, strain, and lifetime.

	Design Name	Tubing Wall Thickness (mm)	Solid Material Property Behavior	Viscosity (kg/m-s)	Fluid Inlet Temperature (°C)	Flow Rate per Cycle (ml)	Lifetime (Life*)
	Base Condition	1.5	Linear	0.001003	20	2.35	1.00
Geometry	Thick Tubing	2.0	Linear	0.001003	20	1.72	3.07
Solid Material Property	Upper Material Bound	1.5	Upper bound	0.001003	20	2.28	0.05
	Lower Material Bound	1.5	Lower bound	0.010000	20	2.34	3.08
Fluid Property	Medium Viscosity	1.5	Linear	0.010000	20	1.66	1.00
	High Viscosity	1.5	Linear	0.100000	20	0.88	1.01
Operating Temperature	High Temperature	1.5	Linear	0.001003	10	2.35	0.97

5.6. Conclusion

This study presented the setup and evaluation of a thin-walled tube geometry with nonlinear and temperature-dependent material properties to create large solid deformation and fluid motion. This model allowed for tubing displacement, fluid pressures, and thermal energy to be exchanged between the fluid and solid domains. The parametric study presented in this chapter shows that an FSI model can be used as a tool to predict system lifetime performance relative to other design alterations. The analysis methods in this study are robust enough to capture the effects of solid material properties, fluid material properties, and operating temperatures on lifetime performance. Of all the design alterations explored, the largest impact on tubing lifetime performance resulted from alterations of material properties, while the largest impact on fluid flow rate was a result of altered fluid viscosity. This computational model was set up and evaluated using commercially available software and inexpensive computational resources, thus demonstrating that FSI modeling can be used as an industry-appropriate design tool.

5.7. Lessons Learned

The coupling of thermal energy between the fluid and solid domains is best achieved when the fluid domain passes near wall temperature and convective heat transfer coefficient to the solid domain and the solid domain passes wall temperature to the fluid domain. Other configurations are possible but are less stable and require significantly more computational time.

FSI problems that utilize a displacement-driven mechanism are more stable than force- or pressure-driven problems, i.e. if a problem can be simplified or defined using displacement, a more stable model may result.

CHAPTER 6: CONCLUSIONS AND RECOMMENDATIONS

This dissertation demonstrates that fluid–structure interaction (FSI) modeling can be efficiently used by industry as a design tool through utilizing inexpensive computational resources and commercially available software. Quantitative validation of FSI models can be difficult due to the challenges associated with measuring the physical systems; therefore, significant effort and care was put into the validation process of these computational models to ensure the models accurately represent the physical system they replicate. As discussed in Chapter 1 in the section titled “Computational Instabilities and Validation,” many of the physical systems where FSI modeling could be a helpful design tool are inherently unstable and therefore require the use of multiple techniques to achieve a converged solution. The insight an FSI model can provide to an engineer in the design process can be invaluable in a way that is both cost-effective and otherwise not possible with experimental testing. The studies presented in this dissertation demonstrate the ability to accurately represent physical systems using FSI modeling under various failure modes. The use of FSI modeling in the design process has a broad application basis as a practical design tool for a number of industries. All computational modeling utilizes assumptions and simplifications that limit the accuracy of the results. Knowing the extent of these assumptions and simplifications is important to understand the confidence in the agreement between the computational model and the physical system.

6.1. Novel Contributions and Limitations of Each Study

6.1.1. Study 1: FSI Flap Validation

The FSI flap validation study presented in Chapter 2 outlined the methods and techniques used to achieve a stable solution for a thin-walled fluid–structure geometry with nonlinear material properties. This study performed the first known, direct, quantitative analysis of experimental and computational results by comparing a physical experiment with FSI modeling results. The quantitative validation achieved via this dataset lays the foundation for using FSI modeling in industry by demonstrating that FSI modeling can accurately represent physical systems.

Although Study 1 produced a quantitative dataset used for validation of this FSI model, the dataset contains assumptions and is limited in scope to the steady-state material properties. During the experimental measurement of the flap deflection, a hysteresis clearly occurred within the flap material. This hysteresis was minimized by using multiple flaps and pre-fatiguing the flaps to reduce the effect of previous stressed conditions. These techniques enabled the study to produce repeatable results and avoid capturing the hysteresis in both the computational model and the experiment. This was sufficient for demonstration and validation of the FSI model during steady-state material properties conditions, but might pose a problem if evaluating the specific performance of the flap at the beginning, middle, and end of life. Modeling the hysteresis of the material can be accomplished using material properties definitions utilizing material hardening properties. Despite these limitations, the conclusions of Study 1 were unaffected because the goal of this study was to achieve a quantitative comparison of experimental and computational results and provide the methods to achieve a stable computational model. Additional experimental testing would be necessary to properly characterize the material hardening as a function of time and stress. However, once these material properties are known, they could be implemented via the methods used in Study 2, Chapter 3.

6.1.2. Study 2: FSI Model with Thermal-Cyclic Loading

The study presented in Chapter 3, which evaluated an FSI model with thermal-cyclic loading, outlined the methods and techniques used to predict fatigue life and optimize lifetime performance by modifying geometry, thermal loads, and material properties. This model is unique because it couples a thermal-driven cycle with thermal expansion, and thermal stress with fatigue life prediction. Previous published work has not linked Computational Fluid Dynamics (CFD) modeling with fatigue methods to predict lifetime. Traditionally, CFD modeling has been used to determine optimal operation conditions, but not the length of time before the thermal system fails.

Although this study demonstrated the ability to alter the fatigue life of the part, constant thermal boundary conditions and idealized material property assumptions were made that limited the depth and extent to which these design alterations can be implemented into the physical system. The thermal boundary conditions applied to the model were generated from a transient CFD model requiring 30 days to run on a high performance computer cluster. Such a substantial amount of computational time made it infeasible to generate an independent thermal boundary condition for each design alteration, so the same thermal boundary conditions were held constant for all geometric and material design alterations. Knowing these limitations is important because a change in part thickness, density or specific heat will affect the thermal mass in the systems, resulting in an altered thermal boundary condition. In order to isolate these material properties modifications as much as possible, material properties modifications were limited to only altering a single property at once. However, it is difficult, if not impossible, to alter a single material property without affecting any others. Because these assumptions were taken into account when forming the conclusions of this study, they do not change the outcome of the study. The goal of the study was to assess the sensitivity of various parameters on fatigue life and not to provide the exact number of cycles before failure. Significantly more computational time and discussions with our industry partner about their requirements regarding design alterations would be required if more detailed design work is desired.

6.1.3. Study 3: Thermal-Fluid Lifetime Design

The thermal-fluid study presented in Chapter 4 demonstrates the capability of using computational fluid dynamics to model for system lifetime. As demonstrated in Chapter 4, previous optimization work in the area of CFD modeling only pertains to optimal operating conditions and does not model the length of service life. This work demonstrated that time-dependent factors can be used not only to accurately predict life, but also as a design tool to find the optimal configuration for the lifetime of thermal-fluid systems.

Although this study demonstrated the ability to accurately predict and prolong the lifetime of a thermal system, its accuracy is limited due to inconsistencies present in the experimental methods. Extensive experimental testing was performed at the University of

Denver to understand the thermal properties of the materials within the physical system. Despite this testing, many of the experiments used to determine the material properties required significant temperature differences for accurate measurements. This required the tests to be performed over the large temperature range that the vessel experienced in operation. The most important material property in the system was the thermal conductivity, followed by the specific heat. The thermal conductivity of the materials used increases as a function of temperature; therefore, using the material properties values that were tested at elevated temperatures would make the prediction of life more conservative. Because these limitations were factored into the interpretation of the results, they do not change the conclusions of the study, but make the thermal lifetime predictions more conservative. The experimental tests performed on the vessel were performed in a thermal chamber by our industry collaborator. Significant discussion and documentation was conducted before, during, and after the experiment with our industry collaborator to reduce the discrepancies between the computational models and the experiments used for validation. Based upon this discussion and documentation, nine vessels were tested at a time in order to minimize discrepancies between results and modeling. During our discussions about the model validation with our industry collaborator, it was determined the limiting factor was the experimental accuracy of ±0.036 Θ.

6.1.4. Study 4: FSI Model of Peristaltic Pump

The peristaltic pump study presented in Chapter 5 outlined the methods and techniques used to create an FSI model containing both mechanical and thermal data transfer. This model demonstrated that a thin-walled geometry with large deformations can be modeled using nonlinear temperature-dependent material properties. Previous published work about modeling in this area has not yet incorporated thermal and mechanical data transfers into a single model. This work is unique due to the multiple data transfers involved, and presented a complex modeling challenge because it performed modeling on an unstable, thin-walled part comprised of nonlinear and temperature-dependent material properties, while experiencing large deflections. Furthermore, this single comprehensive FSI model was evaluated using commercially available software and relatively inexpensive computational resources that companies in industry could access and find cost effective as a practical solution. Previous work demonstrating this level of modeling complexity has only been performed using custom software codes on extremely powerful and expensive computers.

Although this work is novel and contributes knowledge about techniques and methods for successful FSI modeling, it is limited due to the availability of accurate and sufficient material properties. Due to the nonlinear and temperature-dependent material properties, a sufficient database was not available for the desired material properties. Furthermore, testing nonlinear and temperature-dependent material properties is tedious. Peristaltic pump tubing properties were not available, so plasticized PVC—the closest matching material for material properties that were defined as a function of temperature—was used as a substitute. Future experimental work should be conducted to determine the nonlinear behavior of peristaltic pump tubing as a function of temperature. After these properties are defined, they can be implemented in the computational model.

To demonstrate that nonlinear material properties would provide a stable converged solution, extreme nonlinear material properties were evaluated in the FSI model. Due to the missing peristaltic pump tubing material properties, the exact fatigue lifetime could not be determined.

6.2. Conclusion

The techniques and methods presented in this dissertation demonstrate how a physical system can be captured and validated in a single FSI model. Study 1 provided the methods, results, and experimental dataset for future validation of FSI models. Study 2 demonstrated that FSI models can be used as a design tool capable of accounting for geometry modification, material properties changes, and altered thermal conditions. Study 3 demonstrated the use of optimal lifetime design of a thermal-fluid system evaluated over a large domain and evaluation time. This dissertation culminated in Study 4 by combining the techniques and methods outlined in the first three studies to use a single FSI model to simulate a thin-walled part comprised of nonlinear and temperature-dependent material properties, while experiencing large deflections. As with all computational modeling, there are limitations to the accuracy and completeness of each computational model. Knowing these limitations and the effect they have on the results and conclusions directly relates to the accuracy and depth by which modeling can be used as a design tool. Future FSI modeling efforts should attempt to use the techniques and methods outlined in this dissertation, while also accounting for known limitations experienced in computational modeling.

The body of this work demonstrates the ability to perform FSI modeling using commercially available software on relatively inexpensive computational resources. This dissertation indicates that FSI modeling is a viable design tool that can be implemented in an industry setting where results must be generated in a cost-effective and time-efficient manner.

REFERENCES

[1] Souli, M., and Benson, D. J., 2010, Arbitrary Lagrangian-Eulerian and Fluid-Structure Interaction: Numerical Simulation, John Wiley & Sons, Inc.

[2] Jacobs, J., 2012, "Towards a fluid solid interaction model of a dynamic lung," University of Denver.

[3] COMSOL Inc., 2015, "COMSOL Multiphysics®."

[4] ANSYS Inc., 2013, "ANSYS User Manual."

[5] Hart, J., Peters, G., Schreurs, P., and Baaijens, F., 2000, "A two-dimensional fluid-structure interaction model of the aortic value [sic]," J. Biomech., **33**(9), pp. 1079–1088.

[6] Cheng, R., Lai, Y., and Chandran, K., 2004, "Three-dimensional Fluid-Structural Interaction Simulation of Bileaflet Mechanical Heart Valve Flow Dynamics," Ann. Biomed. Eng., **32**(11), pp. 1471–1483.

[7] Carson, R., Greer, J., and Witherell, M., 2008, "Numerical Optimization Utilizing Fluid, Thermal and Structural Modeling of the 155 mm NLOS-C Muzzle Brake," International ANSYS Conference.

[8] Banka, A., Franklin, J., Li, Z., Ferguson, B., and Aronov, M., 2008, "Applying CFD to Characterize Gear Response During Intensive Quenching Process," International ANSYS Conference.

[9] Païdoussis, M. P., 2014, Fluid-Structure Interactions: Slender Structures and Axial Flow, Academic Press.

[10] Elabbasi, N., Bergstrom, J., and Brown, S., 2011, "Fluid-Structure Interaction Analysis of a Peristaltic Pump," 2011 COMSOL Conf.

[11] Zhou, X., Liang, X., Zhao, G., Su, Y., and Wang, Y., 2014, "A New Computational Fluid Dynamics Method for In-Depth Investigation of Flow Dynamics in Roller Pump Systems," Int. Cent. Artif. Organs Transplant., **38**(7),

pp. 106–117.

[12] Ahamed, F., 2015, "The Fluid Structure Interaction Analysis of a Peristaltic Pump," Lappeenranta University of Technology.

[13] Dassault Systemes, 2012, "ABAQUS User Manual."

[14] Fraunhofer Society, 2014, "MpCCI Co-simulation."

[15] Cook, R., Malkus, D., Plesha, M., and Witt, R., 2002, Concepts and Applications of Finite Element Analysis, John Wiley & Sons, Inc.

[16] Thomson, M., and Melville, L., 1966, Theoretical Aerodynamic, Macmillan, Mineola.

[17] Hou, S., 1995, "Lattice Boltzmann method for incompressible, viscous flow," Harvard.

[18] Chen, S., and Doolen, G. D., 1998, "LATTICE BOLTZMANN METHOD FOR FLUID FLOWS," Annu. Rev. Fluid Mech., **30**, pp. 329–364.

[19] Dellar, P. J., 2003, "Incompressible limits of lattice Boltzmann equations using multiple relaxation times," J. Comput. Phys., **190**(2), pp. 351–370.

[20] Glowinski, R., Pan, T.-W., and Periaux, J., 1994, "A fictitious domain method for Dirichlet problem and applications," Comput. Methods Appl. Mech. Eng., **111**(3-4), pp. 283–303.

[21] Bertrand, F., Tanguy, P. A., and Thibault, F., 1997, "A three-dimensional fictitious domain method for incompressible fluid flow problems," Int. J. Numer. Methods Fluids, **25**(6), pp. 719–736.

[22] Loon, R., Anderson, P., Hart, J., and Baaijens, F., 2004, "A combined fictitious domain/adaptive meshing method for fluid–structure interaction in heart valves," Int. J. Numer. Methods Fluids, **46**(5), pp. 533–544.

[23] Loon, R., Anderson, P., and Vosse, F., 2006, “A fluid–structure interaction method with solid-rigid contact for heart valve dynamics,” J. Comput. Phys., **217**(2), pp. 806–823.

[24] Nobili, M., Morbiducci, U., Ponzini, R., Gaudio, C., Balducci, A., Grigion, M., Montevecchi, F., and Redaelli, A., 2008, “Numerical simulation of the dynamics of a bileaflet prosthetic heart valve using a fluid–structure interaction approach,” J. Biomech., **41**(11), pp. 2539–25550.

[25] King, M., Corden, J., David, T., and Fisher, J., 1996, “A three-dimensional, time-dependent analysis of flow through a bileaflet mechanical heart valve: Comparison of experimental and numerical results,” J. Biomech., **29**(5), pp. 609–618.

[26] King, M., David, T., and Fisher, J., 1997, “Three-dimensional study of the effect of two leaflet opening angles on the time-dependent flow through a bileaflet mechanical heart valve,” Med. Eng. Phys., **19**(3), pp. 235–241.

[27] Krafczyk, M., Cerrolaza, M., Schulz, M., and Rank, E., 1998, “Analysis of 3D transient blood flow passing through an artificial aortic valve by Lattice–Boltzmann methods,” J. Biomech., **31**(5), pp. 453–462.

[28] Farhat, C., Lesoinnea, M., and Tallecb, P. Le, 1998, “Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity,” Comput. Methods Appl. Mech. Eng., **157**(1-2), pp. 95–114.

[29] Lai, Y., Chandran, K., and Lemmon, J., 2002, “A numerical simulation of mechanical heart valve closure fluid dynamics,” J. Biomech., **35**(7), pp. 881–892.

[30] Löhner, R., Cebral, J. R., Yang, C., Baum, J. D., Mestreau, E. L., and Soto, O., 2006, “Extending the Range and Applicability of the Loose Coupling Approach for FSI Simulations,” Fluid-Structure Interact., **53**, pp. 82–100.

[31] Wall, W. A., Kuttler, U., Gerstenberger, A., Gee, M., and Forster, C., 2010, “Advances in Computational Fluid-Thin-Walled-Structure Interaction — Formulations and Solvers,” Int. Cent. Mech. Sci., **519**, pp. 175–203.

[32] Turek, S., Hron, J., Madlik, M., Razzaq, M., Wobker, H., and Acker, J., 2010, “Numerical Simulation and Benchmarking of a Monolithic Multigrid Solver for Fluid-Structure Interaction Problems with Application to Hemodynamics,” Fluid Struct. Interact., **73**, pp. 193–220.

[33] Turek, S., Hron, J., Razzaq, M., Wobker, H., and Sch, M., 2010, “Numerical Benchmarking of Fluid-Structure Interaction: A Comparison of Different Discretization and Solution Approaches,” Fluid Struct. Interact., **73**, pp. 413–424.

[34] Chakrabarti, S. K., ed., 2005, Numerical Models in Fluid-Structure Interaction, WIT Press, Southampton.

[35] Belytschko, T., Liu, W. K., and Moran, B., 2010, “Nonlinear finite elements/Lagrangian and Eulerian descriptions,” Wikiversity [Online]. Available: http://en.wikiversity.org/wiki/Nonlinear_finite_elements/Lagrangian_and_Eulerian_descriptions.

[36] Morand, H., and Ohayon, R., 1995, Fluid Structure Interaction: Applied Numerical Methods, John Wiley & Sons, Inc., Saint-Quentin.

[37] Vierendeelsa, J., Lanoyeb, L., Degrootea, J., and Verdonckb, P., 2007, “Implicit coupling of partitioned fluid–structure interaction problems with reduced order models,” Comput. Struct., **85**(11-14), pp. 970–976.

[38] Schäfer, M., Heck, M., and Yigit, S., 2006, “An Implicit Partitioned Method for the Numerical Simulation of Fluid-Structure Interaction,” Fluid-Structure Interact., **53**, pp. 171–194.

[39] CD-adapco, 2015, “STAR-CCM+.”

[40] Kunz, R. F., Chyczewski, T. S., Stinebring, D. R., and Gibeling, H. J., 1999, “MULTI-PHASE CFD ANALYSIS OF NATURAL AND VENTILATED CAVITATION ABOUT SUBMERGED BODIES,” ASME Fluids Eng. Conf., pp. 2–9.

[41] Wiggert, D., and Tijsseling, A., 2001, “fluid transients and FSI in flexible liguid-filled piping.pdf,” Appl. Mech. Rev., **54**(5), pp. 455–481.

[42] Hart, J., Peters, G., Schreurs, P., and Baaijens, F., 2002, "Fluid-structure Interaction in the Aortic Heart Valve," Eindhoven University of Technology, Department of Biomedical Engineering.

[43] Causin, P., and Fabio, J. G., 2004, "Added-mass effect in the design of partitioned algorithms for fluid-structure problems," Inst. Natl. Rech. EN Inform. EN Autom., **194**(42-44).

[44] Turek, S., and Hron, J., 2006, "Proposal for Numerical Benchmarking of Fluid-Structure Interaction between an Elastic Object and Laminar Incompressible Flow," Fluid-Structure Interact., **53**, pp. 371–385.

[45] Wick, T., 2014, "Flapping and contact FSI computations with the fluid–solid interface-tracking/interface-capturing technique and mesh adaptivity," Comput. Mech., **53**, pp. 29–43.

[46] Takizawa, K., Tezduyar, T., Buscher, A., and Asada, S., 2014, "Space–time fluid mechanics computation of heart valve models," Comput. Mech., **54**, pp. 973–987.

[47] Hsu, M.-C., Kamensky, D., Xu, F., Kiendl, J., Wang, C., Wu, M., Mineroff, J., Reali, A., Bazilevs, Y., and Sacks, M., 2015, "Dynamic and fluid–structure interaction simulations of bioprosthetic heart valves using parametric design with T-splines and Fung-type material models," Comput. Mech., **55**, pp. 1211–1225.

[48] Tian, F.-B., Dai, H., Luo, H., Doyle, J., and Rousseau, B., 2014, "Fluid–structure interaction involving large deformations: 3D simulations and applications to biological systems," J. Comput. Phys., **258**, pp. 451–469.

[49] Gundiah, N., Matthews, P. B., Karimi, R., Azadani, A., Guccione, J., Guy, T. S., Saloner, D., and Tseng, E. E., 2008, "Significant Material Property Differences Between the Porcine Ascending Aorta and Aortic Sinuses," J. Heart Valve Dis., **17**, pp. 606–613.

[50] Malecki, R., Rhie, C., McKinney, R., Ouyang, H., Syed, S., Colket, M., and Madabhushi, R., 2001, "Application of an Advanced CFD-based Analysis System to the PW6000 Combustor to Optimize Exit Temperature Distribution – Part I: Description and Validation of the Analysis Tool," ASME Turbo Expo 2001 Power Land, Sea, Air.

[51] Snyder, T., Stewart, J., Matthew, S., and Mckinney, R., 2001, “Application of an Advanced CFD-based Analysis System to the PW6000 Combustor to Optimize Exit Temperature Distribution – Part II: Comparison of Predictions to Full Annular Rig Test Data,” ASME Turbo Expo 2001 Power Land, Sea, Air.

[52] Lee, J. T., 2014, “On-Orbit Cryogenic Propellant Storage Optimization,” University of Denver.

[53] Karl, A., May, G., Barcock, C., Webster, G., and Bayley, N., 2006, “Robust design – methods and application to real work examples,” ASME Turbo Expo 2006 Power Land, Sea, Air, pp. 145–151.

[54] Sivaramakrishna, G., Narayana, V., and Kumar, S., 2014, “Role of CFD in the Design and Development of Gas Turbine Combustors – Preliminary Sizing to Performance Optimization,” 16 Annual CFD Symposium.

[55] Opgenorth, M., Sederstrom, D., McDermott, W., and Lengsfeld, C. S., 2012, “Maximizing pressure recovery using lobed nozzles in a supersonic ejector,” Appl. Therm. Eng., **37**, pp. 396–402.

[56] Opgenorth, M., McDermott, W., and Lengsfeld, C. S., 2011, “Process for Coupling Optimization and Probability,” At. Sprays, **21**(2), pp. 121–126.

[57] ANSYS Inc., “Fluent Users” [Online]. Available: www.Fluentusers.com.

[58] Xiang-hui, M., Xu-dong, D., and You-bai, X., 2009, “Time-varying Performance Prediction and System Identification of Internal Combustion Engines,” J. Shanghai Jiaotong Univ., **14**(6), pp. 701–706.

[59] Peskin, C., and McQueen, D., 1994, “Mechanical equilibrium determines the fractal fiber architecture of aortic heart valve leaflets,” Am. Physiol. Soc.

[60] Makhijani, V., Yang, H., Singhal, A. K., and Hwang, N., 1994, “An Experimental-Computational Analysis of MHV Cavitation: Effects of Leaflet Squeezing and Rebound,” J. Heart Valve Dis.

[61] Tangirala, V., Tolpadi, A., Danis, A., and Mongia, H., 2000, “Parametric

Modeling Approach to Gas Turbine Combustor Design," ASME Turbo Expo 2000 Power Land, Sea, Air.

[62] Bazilevs, Y., Hsu, M.-C., Zhang, Y., W, W., T, K., S, H., and J, I., 2010, "Computational vascular fluid–structure interaction: methodology and application to cerebral aneurysms," Biomech. Model. Mechanobiol., **9**, pp. 481–498.

[63] Muha, B., and Canic, S., 2013, "Existence of a Weak Solution to a Nonlinear Fluid–Structure Interaction Problem Modeling the Flow of an Incompressible, Viscous Fluid in a Cylinder with Deformable Walls," Arch. Ration. Mech. Anal., **207**, pp. 919–968.

[64] Hsu, M.-C., Kamensky, D., Bazilevs, Y., Sacks, M. S., and Hughes, T., 2014, "Fluid–structure interaction analysis of bioprosthetic heart valves: significance of arterial wall deformation," Comput. Mech., **54**, pp. 1055–1071.

[65] Hsu, M.-C., and Bazilevs, Y., 2012, "Fluid–structure interaction modeling of wind turbines: simulating the full machine," Comput. Mech., **50**, pp. 821–833.

[66] Bazilevs, Y., Hsu, M.-C., Kiendl, J., R, W., and K, B., 2011, "3D simulation of wind turbine rotors at full scale. Part II: Fluid–structure interaction modeling with composite blades," Int. J. Numer. Methods Fluids, **65**, pp. 236–253.

[67] Takizawa, K., Moorman, C., Wright, S., Spielman, T., and Tezduyar, T., 2011, "Fluid–structure interaction modeling and performance analysis of the Orion spacecraft parachutes," Int. J. Numer. METHODS FLUIDS2, **65**, pp. 271–285.

[68] Takizawa, K., and Tezduyar, T., 2012, "Computational Methods for Parachute Fluid–Structure Interactions," Arch. Comput. Methods Eng., **19**, pp. 125–169.

[69] Takizawa, K., Spielman, T., Moorman, C., and Tezduyar, T., 2012, "Fluid-Structure Interaction Modeling of Spacecraft Parachutes for Simulation-Based Design," J. Appl. Mech., **79**.

[70] Takizawa, K., Wright, S., Moorman, C., and Tezduyar, T., 2011, "Fluid–structure interaction modeling of parachute clusters," Int. J. Numer. METHODS FLUIDS, **65**, pp. 286–307.

[71] Takizawa, K., Tezduyar, T., Buscher, A., and Asada, S., 2014, “Space–time interface-tracking with topology change (ST-TC),” Comput. Mech., **54**, pp. 955–971.

[72] Takizawa, K., Fritze, M., Montes, D., Spielman, T., and Tezduyar, T. E., 2012, “Fluid–structure interaction modeling of ringsail parachutes with disreefing and modified geometric porosity,” Comput. Mech., **50**, pp. 835–854.

[73] Tezduyar, T. E., Takizawa, K., Brummer, T., and Chen, P. R., 2011, “Space – time fluid – structure interaction modeling of patient-specific cerebral aneurysms,” Int. j. numer. method. biomed. eng., **27**(11), pp. 1665–1710.

[74] Przybylski, R., “Canola Oil: Physical and Chemical Properties,” Canola Counc. Canada.

[75] Norton, R., 2006, Machine Design: An Integrated Approach, Pearson Prentice Hall, Upper Saddle River.

[76] Kaplan, W. A., 1998, Modern Plastics Encyclopedia, McGraw-Hill Book Company, New York, NY.

[77] “Mat Web - Material Property Data” [Online]. Available: http://www.matweb.com/.

[78] MathWorks, 2013, “MatLab.”

[79] MatWeb, “Overview of materials for Very Low Density Polyethylene (VLDPE)” [Online]. Available: http://www.matweb.com/search/DataSheet.aspx?MatGUID=e8da5fa318ea4eafaa33fb386457e999&ckck=1. [Accessed: 23-Mar-2016].

[80] Dunn, D., “Induced Stress in a Constrained Bar,” p. 8.

[81] Alloys, R., 2000, “RA253 MA Data Sheet.”

[82] Stephens, R., Fatemi, A., Stephens, R., and Fuchs, H., 2001, Metal Fatigue in Engineering, John Wiley & Sons, Inc., New York, NY.

[83] Zhang, Z., 2013, “Cyclic Hardening/Softening,” Encycl. Tribol., pp. 687–691.

[84] Winter, J., and Marvalaud Incorporated, 1975, The Material Properties of Gelatin Gels.

[85] Jacobs, J., Tripp, J., Underwood, D., and Lengsfeld, C. S., 2013, “Optimization of Micro-Textured Surfaces for Turbine Vane Impingement Cooling,” ASME Turbo Expo 2013 Turbine Tech. Conf. Expo., **3A**.

[86] Manning, M. C., Chou, D. K., Murphy, B. M., Payne, R. W., and Katayama, D. S., 2010, “Stability of Protein Pharmaceuticals: An Update.,” Pharm. Res., **27**(4), pp. 544–75.

[87] Chi, E. Y., Krishnan, S., Randolph, T. W., and Carpenter, J. F., 2003, “Physical Stability of Proteins in Aqueous Solution: Mechanism and Driving Forces in Nonnative Protein Aggregation.,” Pharm. Res., **20**(9), pp. 1325–36.

[88] Muhlack, R. A., O’Neill, B. K., Waters, E. J., and Colby, C. B., 2016, “Optimal Conditions for Controlling Haze-Forming Wine Protein with Bentonite Treatment: Investigation of Matrix Effects and Interactions Using a Factorial Design,” Food Bioprocess Technol., pp. 1–8.

[89] Rollero, S., With, A., INRA, U., SAS, L., Bloem, A., Camarasa, C., Sanchez, I., Ortiz-Julien, A., Sablayrolles, J.-M., Dequin, S., and Mouret, J.-R., 2014, “Combined effects of nutrients and temperature on the production of fermentative aromas by Saccharomyces cerevisiae during wine fermentation,” Appl. Microb. cell Physiol., **99**(5), pp. 2291–2304.

[90] ToolBox, E., “Polyurethane Insulation” [Online]. Available: http://www.engineeringtoolbox.com/polyurethane-insulation-k-values-d_1174.html. [Accessed: 15-Sep-2015].

[91] MatWeb, “Saint-Gobain Korel® K60 Firm Micro-cellular Polyurethane Foam” [Online]. Available: http://www.matweb.com/search/DataSheet.aspx?MatGUID=ea04d16cd064466cb43e50c2495b7e79. [Accessed: 15-Sep-2015].

[92] MatWeb, “Overview of materials for Thermoset Polyurethane Foam, Unreinforced” [Online]. Available: http://www.matweb.com/search/DataSheet.aspx?MatGUID=91d44cae736e4b36bcba94720654eeae&ckck=1. [Accessed: 15-Sep-2015].

[93] “Thermal Conductivity,” Univ. Phys. 7th Ed. [Online]. Available: http://hyperphysics.phy-astr.gsu.edu/hbase/tables/thrcn.html. [Accessed: 15-Sep-2015].

[94] Federation of European Rigid Polyurethane Foam Association, 2006, Thermal insulation materials made of rigid polyurethane foam, Brussels, Belbium.

[95] Jarfelt, U., and Ramnas, O., 2006, “Thermal Conductivity of Polyurethane Foam - Best Performance,” 10th Int. Symp. Dist. Heat. Cool.

[96] Bergman, T., Lavine, A., Incropera, F., and Dewitt, D., 2011, Heat and Mass Transfer, John Wiley & Sons, Inc.

[97] Tritton, D. J., 1977, Physical Fluid Dynamics, Clarendon Press.

[98] Farid, M. ., Hamad, F. ., and Abu-Arabi, M., 1998, “Phase change cool storage using dimethyl-sulfoxide,” Energy Convers. Manag., **39**(8), pp. 819–826.

[99] Gin, B., Farid, M., and Bansal, P., 2010, “Modelling of Phase Change Material Implemented Into Cold Storage Application,” Int. High Perform. Build. Conf.

[100] Farid, M., Khudhair, A. M., Razack, S., and Al-Hallaj, S., 2004, “A review on phase change energy storage: materials and applications,” Energy Convers. Manag., **45**(9-10), pp. 1597–1615.

[101] Patrick, S. G., 2005, “Testing and Properties,” Practical Guide to Polyvinyl Chloride, Rapra Techniology Limited, pp. 53–59.

[102] Fluid Metering Inc., “Comparion between FMI Technology and peristaltic Pump Technology.”

[103] Verder International, 2016, “Verderflex: Features and benefits of peristaltic

pumps" [Online]. Available: http://www.verderflex.com/en/features-and-benefits/.

[104] Cole-Parmer, "Masterflex Tubing and General Technical Data."

[105] Saller, V., Matilainen, J., Grauschopf, U., Bechtold-peters, K., Mahler, H.-C., and Friess, W., 2014, "Particle Shedding from Peristaltic Pump Tubing in Biopharmaceutical Drug Product Manufacturing," J. Pharm. Sci., **104**, pp. 1440–1450.

[106] Randolph, T., and Schiltz, E., 2012, "EFFECT OF MECHANICAL SHOCK ON PROTEIN AGGREGATION AND PARTICLE FORMATION," Workshop on Protien Aggregation and Immungicity2, Beaver Run, CO.

[107] Watson-Marlow Bredel Pump Academy, "Tubing Failure Modes."

[108] Colas, A., Malczewski, R., and K, U., 2004, "Silicone tubing for Pharmaceutical Processing," Dow Corning, Life Sci.

[109] 2016, Standard Test Methods for Rubber Property — Compression Set.

[110] 2016, Standard Test Methods for Vulcanized Rubber and Thermoplastic Elastomers–Tension.

[111] Compression Polymers Corp - Vycom, "CPC - Vycom FLAMETEC^{TM} KYTEC® PVDF," Matweb [Online]. Available: http://www.matweb.com/search/DataSheet.aspx?MatGUID=e349a7c8d8e549ebba4ba9fe7b5d5640. [Accessed: 29-May-2016].

[112] Chevron, "ChevronTexaco Supreme SAE 10W-40 Motor Oil," MatWeb [Online]. Available: http://www.matweb.com/search/DataSheet.aspx?MatGUID=4470f9cfaac24720a8847fe8f9c455f5.

APPENDICES

Appendix A: Matlab Optimization Scripts for Study 1

```
%create global variable
clear
clc
global gnum
gnum = 1;

%define bounds and options for optimisation
lb=[8.5,  5];
x0=[10,  10];
ub=[15,  15];
options = optimset('DiffMaxChange', 0.1,'DiffMinChange',
0.01, 'display', 'iter', 'TolFun', 1e-2);
[x,fval] =
fmincon(@Run_ANSYS_oil_non_linear_errorbars,x0,[],[],[],[],
lb,ub,[],options)

%%send text message when done
    text=sprintf('Oil optimization Complete - %i steps',
gnum-1);
    send_text_message('503-476-4311', 'verizon',text)

%evaluate FSI model and determine performance
function error = Run_ANSYS_oil_non_linear_errorbars(x0)
global gnum youngs poi;
time = 0.1;
error=0;

%create folder and go into the new folder
dir='C:\Donn_Ansys\FSI_Benchmark\oil_opt\';
cd(dir);
file='Oil40-';
filename = sprintf('%s%0.4i', file, gnum);

%test to see if there is a performance file in the folder
filech='\perform.txt';
check = exist(sprintf('%s%s%s', dir, filename, filech));

%if the file does exist read the file and return the
performance peramiter
if check;
```

```
    cd(filename);
    error_hold=dlmread(filech, '\t');
    error=error_hold;

%if it does not exist
else
mkdir(filename);
cd(filename);

%copy the Gambit and Fluent journel files
copyfile('C:\Donn_Ansys\FSI_Benchmark\oil_opt\set_material_
time_run-non_linear-tall-no_support-
same.wbjn','set_material_time_run.wbjn');
copyfile('C:\Donn_Ansys\FSI_Benchmark\oil_opt\replaceinfile
.m','replaceinfile.m');

%define variable that will determin the geometry
youngs=x0(1)*1.32e6;
poi=x0(2)*0.0000034;
new_youngs = sprintf('%i', youngs);
new_poi = sprintf('%i', poi);
new_time = sprintf('%d', time);
new_file ='holding_file.wbpj';

%replace the old values with the new one's
replaceinfile( 'oldyoungs', new_youngs,
'set_material_time_run.wbjn', '-nobak');
replaceinfile( 'oldpoi', new_poi,
'set_material_time_run.wbjn', '-nobak');
replaceinfile( 'oldtime', new_time,
'set_material_time_run.wbjn', '-nobak');
replaceinfile( 'oldaddress', new_file,
'set_material_time_run.wbjn', '-nobak');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%
    %%run workbench
    command='runwb2 -b -r';
    file ='\set_material_time_run.wbjn';
    funfilef=sprintf('%s%s%s', dir, filename, file);
    runwb=sprintf('%s %s', command, funfilef);
    dos(runwb);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%
```

```
%%results reading and processing
fluid_height = 0;
while fluid_height<0.0300
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%
%%rerun FSI if fluid level still needs to rise

    %%increase model end time
    time=time+0.01;
    new_time = sprintf('%d', time);

copyfile('C:\Donn_Ansys\FSI_Benchmark\oil_opt\set_time_run-
non_linear.wbjn','set_time_run.wbjn');
    replaceinfile( 'oldtime', new_time,
'set_time_run.wbjn', '-nobak');
    replaceinfile( 'oldaddress', filename,
'set_time_run.wbjn', '-nobak');

    %%run workbench
    command='runwb2 -b -r';
    file ='\set_time_run.wbjn';
    funfilef=sprintf('%s%s%s', dir, filename, file);
    runwb=sprintf('%s %s', command, funfilef);
    dos(runwb);

%%read input file into table (x, y, z, water-vof)
    file_name='\profile_fluid_output';
    fluent_folder='\holding_file_files\dp0\FFF\Fluent';
    full_name = sprintf('%s%s%s%s', dir, filename,
fluent_folder, file_name);
    profile=dlmread(full_name, ',',5,1);
    profile_size=size(profile);
    profile_size=profile_size(1);

%%get average fluid height from output file
    height_hold=0;
    fluid_count=0;
    i=1;
    while i<profile_size
        water_vof=profile(i,4);
        if (water_vof>0.1) && (water_vof<0.8)
            fluid_height=profile(i,2);
            height_hold=fluid_height+height_hold;
            fluid_count=fluid_count+1;
        end
```

```
        i = i+1;
    end
    fluid_height=height_hold/fluid_count;

%%write file for monitoring
    monitor1=[time fluid_height];
    dlmwrite('monitor.txt', monitor1, 'delimiter', '\t', '-
append');


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%
%%measure line deflection data
%%measure fluid deflection
        %%get deflection point 30 mm
        def30_hold=0;
        def30_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.028) && (height<0.032)
                def30=profile(i,1);
                def30_hold=def30_hold+def30;
                def30_count=def30_count+1;
            end
            i = i+1;
        end
        def30=def30_hold/def30_count-0.015;
        %%get deflection point 50 mm
        def50_hold=0;
        def50_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.048) && (height<0.052)
                def50=profile(i,1);
                def50_hold=def50_hold+def50;
                def50_count=def50_count+1;
            end
            i = i+1;
        end
        def50=def50_hold/def50_count-0.015;
        %%get deflection point 70 mm
        def70_hold=0;
        def70_count=0;
```

```
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.068) && (height<0.072)
                def70=profile(i,1);
                def70_hold=def70_hold+def70;
                def70_count=def70_count+1;
            end
            i = i+1;
        end
        def70=def70_hold/def70_count-0.015;
        %%place deflection data in matrix
        def(1,1)=def30;
        def(1,2)=def50;
        def(1,3)=def70;

        %% write data to file
        deflection=[time fluid_height def30 def50 def70];
        dlmwrite('line_deflection.txt', deflection,
'delimiter', '\t', '-append');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%

%%determine if fluid_height is at 20mm
    if (fluid_height>0.0195) && (fluid_height<0.02001)
        fluid_h20=fluid_height;
        %%get deflection data at 3 points
        %%get deflection point 30 mm
        def30_hold=0;
        def30_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.028) && (height<0.032)
                def30=profile(i,1);
                def30_hold=def30_hold+def30;
                def30_count=def30_count+1;
            end
            i = i+1;
        end
        def30=def30_hold/def30_count-0.015;
        %%get deflection point 50 mm
        def50_hold=0;
        def50_count=0;
        i=1;
```

```
        while i<1536
            height=profile(i,2);
            if (height>0.048) && (height<0.052)
                def50=profile(i,1);
                def50_hold=def50_hold+def50;
                def50_count=def50_count+1;
            end
            i = i+1;
        end
        def50=def50_hold/def50_count-0.015;
        %%get deflection point 70 mm
        def70_hold=0;
        def70_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.068) && (height<0.072)
                def70=profile(i,1);
                def70_hold=def70_hold+def70;
                def70_count=def70_count+1;
            end
            i = i+1;
        end
        def70=def70_hold/def70_count-0.015;
        %%place deflection data in matrix
        def(1,1)=def30;
        def(1,2)=def50;
        def(1,3)=def70;

        %% write data to file
        deflection=[time fluid_height def30 def50 def70];
        dlmwrite('monitor_deflection.txt', deflection,
'delimiter', '\t', '-append');
    end

%%determine if fluid_height is at 22.5mm
    if (fluid_height>0.022) && (fluid_height<0.02251)
        fluid_h22=fluid_height;
        %%get deflection data at 3 points
        %%get deflection point 30 mm
        def30_hold=0;
        def30_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
```

```
            if (height>0.028) && (height<0.032)
                def30=profile(i,1);
                def30_hold=def30_hold+def30;
                def30_count=def30_count+1;
            end
            i = i+1;
        end
        def30=def30_hold/def30_count-0.015;
        %%get deflection point 50 mm
        def50_hold=0;
        def50_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.048) && (height<0.052)
                def50=profile(i,1);
                def50_hold=def50_hold+def50;
                def50_count=def50_count+1;
            end
            i = i+1;
        end
        def50=def50_hold/def50_count-0.015;
        %%get deflection point 70 mm
        def70_hold=0;
        def70_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.068) && (height<0.072)
                def70=profile(i,1);
                def70_hold=def70_hold+def70;
                def70_count=def70_count+1;
            end
            i = i+1;
        end
        def70=def70_hold/def70_count-0.015;
        %%place deflection data in matrix
        def(2,1)=def30;
        def(2,2)=def50;
        def(2,3)=def70;
        %% write data to file
        deflection=[time fluid_height def30 def50 def70];
        dlmwrite('monitor_deflection.txt', deflection,
'delimiter', '\t', '-append');
    end
```

```
%%determine if fluid_height is at 25mm
    if (fluid_height>0.0245) && (fluid_height<0.02501)
        fluid_h25=fluid_height;
        %%get deflection data at 3 points
        %%get deflection point 30 mm
        def30_hold=0;
        def30_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.028) && (height<0.032)
                def30=profile(i,1);
                def30_hold=def30_hold+def30;
                def30_count=def30_count+1;
            end
            i = i+1;
        end
        def30=def30_hold/def30_count-0.015;
        %%get deflection point 50 mm
        def50_hold=0;
        def50_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.048) && (height<0.052)
                def50=profile(i,1);
                def50_hold=def50_hold+def50;
                def50_count=def50_count+1;
            end
            i = i+1;
        end
        def50=def50_hold/def50_count-0.015;
        %%get deflection point 70 mm
        def70_hold=0;
        def70_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.068) && (height<0.072)
                def70=profile(i,1);
                def70_hold=def70_hold+def70;
                def70_count=def70_count+1;
            end
            i = i+1;
```

```
        end
        def70=def70_hold/def70_count-0.015;
        %%place deflection data in matrix
        def(3,1)=def30;
        def(3,2)=def50;
        def(3,3)=def70;
        %% write data to file
        deflection=[time fluid_height def30 def50 def70];
        dlmwrite('monitor_deflection.txt', deflection,
'delimiter', '\t', '-append');
    end

%%determine if fluid_height is at 27.5mm
    if (fluid_height>0.0265) && (fluid_height<0.02751)
        fluid_h27=fluid_height;
        %%get deflection data at 3 points
        %%get deflection point 30 mm
        def30_hold=0;
        def30_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.028) && (height<0.032)
                def30=profile(i,1);
                def30_hold=def30_hold+def30;
                def30_count=def30_count+1;
            end
            i = i+1;
        end
        def30=def30_hold/def30_count-0.015;
        %%get deflection point 50 mm
        def50_hold=0;
        def50_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.048) && (height<0.052)
                def50=profile(i,1);
                def50_hold=def50_hold+def50;
                def50_count=def50_count+1;
            end
            i = i+1;
        end
        def50=def50_hold/def50_count-0.015;
        %%get deflection point 70 mm
```

```
        def70_hold=0;
        def70_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.068) && (height<0.072)
                def70=profile(i,1);
                def70_hold=def70_hold+def70;
                def70_count=def70_count+1;
            end
            i = i+1;
        end
        def70=def70_hold/def70_count-0.015;
        %%place deflection data in matrix
        def(4,1)=def30;
        def(4,2)=def50;
        def(4,3)=def70;
        %% write data to file
        deflection=[time fluid_height def30 def50 def70];
        dlmwrite('monitor_deflection.txt', deflection,
'delimiter', '\t', '-append');
    end

    %%determine if fluid_height is at 30.0mm
    if (fluid_height>0.029) && (fluid_height<0.030)
        fluid_h30=fluid_height;
        %%get deflection data at 3 points
        %%get deflection point 30 mm
        def30_hold=0;
        def30_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.028) && (height<0.032)
                def30=profile(i,1);
                def30_hold=def30_hold+def30;
                def30_count=def30_count+1;
            end
            i = i+1;
        end
        def30=def30_hold/def30_count-0.015;
        %%get deflection point 50 mm
        def50_hold=0;
        def50_count=0;
        i=1;
```

```
        while i<1536
            height=profile(i,2);
            if (height>0.048) && (height<0.052)
                def50=profile(i,1);
                def50_hold=def50_hold+def50;
                def50_count=def50_count+1;
            end
            i = i+1;
        end
        def50=def50_hold/def50_count-0.015;
        %%get deflection point 70 mm
        def70_hold=0;
        def70_count=0;
        i=1;
        while i<1536
            height=profile(i,2);
            if (height>0.068) && (height<0.072)
                def70=profile(i,1);
                def70_hold=def70_hold+def70;
                def70_count=def70_count+1;
            end
            i = i+1;
        end
        def70=def70_hold/def70_count-0.015;
        %%place deflection data in matrix
        def(5,1)=def30;
        def(5,2)=def50;
        def(5,3)=def70;
        %% write data to file
        deflection=[time fluid_height def30 def50 def70];
        dlmwrite('monitor_deflection.txt', deflection,
'delimiter', '\t', '-append');
    end

end

%%exit if fluid_height is over 30mm
    %%calculate error/performance perameter
    exp_lower=[0.0000 0.00011 0.00045; ...
            0.00000 0.00136 0.00183; ...
            0.00114 0.00198 0.00389; ...
            0.00187 0.00448 0.00738; ...
            0.00453 0.00925 0.01366];
    exp_upper=[0.001 0.00111 0.00145; ...
            0.00200 0.00236 0.00371; ...
```

```
                0.00214 0.00325 0.00543; ...
                0.00313 0.00552 0.00853; ...
                0.00547 0.01052 0.01657];

        a=1;
        b=1;
        while a<=5
            while b<=3
                def_exp(a,b) = def(a,b);
                if def(a,b) < exp_lower(a,b)
                    def_exp(a,b) = exp_lower(a,b);
                end
                if def(a,b) > exp_upper(a,b)
                    def_exp(a,b) = exp_upper(a,b);
                end
                b=b+1;
            end
            b=1;
            a=a+1;
        end

        a=1;
        b=1;
        while a<=5
            while b<=3
                error=(def_exp(a,b)-def(a,b))^2 + error;
                b=b+1;
            end
            b=1;
            a=a+1;
        end

    %%scall error by 1,000,000
    error = error * 10000000;

    %%write data to file
        all=[gnum youngs poi error];
        perform=error;
        deflection=[fluid_h20 def(1,1) def(1,2) def(1,3) ...
                    fluid_h22 def(2,1) def(2,2) def(2,3) ...
                    fluid_h25 def(3,1) def(3,2) def(3,3) ...
                    fluid_h27 def(4,1) def(4,2) def(4,3) ...
                    fluid_h30 def(5,1) def(5,2) def(5,3)];
        dlmwrite('deflection_data.txt', deflection,
    'delimiter', '\t');
```

```
    dlmwrite('perform.txt', perform, 'delimiter', '\t');
    cd(dir);
    dlmwrite('run_info.txt', all, 'delimiter', '\t', '-
append');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%
end
cd(dir);
gnum=gnum+1;
```

Appendix B: Fluent User-Defined Functions for Study 2

```
#include "udf.h"
#include <stdio.h>

float time_scale[44500], scaling_table[44500];
float time_scale_hold, scaling_table_hold;
float solid_exterior2plusx[35], solid_exterior2plus0[35],
solid_exterior2plus6[35], solid_exterior2plus10[35];
float solid_exterior2plusx_hold, solid_exterior2plus0_hold,
solid_exterior2plus6_hold, solid_exterior2plus10_hold;
float solid_exterior2minusx[162],
solid_exterior2minus0[162], solid_exterior2minus6[162],
solid_exterior2minus10[162];
float solid_exterior2minusx_hold,
solid_exterior2minus0_hold, solid_exterior2minus6_hold,
solid_exterior2minus10_hold;
float solid_fixedx[200], solid_fixed0[200],
solid_fixed6[200], solid_fixed10[200];
float solid_fixedx_hold, solid_fixed0_hold,
solid_fixed6_hold, solid_fixed10_hold;
float fluid_interfaceplusx[165], fluid_interfaceplus0[165],
fluid_interfaceplus6[165], fluid_interfaceplus10[165];
float fluid_interfaceplusx_hold, fluid_interfaceplus0_hold,
fluid_interfaceplus6_hold, fluid_interfaceplus10_hold;
float fluid_interfaceminusx[165],
fluid_interfaceminus0[165], fluid_interfaceminus6[165],
fluid_interfaceminus10[165];
float fluid_interfaceminusx_hold,
fluid_interfaceminus0_hold, fluid_interfaceminus6_hold,
fluid_interfaceminus10_hold;
float time_hold;

char line[80];
FILE *finp, *fout; /* declare file pointers */

DEFINE_EXECUTE_ON_LOADING(report_version, libname)
{
     /*read scaling data*/
     int i=0;
     time_hold=0;
     finp = fopen("scaling_data.txt", "r"); /* open finp
for read */
     while( feof(finp) == 0 ) /* read until EOF reached in
input */
```

```
          {
               fgets(line, 70, finp); /* read 70 characters
*/
               sscanf(line, "%g %g", &time_scale_hold,
&scaling_table_hold);
               time_scale[i] = time_scale_hold;
               scaling_table[i] = scaling_table_hold;
               i=i+1;
          }
     fclose(finp); /* close finp */
    Message("Reading scaling data complete\n");

     /*read data*/
     i=0;
     finp = fopen("solid_exterior2-plus.txt", "r"); /* open
finp for read */
     while( feof(finp) == 0 ) /* read until EOF reached in
input */
          {
               fgets(line, 70, finp); /* read 70 characters
*/
               sscanf(line, "%g %g %g %g",
&solid_exterior2plusx_hold, &solid_exterior2plus0_hold,
&solid_exterior2plus6_hold, &solid_exterior2plus10_hold);
               solid_exterior2plusx[i] =
solid_exterior2plusx_hold;
               solid_exterior2plus0[i] =
solid_exterior2plus0_hold;
               solid_exterior2plus6[i] =
solid_exterior2plus6_hold;
               solid_exterior2plus10[i] =
solid_exterior2plus10_hold;
               i=i+1;
          }
     fclose(finp); /* close finp */
    Message("Reading plus temperature data complete\n");

     /*read profile data plus*/
     i=0;
     finp = fopen("solid_exterior2-minus.txt", "r"); /*
open finp for read */
     while( feof(finp) == 0 ) /* read until EOF reached in
input */
          {
```

```
               fgets(line, 70, finp); /* read 70 characters
*/
               sscanf(line, "%g %g %g %g",
&solid_exterior2minusx_hold, &solid_exterior2minus0_hold,
&solid_exterior2minus6_hold, &solid_exterior2minus10_hold);
               solid_exterior2minusx[i] =
solid_exterior2minusx_hold;
               solid_exterior2minus0[i] =
solid_exterior2minus0_hold;
               solid_exterior2minus6[i] =
solid_exterior2minus6_hold;
               solid_exterior2minus10[i] =
solid_exterior2minus10_hold;
               i=i+1;
          }
     fclose(finp); /* close finp */
    Message("Reading minus temperature data complete\n");

     /*read profile data plus*/
     i=0;
     finp = fopen("solid_exterior2-minus.txt", "r"); /*
open finp for read */
     while( feof(finp) == 0 ) /* read until EOF reached in
input */
          {
               fgets(line, 70, finp); /* read 70 characters
*/
               sscanf(line, "%g %g %g %g",
&solid_fixedx_hold, &solid_fixed0_hold, &solid_fixed6_hold,
&solid_fixed10_hold);
               solid_fixedx[i] = solid_fixedx_hold;
               solid_fixed0[i] = solid_fixed0_hold;
               solid_fixed6[i] = solid_fixed6_hold;
               solid_fixed10[i] = solid_fixed10_hold;
               i=i+1;
          }
     fclose(finp); /* close finp */
    Message("Reading solid_fixed temperature data
complete\n");

     /*read profile data plus*/
     i=0;
     finp = fopen("fluid_interface-plus.txt", "r"); /* open
finp for read */
```

```
     while( feof(finp) == 0 ) /* read until EOF reached in
input */
          {
               fgets(line, 70, finp); /* read 70 characters
*/
               sscanf(line, "%g %g %g %g",
&fluid_interfaceplusx_hold, &fluid_interfaceplus0_hold,
&fluid_interfaceplus6_hold, &fluid_interfaceplus10_hold);
               fluid_interfaceplusx[i] =
fluid_interfaceplusx_hold;
               fluid_interfaceplus0[i] =
fluid_interfaceplus0_hold;
               fluid_interfaceplus6[i] =
fluid_interfaceplus6_hold;
               fluid_interfaceplus10[i] =
fluid_interfaceplus10_hold;
               i=i+1;
          }
     fclose(finp); /* close finp */
    Message("Reading fluid_interiorplus data complete\n");

     /*read profile data plus*/
     i=0;
     finp = fopen("fluid_interface-minus.txt", "r"); /*
open finp for read */
     while( feof(finp) == 0 ) /* read until EOF reached in
input */
          {
               fgets(line, 70, finp); /* read 70 characters
*/
               sscanf(line, "%g %g %g %g",
&fluid_interfaceminusx_hold, &fluid_interfaceminus0_hold,
&fluid_interfaceminus6_hold, &fluid_interfaceminus10_hold);
               fluid_interfaceminusx[i] =
fluid_interfaceminusx_hold;
               fluid_interfaceminus0[i] =
fluid_interfaceminus0_hold;
               fluid_interfaceminus6[i] =
fluid_interfaceminus6_hold;
               fluid_interfaceminus10[i] =
fluid_interfaceminus10_hold;
               i=i+1;
          }
     fclose(finp); /* close finp */
    Message("Reading fluid_interiorminus data complete\n");
```

```
}

DEFINE_PROFILE(solid_exterior2_plus,t,i)
{
   face_t f;

   real point[ND_ND];
   real x;
   real scaling;
   real time = CURRENT_TIME;
   real profile_max;
   real temp, temp_u, temp_l;
   int j;

   begin_f_loop(f,t)
   {
      F_CENTROID(point,f,t);
      x = point[0];

     /* define profile*/
       j=0;
       temp_u=300;

       if (time < 6)
       {
            profile_max=547;
            while (solid_exterior2plusx[j] < x)
            {
                 temp_l = temp_u;
                 temp_u = solid_exterior2plus0[j+1];
                 j=j+1;
            }
       }
       if (time >= 6 && time < 10)
       {
            profile_max=1033;
            while (solid_exterior2plusx[j] < x)
            {
                 temp_l = temp_u;
                 temp_u = solid_exterior2plus6[j+1];
                 j=j+1;
            }
       }
       if (time >= 10)
```

```
       {
            profile_max=893;
            while (solid_exterior2plusx[j] < x)
            {
                 temp_l = temp_u;
                 temp_u = solid_exterior2plus10[j+1];
                 j=j+1;
            }
       }

     /* define scaling*/
       j=0;
       if (time != time_hold)
       {
            while (time_scale[j] < time)
            {
                 scaling = scaling_table[j]/profile_max;
                 j=j+1;
            }
       }
       time_hold = time;

     /* define scaled profile*/
       temp = (temp_l + temp_u)/2;
       temp = ((temp-300)*scaling)+300;
       F_PROFILE(f,t,i) = temp;
   }
   end_f_loop(f,t)
}

DEFINE_PROFILE(solid_exterior2_minus,t,i)
{
   face_t f;

   real point[ND_ND];
   real x;
   real scaling;
   real time = CURRENT_TIME;
   real profile_max;
   real temp, temp_u, temp_l;
   int j;

   begin_f_loop(f,t)
   {
      F_CENTROID(point,f,t);
```

```
 x = point[0];

/* define profile*/
  j=0;
  temp_u=300;

  if (time < 6)
  {
       profile_max=547;
       while (solid_exterior2minusx[j] < x)
       {
            temp_l = temp_u;
            temp_u = solid_exterior2minus0[j+1];
            j=j+1;
       }
  }
  if (time >= 6 && time < 10)
  {
       profile_max=1033;
       while (solid_exterior2minusx[j] < x)
       {
            temp_l = temp_u;
            temp_u = solid_exterior2minus6[j+1];
            j=j+1;
       }
  }
  if (time >= 10)
  {
       profile_max=893;
       while (solid_exterior2minusx[j] < x)
       {
            temp_l = temp_u;
            temp_u = solid_exterior2minus10[j+1];
            j=j+1;
       }
  }

/* define scaling*/
  j=0;
  if (time != time_hold)
  {
       while (time_scale[j] < time)
       {
            scaling = scaling_table[j]/profile_max;
            j=j+1;
```

```
            }
       }
       time_hold = time;

     /* define scaled profile*/
       temp = (temp_l + temp_u)/2;
       temp = ((temp-300)*scaling)+300;
       F_PROFILE(f,t,i) = temp;
   }
   end_f_loop(f,t)
}
DEFINE_PROFILE(solid_fixed_t,t,i)
{
   face_t f;

   real point[ND_ND];
   real x;
   real time = CURRENT_TIME;
   real temp, temp_u, temp_l;
   int j;

   begin_f_loop(f,t)
   {
      F_CENTROID(point,f,t);
      x = point[0];

     /* define profile*/
       j=0;
       temp_u=300;
       if (time < 6)
       {
            while (solid_fixedx[j] < x)
            {
                 temp_l = temp_u;
                 temp_u = solid_fixed0[j+1];
                 j=j+1;
            }
       }
       if (time >= 6 && time < 10)
       {
            while (solid_fixedx[j] < x)
            {
                 temp_l = temp_u;
                 temp_u = solid_fixed6[j+1];
                 j=j+1;
```

```
            }
       }
       if (time >= 10)
       {
            while (solid_fixedx[j] < x)
            {
                 temp_l = temp_u;
                 temp_u = solid_fixed10[j+1];
                 j=j+1;
            }
       }
       temp = (temp_l + temp_u)/2;
       F_PROFILE(f,t,i) = temp;
   }
   end_f_loop(f,t)
}
DEFINE_PROFILE(fluid_int_minus,t,i)
{
   face_t f;

   real point[ND_ND];
   real x;
   real time = CURRENT_TIME;
   real temp, temp_u, temp_l;
   int j;

   begin_f_loop(f,t)
   {
      F_CENTROID(point,f,t);
      x = point[0];

     /* define profile*/
       j=0;
       temp_u=300;
       if (time < 6)
       {
            while (fluid_interfaceminusx[j] < x)
            {
                 temp_l = temp_u;
                 temp_u = fluid_interfaceminus0[j+1];
                 j=j+1;
            }
       }
       if (time >= 6 && time < 10)
       {
```

```
            while (fluid_interfaceminusx[j] < x)
            {
                 temp_l = temp_u;
                 temp_u = fluid_interfaceminus6[j+1];
                 j=j+1;
            }
       }
       if (time >= 10)
       {
            while (fluid_interfaceminusx[j] < x)
            {
                 temp_l = temp_u;
                 temp_u = fluid_interfaceminus10[j+1];
                 j=j+1;
            }
       }
       temp = (temp_l + temp_u)/2;
       F_PROFILE(f,t,i) = temp;
   }
   end_f_loop(f,t)
}
DEFINE_PROFILE(fluid_int_plus,t,i)
{
   face_t f;

   real point[ND_ND];
   real x;
   real time = CURRENT_TIME;
   real temp, temp_u, temp_l;
   int j;

   begin_f_loop(f,t)
   {
      F_CENTROID(point,f,t);
      x = point[0];

     /* define profile*/
       j=0;
       temp_u=300;
       if (time < 6)
       {
            while (fluid_interfaceplusx[j] < x)
            {
                 temp_l = temp_u;
                 temp_u = fluid_interfaceplus0[j+1];
```

```
                 j=j+1;
            }
       }
       if (time >= 6 && time < 10)
       {
            while (fluid_interfaceplusx[j] < x)
            {
                 temp_l = temp_u;
                 temp_u = fluid_interfaceplus6[j+1];
                 j=j+1;
            }
       }
       if (time >= 10)
       {
            while (fluid_interfaceplusx[j] < x)
            {
                 temp_l = temp_u;
                 temp_u = fluid_interfaceplus10[j+1];
                 j=j+1;
            }
       }
       temp = (temp_l + temp_u)/2;
       F_PROFILE(f,t,i) = temp;
   }
   end_f_loop(f,t)
}
```

Appendix C: Fluent User-Defined Functions for Study 3

```
!Reading tabulare temperature data for BC
#include "udf.h"
#include <stdio.h>

float hour[600], hot[600], cold[600], mean[600];
float hour_hold, hot_hold, cold_hold, mean_hold;
float table_size;

char line[80];
FILE *finp, *fout; /* declare file pointers */

DEFINE_EXECUTE_ON_LOADING(read_data, libname)
{
     /*read temperature data*/
     int i=0;
     finp = fopen("temp_data.txt", "r"); /* open finp for
read */
     while( feof(finp) == 0 ) /* read until EOF reached in
input */
          {
               fgets(line, 70, finp); /* read 70 characters
*/
               sscanf(line, "%g %g %g %g", &hour_hold,
&hot_hold, &cold_hold, &mean_hold);
               hour[i] = hour_hold*3600;
               hot[i] = hot_hold;
               cold[i] = cold_hold;
               mean[i] = mean_hold;
               i=i+1;
          }
     table_size = i-2;

     fclose(finp); /* close finp */
    Message("\n\nReading temperature data complete\n\n");
}

DEFINE_PROFILE(hot_temp,t,i)
{
   real time = CURRENT_TIME;
   real temp = 1;
   face_t f;
   int j;
```

```
   begin_f_loop(f,t)
   {
     /* define profile as a function of hour*/
       j=0;
       while (j < table_size)
       {
            if (hour[j] <= time)
            {
                 temp = hot[j];
            }
            j=j+1;
       }
       F_PROFILE(f,t,i) = temp;
   }
   end_f_loop(f,t)
}

DEFINE_PROFILE(cold_temp,t,i)
{
   real time = CURRENT_TIME;
   real temp = 1;
   face_t f;
   int j;

   begin_f_loop(f,t)
   {
     /* define profile as a function of hour*/
       j=0;
       while (j < table_size)
       {
            if (hour[j] <= time)
            {
                 temp = cold[j];
            }
            j=j+1;
       }
       F_PROFILE(f,t,i) = temp;
   }
   end_f_loop(f,t)
}

DEFINE_PROFILE(mean_temp,t,i)
{
   real time = CURRENT_TIME;
   real temp = 1;
```

```
   face_t f;
   int j;

   begin_f_loop(f,t)
   {
     /* define profile as a function of hour*/
       j=0;
       while (j < table_size)
       {
            if (hour[j] <= time)
            {
                 temp = mean[j];
            }
            j=j+1;
       }
       F_PROFILE(f,t,i) = temp;
   }
   end_f_loop(f,t)
}
```

Appendix D: ANSYS Mechanical APDL Custom Scripting for Study 4

```
!define thermal material properties
mp,c,3,200
mp,kxx,3,0.001
mp,kyy,3,0.001
mp,kzz,3,0.001

mp,c,12,1
mp,kxx,12,0.001
mp,kyy,12,1
mp,kzz,12,1

!define frictional heating parameter between tube to base
keyopt,5,1,1
keyopt,6,1,1

rmodif,5,9,500e6
rmodif,5,14,100
rmodif,5,15,1
rmodif,5,18,0.50

rmodif,6,9,500e6
rmodif,6,14,100
rmodif,6,15,1
rmodif,6,18,0.50

!define frictional heating parameter between tube to cam
keyopt,8,1,1
keyopt,9,1,1

rmodif,8,9,500e6
rmodif,8,14,100
rmodif,8,15,1
rmodif,8,18,0.50

rmodif,9,9,500e6
rmodif,9,14,100
rmodif,9,15,1
rmodif,9,18,0.50

!define frictional heating parameter between tube to tube
keyopt,10,1,1
keyopt,11,1,1
```

```
rmodif,10,9,500e6
rmodif,10,14,100
rmodif,10,15,1
rmodif,10,18,0.50

rmodif,11,9,500e6
rmodif,11,14,100
rmodif,11,15,1
rmodif,11,18,0.50

!changes element type form SOLID186/187 to SOLID226/227,
enables temperature degrees of freedom

/prep7
! Get max element type number
*get,etype_num,etyp,0,num,max

! Define coupled field elements with thermal-structural DOF
et,etype_num+1,solid226,11
et,etype_num+2,solid227,11

! Change solid187 to solid227
esel,s,ename,,187
emodif,all,type,etype_num+2

! Change solid186 to solid226
esel,s,ename,,186
emodif,all,type,etype_num+1

! Select all elements
esel,all

/solu

! Thermal Boundary Conditions
! This commenad set the initial temperature to 20.0 C and
applies a convection coefficient of 500 W/m2-C

! Set initial temperature condition to
ic,all,temp,30.0

! Apply a temperature constraint on the tubing_exterior
d,tube_exterior,temp,30.0
```