High-Fidelity Material Response Modeling as Part of an Aerothermoelastic Framework for Hypersonic Flows

by

Jonathan E. Wiebenga

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Aerospace Engineering)
in The University of Michigan
2014

Doctoral Committee:

Professor Iain D. Boyd, Chair
Professor James R. Barber
Associate Professor Krzysztof J. Fidkowski
Ryan C. Gosse, Air Force Research Laboratory

To Dad and Mom

ACKNOWLEDGEMENTS

Over the course of my time at the University of Michigan I have been blessed with an abundance of support and help from too many people to count. From classmates and friends, to faculty and staff, I am grateful to all of them for the roles that they have played in my life, and for helping to make this dissertation possible.

I would first like to thank my advisor, Professor Iain Boyd, for his support and guidance during my graduate studies. His patience and advice, even when things were not going well, helped to keep me on track. I do not think that I could have asked for a better adviser. I would also like to thank Professor James Barber, Professor Krzysztof Fidkowski, and Ryan Gosse for taking the time to serve on my committee, and for being available to answer my questions. I am especially grateful to Ryan Gosse and Jon Poggie for providing me with the opportunity to spend a summer at AFRL.

Thanks are also due to fellow members of the Nonequilibrium Gas and Plasma Dynamics Laboratory with whom I've had the pleasure of working. The support and camaraderie of group members made the NGPDL an enjoyable research group to be a part of. I would also like to thank Hicham Alkandry for his patience in answering my questions about LeMANS, and Alexandre Martin for his help in starting my research on material response. I am very thankful for the many friends that I have had the opportunity to make at the University of Michigan. I am especially grateful to Brandon Smith, Lucas Hansen, and Ashley Verhoff for all of the good food, movie nights, game nights, kayaking trips, trivia victories, and other fun times that helped

to take my mind off of work, and made life enjoyable.

My family has always been there for me, and for that I want to express my sincere thanks. Thank you to my parents, Dave and Barb, for their unwavering love and support. Thank you also to my sisters, Shelby and Carrie, for their love, and for not giving their little brother too hard of a time about being in school for so long. Finally, I would like to thank my wife, Michelle, for her constant love, support, encouragement, and patience. I could not have done this without you.

TABLE OF CONTENTS

LIST OF FIGURES

ix

LIST OF TABLES

LIST OF APPENDICES

Appendix

LIST OF ACRONYMS

CFD computational fluid dynamics

CSD computational structural dynamics

CVFEM control volume finite element method

FRSC finite-rate surface chemistry

GMRES Generalized Minimal Residual

CSR compressed sparse row

GHL generalized Hooke's law

ASCC ABRES Shape Change Code

TPS thermal protection system

CHAPTER I

Introduction

1.1 Motivation

Hypersonic flight vehicles range in design from blunt-bodied reentry vehicles such as the Stardust sample return capsule shown in Figure 1.1a, to long, slender lifting body designs such as the X-43 that is shown in Figure 1.1b. The differences in design are driven by the flight conditions that a vehicle will encounter, as well as the purpose of the vehicle. Reentry vehicles are typically blunt-bodied because they must absorb very high heat loads for short periods of time, and they only require limited maneuverability. Lifting body designs, on the other hand, are typically used for vehicles that are designed for sustained flight within the atmosphere. These types of vehicles are subject to much lower heat loads than reentry vehicles, and they must incorporate a propulsion system, as well as the capability to maneuver within the atmosphere. While the applications for these different types of vehicles and the flight conditions that they will encounter may vary significantly, the ability to predict the aerothermal or aerothermoelastic response of a vehicle to the applied thermal and aerodynamic loads is crucial to a successful design.

The terms "aerothermal" and "aerothermoelastic" refer to the fluid, thermal, and possibly structural interactions that become important at high flight speeds. The connections between these various disciplines are often depicted geometrically using the

1
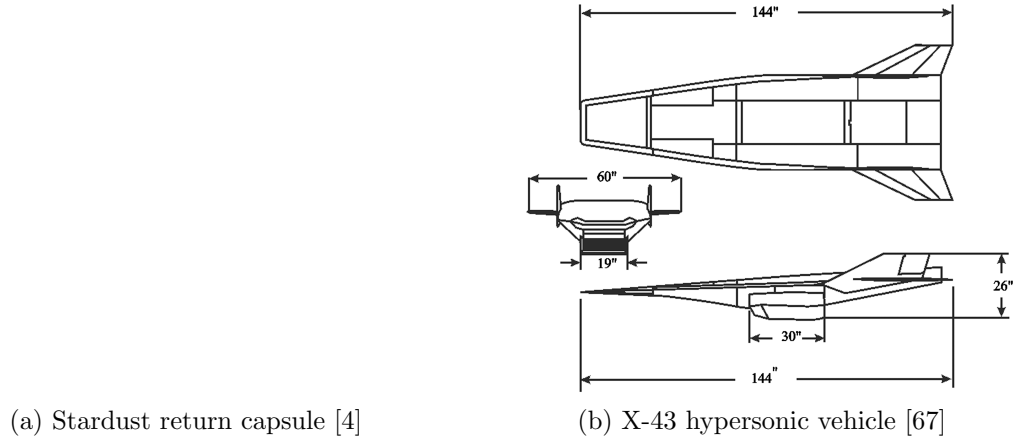
Figure 1.1: Examples of hypersonic vehicles.

aeroservothermoelastic hexahedron shown in Figure 1.2. Each node of the hexahedron corresponds to a different component of the full coupled problem, and the lines and faces between the nodes indicate coupling between those components. Aerothermal analysis is depicted by the line connecting the nodes for aerodynamic and thermal effects, and aerothermoelasticity extends this analysis to include the elastic node and the inertia node for dynamic problems. These four nodes form what is referred to as the aerothermoelastic tetrahedron [32].
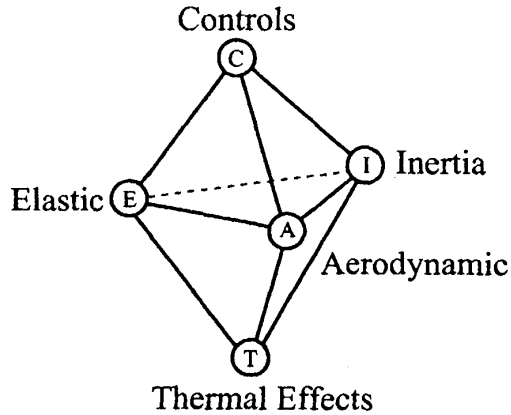
Figure 1.2: Aeroservothermoelastic hexahedron [43].

2

Figure 1.3 shows the trajectory for the Stardust sample return capsule, and the operating envelopes for several different hypersonic vehicles. It is obvious from these figures that hypersonic vehicles are expected to operate over a very wide range of velocities and at high altitudes. The combination of high altitude and high speed makes it very difficult and expensive to experimentally test hypersonic vehicles. In some cases it is not possible to match all of the flow and geometric parameters during a ground test that are relevant for an aerothermoelastic analysis. The cost and difficulty of hypersonic ground and flight testing makes the development of accurate and efficient computational tools a high priority for the design and analysis of hypersonic vehicles. However, the complicated flow physics that arise at high-speeds, as well as the difficulties inherent in accounting for multiple different disciplines in a coupled analysis, make the computational modeling of aerothermoelasticity a non-trivial problem.

(a) Stardust flight trajectory [4, 76]   (b) Hypersonic vehicle operating envelopes [67]
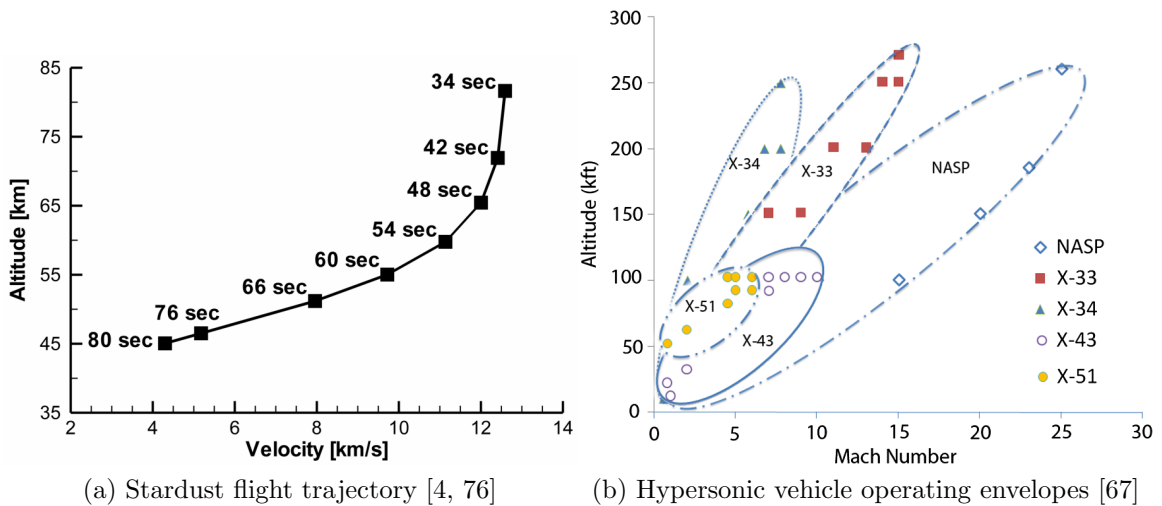
Figure 1.3: Stardust sample return capsule trajectory, and operating envelopes for several hypersonic vehicles.

As vehicle flight speed increases, the aerodynamic heating that is experienced by the vehicle also increases. This heating is due in large part to viscous dissipation within the boundary layer, however, for blunt bodies, the strong bow shock that

forms at hypersonic speeds also leads to substantial heating [7]. At large enough speeds, the vibrational energy modes of the air molecules can become excited, and, if the temperature continues to increase, the molecules will dissociate and the gas flow will become chemically reacting. The capability to predict these types of high-temperature gas phenomena greatly increases the computational cost and complexity of a fluid dynamics simulation, but it is crucial for accurate prediction of the heat flux and aerodynamic loads on a vehicle.

The heating of a flight vehicle through aerothermal loads can lead to several important structural effects. In his 1956 paper [14], Bisplinghoff identifies four important high-temperature structural considerations: deterioration of mechanical properties, thermal stresses, modification of stiffness and vibration properties, and lower boundaries for aeroelastic instabilities. Depending on the flight time and trajectory of a vehicle, these four different high-temperature effects can become more or less important. In the case of a reentry vehicle, flight times tend to be very short, but peak heating rates can be very large. For this type of trajectory, an ablative thermal protection system (TPS) is often used, which keeps the load bearing structure at a relatively low temperature, and the thermal impact on the elastic properties of the structure is minimal. It is still very important, however, to be able to accurately model the heating loads on the vehicle in order to properly size the TPS. In the case of an ablative TPS, surface recession can alter the vehicle geometry, and ablation products can alter the flow chemistry, so it is important to account for the coupled fluid-thermal interactions.

For air-breathing hypersonic vehicles, the projected flight times are much longer than reentry vehicles, but due to the lower flight speeds the heating rate tends to be much lower. This leads to a long thermal soak where the load bearing structure of the vehicle can heat up and remain at an elevated temperature for an extended period of time. For this reason, structural analysis of a hypersonic vehicle must account for

the time history of the aerodynamic heating. Figure 1.4a shows the vehicle exposure times and temperatures for a variety of different high-speed vehicles. Air-breathing hypersonic vehicles such as those listed in Figure 1.3b will tend to have exposure times closer to that of the X-30 than the other listed vehicles. Figure 1.4b shows the moduli of several different materials as functions of temperature, showing the degradation in material properties as temperature increases. For vehicles that undergo significant heating of the structure, it becomes important to couple the elastic analysis with the thermal analysis.

As a structure such as a panel heats up, it can deform due to stresses caused by thermal expansion. The deformation can then impact the flow field, leading to high localized pressure and heating loads [75], and causing a coupling between the fluid and structure that would not occur at low temperatures.

(a) Exposure times   (b) Modulus decrease
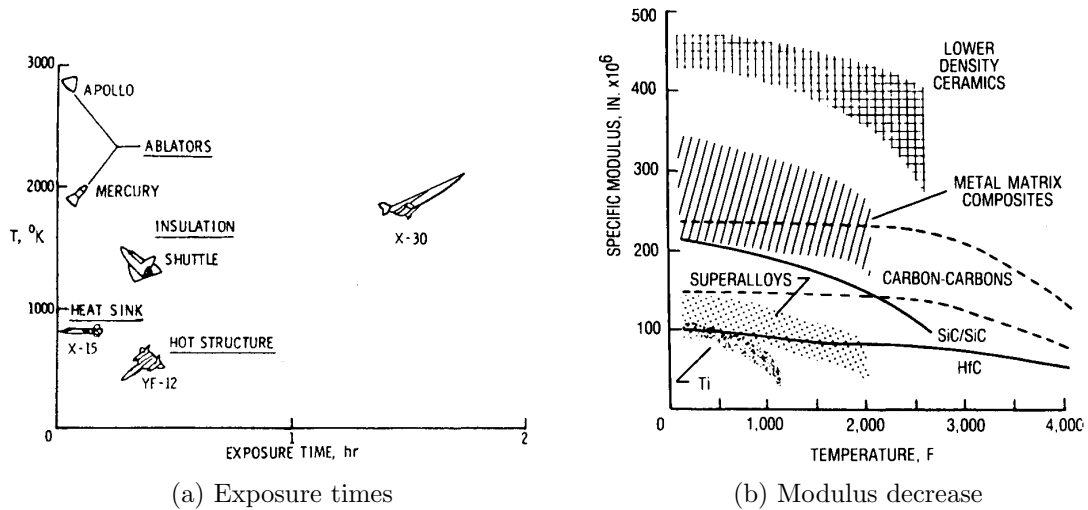
Figure 1.4: Flight exposure times for several hypersonic vehicles, and the decrease in modulus with temperature for several materials [93].

These strong interactions between the flow and structure, which are driven by thermal effects, lead to the need for a coupled computational analysis framework. The complexity of the high-speed flow physics, and the possibly complicated material

response, indicates the need for high-fidelity modeling. At the same time, however, the potentially long thermal soak times for a vehicle make a computationally efficient analysis framework necessary.

1.2   Review of Related Work

The necessity of considering aerothermal loads in the design of flight vehicles has its origins in the late 1940s and the 1950s, shortly after the first supersonic aircraft, the Bell X-1, was flown [93]. The development of high-speed aircraft proceeded relatively quickly after this point. As flight speeds increased, the importance of including aerothermal effects in the structural analysis also increased, and by the mid-1950s the main structural and aeroelastic considerations for high-speed flight had been identified [14]. With the advent of the space race, the need for aerothermal analysis was extended to the design of blunt-bodied reentry vehicles and thermal protection systems [93], which often consisted of ablative materials.

In these early years, the computational capability did not exist to perform an aerothermal or aerothermoelastic analysis of a vehicle using the full set of governing equations for the flow and structure, so many approximate methods for determining the aerodynamic thermal and pressure loads were developed. A discussion of some of the early approaches for predicting heating can be found in Van Driest [99] and Truitt [97], but one method that is still in use today is Eckert's reference enthalpy method [36]. An overview of many of the approximate aerodynamics theories that were developed for high-speed flight can be found in McNamara and Friedmann [67]. One of the most successful and widely used approximate aerodynamic models for aeroelasticity is piston theory [8, 57], which provides a simple relationship between surface motion and aerodynamic pressure.

Early approaches to coupling aerodynamic and structural response generally ignored thermal effects and focused on panel flutter [33, 35]. These studies used piston

theory and plate theory to develop an equation of motion that could be solved to study the unsteady response of an elastic panel in a high-speed flow. During this same time period, the capability to model the thermal effects on an ablative heat shield was also being introduced. One of the most widely used ablative material response tools to be developed was the one-dimensional Charring Materials Ablation (CMA) code [71].

More recently, as available computing power has increased, there have been many different studies performed on coupling aerodynamic, structural, and thermal models in order to study problems that are relevant to the design of hypersonic vehicles. Thornton and Dechaumphai [95] combined the solution of the fluid, thermal, and structural problems into a single finite element framework in order to study the response of thin metallic panels in hypersonic flow. The laminar Navier-Stokes equations for a perfect gas were solved along with the structural equations for a solid. A quasi-static coupling procedure was used where the solution sequence alternated between coupled flow and thermal analyses and structural thermal analysis. At a select number of times, the panel temperatures were used to update the structural deformations. This study showed both convex and concave panel deformations depending on how the panel boundary conditions were implemented. This solution approach was also applied to studies involving aerodynamically heated leading edges [30] and shock-shock interactions on cylinders and leading edges [31].

Löhner et al. [58] implemented a loosely coupled procedure for computing fluid-structure-thermal interaction problems. This approach allowed for the use of three independent codes to be used for the sub-problems, and a master code that coordinated the other codes and facilitated the passing of boundary information between codes was required. The use of off-the-shelf codes for the fluid, structure, and thermal problems greatly decreased the development time of a fluid-structure-thermal interaction framework.

Coupled analysis of a realistic metallic TPS was performed by Kontinos and Palmer [54] using a steady-state, axisymmetric computational fluid dynamics (CFD) code, and finite element models for the thermal and structural response of the TPS. Heat flux values at each coupled iteration time were obtained from a predefined flight trajectory and were applied uniformly over the surface of a panel, and the transient thermal response of the panel was computed. Using the thermal solution, the elastic response of the panel was computed, and the updated geometry, defined in terms of the panel bow height, was used to update the surface heat flux based on a pre-computed database of CFD solutions for flow over panels with varying bow height. The thermal calculation was then updated, and the thermal-structure procedure was iterated until the computed panel deflections converged.

Recently, there have also been several studies on fluid-thermal-structural coupling for hypersonic flows that make use of lower-order models that account for thermal effects, and are computationally inexpensive enough to study the dynamic response of a structure over longer flight times. Culler and McNamara implemented an aerothermoelastic framework that used piston theory, Eckert's reference enthalpy method, and von Kármán plate theory to study the coupled response of a simply-supported insulated metallic panel [27, 26] and a carbon-carbon skin panel [28].

More detailed aerothermal and aerothermoelastic models are proposed by Crowell et al. [25] and by Miller et al. [70]. In the first paper, the NASA CFL3D code [12] was coupled to a finite element model for solid heat transfer in a partitioned, time-accurate fashion. The resulting framework was used to investigate the aerothermal problem of an unsteady shock impinging on a vibrating panel. Both one-way and two-way coupling procedures were investigated, as well as the effect of quasi-steady and unsteady flow models. In the paper by Miller et al. [70], a complete aerothermoelastic framework that couples implicit CFD and finite element analyses in a partitioned, fully time-accurate fashion was developed, and, using some simplified models, the

8

order of accuracy of the coupled framework was demonstrated.

On a slightly different front, coupled analysis for the aerothermal modeling of problems involving surface ablation has also moved towards higher fidelity simulations. Blackwell and Hogan [15] introduced the use of the control volume finite element method (CVFEM) to solve the heat conduction equation in multiple dimensions on unstructured meshes, with an aerodynamic heating boundary condition to allow for uncoupled aerothermal analysis. Hogan, Blackwell, and Cochran [48] built on this work by including a thermochemical ablation boundary condition to account for the heat loss and surface recession caused by surface ablation. A mesh motion procedure based on treating the mesh as a linear elastic solid [59] was included to account for the change in geometry due to surface recession. Kuntz et al. [55] then demonstrated an iterative procedure for the coupled computation of an external flow field with a Navier-Stokes solver, and multi-dimensional material response with thermochemical ablation occurring at the solid surface.

1.3   Scope of Present Work

The goal of this dissertation is to develop a general material response code that is capable of analyzing the thermal and elastic behavior of a hypersonic vehicle, and coupling that code with an established Navier-Stokes CFD code. The material response code is to be general enough to be used on a wide variety of geometries, ranging from blunt reentry vehicles to the panels or sharp leading edges found on modern air-breathing hypersonic vehicles. The ability to model two or three-dimensional structures with anisotropic, temperature dependent material properties is included in the code, as well as the capability to model the thermochemical surface ablation of non-charring ablative materials. This generality allows the material response code to be used for a wide range of flight conditions and a wide range of vehicle types.

The code uses unstructured meshes and the CVFEM to discretize the governing

equations. Use of the same mesh for the thermal and structural problems allows for straightforward coupling of the thermal and structural solutions. The thermal model is time accurate, and the structural model can be used to study either dynamic or static structural responses depending on the type of problem being studied. A total Lagrangian structural mechanics formulation is used in conjunction with the Green-Lagrange strain tensor and the Generalized Hooke's Law to allow for the study of large elastic deformations. To ensure the correct implementation of the thermal and structural models, the Method of Manufactured Solutions is used to verify the code order of accuracy.

In addition to the development of a material response code, a framework is implemented for coupling the material response code with a hypersonic CFD code to allow for high-fidelity aerothermal and aerothermoelastic simulations of hypersonic vehicles. This work focuses specifically on the use of the framework for steady flow problems with quasi-static thermal-structural response. The framework is general enough, however, to support fully dynamic simulations.

Chapter II presents the governing equations for the CFD code, LeMANS, and for the thermal and structural portions of the material response code. Descriptions of the necessary constitutive relations are given along with the modeling assumptions that are made. The modeling of surface ablation, and the thermochemical ablation boundary condition are also discussed.

Chapter III gives details on the numerical approaches used to solve the fluid, thermal, and structural models presented in Chapter II. A brief overview is given of the finite volume discretization used in LeMANS, along with the approach used to enforce boundary conditions. Then, a description of the CVFEM is presented, and its application to the thermal and structural governing equations is discussed, along with the implementation of boundary conditions. Details on the systems of nonlinear equations that result from discretization are discussed, and the methods used to solve

them are outlined. Finally, the aerothermoelastic coupling framework is described.

The results of verification tests performed on LeMANS and the material response code are presented in Chapter IV. The Method of Manufactured Solutions and its application to the codes that are tested is presented, along with a discussion of code order of accuracy. The results from using the Method of Manufactured Solutions are shown, and any discrepancies between the expected and observed orders of accuracy for the tested codes are discussed.

Chapter V shows the results of using the aerothermoelastic framework on two different test cases. The first case involves aerothermal coupling and surface ablation for a reentry vehicle. The second case uses quasi-static aerothermoelastic coupling to investigate the response of a thermally insulated metallic panel exposed to hypersonic flow.

Finally, Chapter VI gives a summary of the work presented in the previous chapters, and highlights the contributions of this work to the topic of coupled hypersonic flow-thermal-structural response. Additionally, suggestions for future research are presented.

CHAPTER II

Mathematical Formulation

2.1 Introduction

Modeling the aerothermoelastic response of a structure involves a wide range of physical processes across multiple disciplines, and thus multiple governing equations are needed in order to fully describe the problem. The choice of governing equations presents a trade-off between simulation accuracy and speed, and so the choice is often governed by the desired application. The goal of this work is to develop a framework that allows for high-fidelity simulations of aerothermal and aerothermoelastic phenomena present in hypersonic flow, with an emphasis on problems that can be classified as quasi-static. In this context, quasi-static means that the relevant flow time scales are much shorter than the relevant elastic and thermal time scales of the structure. This leads to the assumption that the flow can be modeled as steady-state, but the elastic and thermal responses are time resolved.

Following from this goal, the flow is modeled using the Navier-Stokes equations, and Cauchy's equation and the heat equation are used for conservation of momentum and energy within a structure. The Navier-Stokes equations allow for the simulation of continuum flows that are in weak thermal and chemical nonequilibrium, which is appropriate for many hypersonic flows. The heat equation and Cauchy's equation allow for simulation of anisotropic materials with temperature dependent properties.

Thermo-mechanical coupling is neglected, however, so the structural energy and momentum equations are uncoupled.

This chapter presents the governing equations, associated models, and assumptions used in this work for the flow and structural components of the coupled problem. Emphasis is placed on the thermal and structural models as these were the main focus of this work, but an overview of the CFD modeling is also provided. For full details of the CFD model, the reader is referred to Scalabrin [87].

2.2 Computational Fluid Dynamics: LeMANS

The CFD code used in this study is LeMANS [87, 66], which is a Navier-Stokes code developed at the University of Michigan for use on hypersonic problems. The governing equations solved in LeMANS are shown in Equations 2.1 - 2.4.

$$\frac{\partial \rho_s}{\partial t} + \frac{\partial}{\partial x_i}(\rho_s u_i + J_{i,s}) = \dot{\omega}_s \tag{2.1}$$

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial}{\partial x_i}(\rho u_i u_j + p\delta_{ij} - \tau_{ij}) = 0 \tag{2.2}$$

$$\frac{\partial E}{\partial t} + \frac{\partial}{\partial x_i}((E + p - \tau_{ij})u_i + q_i + \sum J_{i,s}h_s) = 0 \tag{2.3}$$

$$\frac{\partial E_{ve}}{\partial t} + \frac{\partial}{\partial x_i}(E_{ve}u_i - q_{ve_i} + \sum J_{i,s}e_{ve,s}) = \dot{\omega}_v \tag{2.4}$$

Equation 2.1 represents conservation of mass on a per species basis, where ρ_s is the density of species s, u_i is the ith component of the velocity vector, $J_{i,s}$ is the mass diffusion of species s in the ith direction, and $\dot{\omega}_s$ is the mass production rate of species s. When solving for the flow field quantities, it is necessary to solve N_s conservation of mass equations, where N_s is the total number of species. The species mass diffusion fluxes are modeled using a modified form of Fick's law [91] shown in Equation 2.5 that ensures that the sum of the diffusion fluxes is zero. In this equation, ρ is the total density of the gas mixture, D_s is the diffusion coefficient of species s, Y_s is the mass

fraction of species s, C_s is the charge of species s, and M_s is the species molecular weight.

$$\left.\begin{array}{c} J_{i,s\neq e} = -\rho D_s \dfrac{\partial Y_s}{\partial x_i} - Y_s \sum_{r\neq e} -\rho D_r \dfrac{\partial Y_r}{\partial x_i} \\[2em] J_{i,e} = M_e \sum_{s\neq e} \dfrac{J_{i,s}C_s}{M_s} \end{array}\right\} \qquad (2.5)$$

The species mass production rates are computed using either a 5-species or 11-species finite-rate chemistry model for reacting air. More complicated chemistry sets that include ablation products along with air species can also be used. In order to account for the level of nonequilibrium in the flow when computing the forward and backward reaction rates, Park's two-temperature model [77] is adopted. This approach uses a controlling temperature for the reaction rates, T_c, that is determined as: $T_c = T_{tr}^a T_{ve}^b$, where the subscripts indicate the translational-rotational or vibrational-electronic energy modes, and a and b allow for different weighting of each energy mode. The forward reaction rates are then calculated using Arrhenius curve fits on the controlling temperature, and the backward rates are obtained using a backward controlling temperature and equilibrium constants determined from Gibb's free energy as shown in Equation 2.6. In this equation, p_0 is a reference pressure equal to 1 bar and $\nu_r = \sum_s \left(\beta_{s,r} - \alpha_{s,r} \right)$ where α and β are the stoichiometric coefficients for each species s in each reaction r. The necessary enthalpy $(\hat{h}_s)$ and entropy $(\hat{s}_s)$ values are determined from curve fits, and can be found in Scalabrin [87].

$$K_{eq} = \left(\frac{p_0}{RT_{tr}} \right)^{\nu_r} \exp \left\{ -\sum_s \left(\beta_{s,r} - \alpha_{s,r} \right) \left[\frac{\hat{h}_s}{RT_{tr}} - \frac{\hat{s}_s}{R} \right] \right\} \qquad (2.6)$$

Conservation of momentum is given by Equation 2.2 where p is the pressure and τ_{ij} is the viscous stress tensor. Viscous stresses are modeled in LeMANS by assuming a Newtonian fluid and making use of Stokes' hypothesis [105]. This leads to the

formulation:

$$\tau_{ij} = \mu \left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \right) - \frac{2}{3} \mu \frac{\partial u_k}{\partial x_k} \delta_{ij} \qquad (2.7)$$

where μ is the coefficient of viscosity of the mixture and δ_{ij} is the Kronecker delta.

Equation 2.3 shows the conservation of total energy, where E is the total energy per unit volume, q_i is the ith component of the total heat flux vector, and h_s is the enthalpy of species s.

The Navier-Stokes equations are capable of modeling weakly nonequilibrium flows, and at hypersonic speeds this capability can become important. LeMANS is capable of simulating flows in rotational, vibrational, and electronic nonequilibrium; however, for this work it is assumed that only the vibrational mode is important. In this case, it is assumed that the rotational and translational energy modes of all species can be described by the temperature T_{tr}, and the vibrational and electronic modes of all species can be described by a single temperature, T_{ve}. This approach assumes that the continuum approximation is valid and therefore the rotational nonequilibrium will be negligible since the rotational and translational energies will reach equilibrium after just a small number of collisions [87]. For the vibrational modes, it is assumed that the transfer of energy between electrons and the vibrational mode of molecules is very fast [45], and that vibrational-vibrational energy transfer between molecules is very efficient, which leads to similar vibrational temperatures for different molecules [23, 56].

The vibrational-electron-electronic temperature, T_{ve}, is obtained by solving an additional conservation of energy equation given by Equation 2.4. In this equation, E_{ve} is the vibrational-electron-electronic energy per unit volume of the mixture, q_{ve_i} is vibrational-electron-electronic heat flux in the ith direction, $e_{ve,s}$ is the vibrational-electron-electronic energy per unit mass of species s, and $\dot{\omega}_{ve}$ is the vibrational energy source term. This term can include changes in vibrational energy due to the work done on electrons by an electric field, chemical reactions, vibrational relaxation, energy

15

transfer between heavy particles and electrons, and impact ionization reactions.

LeMANS is capable of modeling the mixture transport properties in multiple ways, but this work makes use of Wilke's semi-empirical mixing rule [107] to determine the mixture viscosity and thermal conductivity. Blottner's curve fits [16] are used to calculate species viscosities, and species thermal conductivities are determined separately for the translational-rotational and vibrational-electron-electronic energy modes using Eucken's relation [102].

Conductive heat fluxes are modeled using Fourier's law, and the total conductive heat flux is computed as shown in Equation 2.8 with contributions from both energy modes.

$$q_{cond_i} = -\left(\kappa_{tr} \frac{\partial T_{tr}}{\partial x_i} + \kappa_{ve} \frac{\partial T_{ve}}{\partial x_i} \right) \tag{2.8}$$

An additional heat flux due to mass diffusion is also computed as shown in Equation 2.9.

$$q_{md_i} = -\sum_s J_{i,s} h_s \tag{2.9}$$

The total heat flux is then given by Equation 2.10.

$$q_{tot_i} = q_{cond_i} + q_{md_i} \tag{2.10}$$

2.3 Material Response

The following sections detail the governing equations and models used in the material response portion of this work. The material response module consists of two components: a heat equation solver for obtaining the thermal response, and a structural mechanics solver for determining the elastic response of a structure. It is assumed that the material properties are functions of temperature, but thermo-mechanical coupling is neglected. While this coupling may be important in some analyses, it is generally neglected in aerospace problems of the type considered in

this work [93]. The main result of this assumption is that the energy and momentum equations are decoupled and can be solved independently. In order to include the effect of temperature on a structure, the energy equation is solved first, and the computed temperature field is then used as a thermal load in the momentum equation.

2.3.1   Thermal response

The thermal response of a material is calculated from the energy balance shown in Equation 2.11, where $\tilde{q}_i$ is the heat flux in the ith direction, ρ is the material density, h is the specific enthalpy, v_{cs_i} is the grid velocity in the ith direction, and e is the specific internal energy of the material. This is the heat equation with a source term and written for an arbitrarily moving control volume.

$$\int_{\partial\Omega} \tilde{q}_i \hat{n}_i \, \mathrm{d}\partial\Omega \; - \; \int_{\partial\Omega} \rho h v_{cs_i} \hat{n}_i \, \mathrm{d}\partial\Omega \; + \; \frac{d}{dt}\int_{\Omega} \rho e \mathrm{d}\Omega \; = \; \int_{\Omega} Q \mathrm{d}\Omega \qquad (2.11)$$

The reason for formulating the energy equation for a moving control volume is to allow for the modeling of surface ablation. Broadly speaking there are two main classes of ablative materials that are used on aerospace vehicles: charring and non-charring [62]. Charring materials consist of a matrix filled with a resin, and as the material is heated the resin can decompose, or pyrolyze, which serves to keep the vehicle's structure at a relatively cool temperature. These types of ablators may or may not experience surface recession once a sufficient amount of resin has pyrolyzed from the interior of the material. Non-charring ablators, on the other hand, do not contain a decomposing resin. Chemical reactions at the material surface serve to reduce the heat flux to the vehicle, resulting in surface recession that must be modeled.

Only non-charring materials are considered in this work, so the effects of pyrolysis

are neglected, but the ability to model a recessing surface is included. The grid convection term in Equation 2.11 allows for the deformation of the entire mesh in response to surface recession in order to prevent the formation of highly skewed or overlapping grid elements near the surface.

The heat flux is modeled using Fourier's Law as shown in Equation 2.12 where $\bar{\kappa}_{ij}$ is the thermal conductivity tensor and T is the material temperature. The tensor formulation allows for heat conduction in fully anisotropic materials to be modeled.

$$\tilde{q}_i = -\bar{\kappa}_{ij}\frac{\partial T}{\partial x_j} \tag{2.12}$$

The thermal conductivity, specific enthalpy, and specific energy are assumed to be functions of temperature. The specific enthalpy and energy, which are the same for a solid material, are determined from a user-input table of the specific heat, c_p, as a function of temperature via parabolic interpolation. Parabolic interpolation is used since the specific heat is assumed to vary linearly between the input values, so it would be incorrect to also assume a linear variation in internal energy. The interpolation routine that is used is described in Amar[5].

In addition to temperature dependence, all of the material properties needed for the thermal response equation are assumed to be functions of material composition, where the composition varies linearly between a virgin state and a char state. The purpose of interpolating between two states is to enable the simulation of pyrolyzing ablative materials where in-depth decomposition of the material occurs. The interpolation scheme used is shown in Equation 2.13 for a generic property, Θ_v with y_v representing the fraction of material still in the virgin state, and the subscripts indicating properties in the fully virgin and fully char states. The study of pyrolyzing ablative materials is outside the scope of this work, so for all properties it is assumed that the virgin and char states are identical ($\Theta_v = \Theta_c$), and therefore the material

composition does not change and all properties are constant with respect to material composition. Density is therefore constant throughout a simulation since it does not vary with temperature either.

$$\Theta(T) = y_v \Theta_v(T) + (1 - y_v)\Theta_c(T) \tag{2.13}$$

In cases where there is surface recession due to ablation, v_{cs_i} will be non-zero. The material velocity at the recessing surface is determined in one of two ways. First, the velocity can be obtained through the use of a thermochemical ablation, or B', table as described in Amar [5]. This approach requires the formation of an input table before computing the material response, and assumes equilibrium chemistry, which may not be a valid assumption in all cases. The second approach is to use a finite-rate surface chemistry (FRSC) code to compute a recession rate at the material surface [4]. This method is generally preferable since it allows for nonequilibrium chemistry and avoids the need to generate and interpolate a thermochemical table. The end result of both of these approaches is a surface recession rate, which allows v_{cs_i} at the material surface to be computed as shown in Equation 2.14, where $\dot{s}$ is the recession rate and $\hat{n}_{in_i}$ is the inward pointing surface normal.

$$v_{cs_i} = \dot{s}\hat{n}_{in_i} \tag{2.14}$$

The control volume velocities for the interior of the domain are then determined based on the chosen mesh motion scheme. In this work, the mesh is considered to be an elastic solid [59, 48], and the deformation is computed by solving the equilibrium stress equations with zero body force shown in Equation 2.15. Since only the mesh displacements are desired and the stresses have no physical meaning, the elastic constants for the fictitious mesh "material" can be chosen arbitrarily. Following the work

of Hogan [48], the elastic properties are chosen to be representative of steel.

$$\frac{\partial \sigma_{ij}}{\partial x_j} = 0 \tag{2.15}$$

The source term, Q, is an optional addition to Equation 2.11, and can represent either an internal volumetric energy source or sink.

2.3.2 Structural Response

The elastic response of a structure is determined from Equation 2.16, which is known as Cauchy's equation. This equation states the conservation of momentum for a continuum. In Equation 2.16, ρ is the material density, u_i is the displacement in the ith direction, b_i is the body force in the ith direction, and σ_{ij} is the stress tensor. The double dots above a variable indicate the second derivative with respect to time, so $\ddot{u}_i$ is the acceleration. In this work, body forces and structural damping are neglected, so the governing equation that is actually solved is shown in Equation 2.17.

$$\rho \ddot{u}_i - b_i - \frac{\partial \sigma_{ij}}{\partial x_j} = 0 \tag{2.16}$$

$$\rho \ddot{u}_i - \frac{\partial \sigma_{ij}}{\partial x_j} = 0 \tag{2.17}$$

The stress tensor can be broken into several different contributions as shown in Equation 2.18, where σ_{ij}^T is the total stress, σ_{ij}^e is the elastic stress, σ_{ij}^{th} is the thermal stress, and σ_{ij}^0 is the initial stress.

$$\sigma_{ij}^e = \sigma_{ij}^T - \sigma_{ij}^{th} - \sigma_{ij}^0 \tag{2.18}$$

The generalized Hooke's law (GHL) is used to provide a linear relationship between the elastic stresses and strains, where the elastic strains can be written as $\epsilon_{ij}^e = \epsilon_{ij}^T - \epsilon_{ij}^{th} - \epsilon_{ij}^0$. Using the GHL, Equation 2.18 can be written in terms of strains as

shown in Equation 2.19, where C_{ijkl} is the fourth-order stiffness tensor.

$$\sigma_{ij}^e = C_{ijkl}\left(\epsilon_{kl}^T - \epsilon_{kl}^{th} - \epsilon_{kl}^0\right) \tag{2.19}$$

The total strains are computed using the Green-Lagrange strain tensor as shown in Equation 2.20, and the thermal strains are given by Equation 2.21. In this equation, α_i is the thermal expansion coefficient in the ith direction and ΔT is the change in temperature between an initial state and the current state of the material. The initial strains are user inputs and in this work they are neglected.

$$\epsilon_{ij}^T = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} + \frac{\partial u_k}{\partial x_i}\frac{\partial u_k}{\partial x_j}\right) \tag{2.20}$$

$$\epsilon_{ij}^{th} = \alpha_i \Delta T \delta_{ij} \tag{2.21}$$

In conjunction with the Green-Lagrange strain tensor, the 2nd Piola-Kirchhoff stress tensor is used. This tensor is work-conjugate with the Green-Lagrange strain tensor and is appropriate for analyses involving large displacements or rotations and either large or small strains [13]. The 2nd Piola-Kirchhoff stress tensor, $\tilde{\boldsymbol{\sigma}}$, is related to the Cauchy stress tensor, $\boldsymbol{\sigma}^c$, through Equation 2.22, where $\mathbf{F}$ is the deformation gradient tensor given by Equation 2.23. This tensor relates the initial geometry defined by x_0, y_0, and z_0 to the current geometry given by $x_0 + u$, $y_0 + v$, and $z_0 + w$ where u, v, and w are the displacements in each of the coordinate directions. The deformation gradient tensor is the gradient of the current displacement vector with respect to the initial undeformed geometry.

$$\begin{aligned}
\tilde{\boldsymbol{\sigma}} &= \frac{{}^0\rho}{{}^t\rho}\mathbf{F}^{-1}\boldsymbol{\sigma}^c\left(\mathbf{F}^{-1}\right)^T \\
&= \det\left(\mathbf{F}\right)\mathbf{F}^{-1}\boldsymbol{\sigma}^c\left(\mathbf{F}^{-1}\right)^T
\end{aligned} \tag{2.22}$$

21

$$\mathbf{F} = \begin{bmatrix} \frac{\partial(x_0+u)}{\partial x_0} & \frac{\partial(x_0+u)}{\partial y_0} & \frac{\partial(x_0+u)}{\partial z_0} \\ \frac{\partial(y_0+v)}{\partial x_0} & \frac{\partial(y_0+v)}{\partial y_0} & \frac{\partial(y_0+v)}{\partial z_0} \\ \frac{\partial(z_0+w)}{\partial x_0} & \frac{\partial(z_0+w)}{\partial y_0} & \frac{\partial(z_0+w)}{\partial z_0} \end{bmatrix} \tag{2.23}$$

The purpose of using the Green-Lagrange strain tensor, which is appropriate for large strains and large rotations, and the GHL, which is generally only appropriate for small strains, is to be able to model structures where there may be large rotations, but the strains remain small. One example of this type of problem which is studied in this work is a metallic plate deforming due to a thermal load. If the large rotation but small strain assumptions are not valid for a given case, then a more appropriate stress-strain relationship should be employed.

The governing equation must also be formulated in such a way as to account for the change in geometry due to finite deformations. There is more than one approach to accomplish this, but in this work the total Lagrangian [13] formulation is adopted, which measures the stress tensor relative to the original geometry. Equation 2.24 shows this formulation of the governing equation given by Equation 2.16.

$$\rho \ddot{u}_i - \frac{\partial}{\partial x_{0_j}} \left(F_{ik} \tilde{\sigma}_{jk} \right) = 0 \tag{2.24}$$

Equation 2.24 can be rewritten in integral form as shown in Equation 2.25, which is a more useful form for the numerical scheme discussed in Chapter III. Note that the integration in Equation 2.25 is performed over the initial undeformed geometry in keeping with the total Lagrangian approach.

$$\int_{\Omega_0} \rho \ddot{u}_i \mathrm{d}\Omega_0 - \int_{\Omega_0} \frac{\partial}{\partial x_{0_j}} \left(F_{ik} \tilde{\sigma}_{jk} \right) \mathrm{d}\Omega_0 = 0 \tag{2.25}$$

The notation used for the structural response equations can be simplified by taking into account the symmetry of the stress and strain tensors, which reduces the number

of independent stress and strain terms to six each. This allows the tensors σ_{ij} and ϵ_{ij} to be represented as the vectors σ_i and ϵ_i shown in Equation 2.26. Note that in this instance σ_{ij} and ϵ_{ij} are generic stress and strain tensors, and this simplification is valid for any of the stress and strain tensors that are used.

$$\sigma_i = \begin{pmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{pmatrix} = \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{pmatrix} \quad \text{and,} \quad \epsilon_i = \begin{pmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{33} \\ \epsilon_{12} \\ \epsilon_{13} \\ \epsilon_{23} \end{pmatrix} = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \\ \epsilon_6 \end{pmatrix} \tag{2.26}$$

In a similar manner, these symmetries reduce the total number of elastic constants for a fully anisotropic material from 81 to 36, and the stiffness tensor, C_{ijkl}, can be represented as a 6×6 stiffness matrix, $\mathbf{D}$. In this work the structural response is considered only for isotropic and orthotropic materials, and for these cases the stiffness matrix has the general form shown in Equation 2.27.

$$\mathbf{D} = \begin{bmatrix} d_{11} & d_{12} & d_{13} & 0 & 0 & 0 \\ d_{21} & d_{22} & d_{23} & 0 & 0 & 0 \\ d_{31} & d_{32} & d_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & d_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & d_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & d_{66} \end{bmatrix} \tag{2.27}$$

In the case of an orthotropic material, it is assumed that the material has three orthogonal planes of symmetry [61]. The material properties needed in the constitutive relations are Young's modulus (E), shear modulus (G), and Poisson's ratio (ν). For a three-dimensional problem this results in nine independent elastic constants

that need to be specified. In the case of an isotropic material, the elastic constants are independent of the orientation of the axes, and so only two independent constants need to be specified: Young's modulus and Poisson's ratio. The shear modulus can then be determined as shown in Equation 2.28 [61]. More details on the constitutive relations can be found in Appendix A.

$$G = \frac{E}{2\left(1+\nu\right)} \tag{2.28}$$

In addition to the elastic constants, it is also necessary to specify thermal expansion coefficients (α) in order to compute thermal stresses. For an orthotropic material there are three separate coefficients, one for each coordinate direction, and for an isotropic material there is only one. The elastic constants and thermal expansion coefficients are all assumed to be functions of temperature, and so can vary throughout a material if a temperature gradient is present. The density, however, is assumed to be constant.

CHAPTER III

Numerical Methods

3.1 Introduction

A variety of different techniques are available to reduce a system of continuous governing equations into a set of discrete equations that can be solved on a computer. Three of the most widely used approaches are the finite difference method, the finite volume method, and the finite element method [47]. While each of these methods is applicable to a wide range of problems, the finite volume method is the most widely used approach for fluid dynamics problems, and the finite element method is the standard approach in the field of structural analysis [47]. In this thesis, a finite volume approach is used to discretize the flow field, and a finite volume/finite element hybrid approach, known as the control volume finite element method (CVFEM), is used to discretize the thermal and elastic material response problems.

This chapter details the numerical approach used to solve the equations presented in Chapter II. Included is an overview of the spatial and temporal discretization techniques used in LeMANS and the material response code, as well as the numerical methods used to solve the resulting sets of discrete equations. The different types of boundary conditions and their implementation are discussed, with an emphasis on the boundary conditions needed to facilitate aerothermoelastic coupling. Finally, the aerothermoelastic coupling procedure is described, including the technique for moving

the flow mesh in response to material deformation.

3.2   LeMANS

LeMANS is a Navier-Stokes code that uses a second-order accurate finite volume method for spatial discretization and the backward Euler scheme for first order accurate fully implicit time integration. The discrete form of Equations 2.1 - 2.4 that is obtained using this discretization approach is shown in Equation 3.1. In this equation, $\Delta \mathbf{Q}_{cl}$ is the change in conserved properties, $\mathbf{F}_{\hat{n},j}$ and $\mathbf{F}_{v\hat{n},j}$ are the inviscid and viscous fluxes normal to face j, and $\mathbf{S}_{cv,cl}$ is the vector of source terms. The time step is given by Δt and the time level is indicated by n. The volume of cell cl and the area of face j are given by V_{cl} and s_j, respectively. The right hand side of Equation 3.1 is known as the residual, and is given the symbol $\mathbf{R}_{cl}$. As a solution approaches steady-state, the residual should tend towards zero, which is a useful criterion for determining when to end a simulation. The conserved quantities, flux vectors in the x-direction, and source terms, are shown in Equations 3.2 and 3.3.

$$\frac{V_{cl}}{\Delta t}\Delta \mathbf{Q}_{cl} = \left[-\sum_{j \in cl} \left(\mathbf{F}_{\hat{n},j} - \mathbf{F}_{v\hat{n},j} \right) s_j + V_{cl}\mathbf{S}_{cv,cl} \right]^{n+1} = \mathbf{R}_{cl}^{n+1} \qquad (3.1)$$

$$\mathbf{Q} = \begin{Bmatrix} \rho_1 \\ \vdots \\ \rho_{ns} \\ \rho u \\ \rho v \\ \rho w \\ E \\ E_{ve} \end{Bmatrix} \qquad \mathbf{S}_{cv} = \begin{Bmatrix} \dot{\omega}_w \\ \vdots \\ \dot{\omega}_{ns} \\ 0 \\ 0 \\ 0 \\ 0 \\ \dot{\omega}_v \end{Bmatrix} \qquad \mathbf{F}_x = \begin{Bmatrix} \rho_1 u \\ \vdots \\ \rho_{ns} u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ (E+p)u \\ E_{ve} u \end{Bmatrix} \qquad (3.2)$$

$$\mathbf{F}_{v,x} = \left\{ \begin{array}{c} -J_{x,1} \\ \vdots \\ -J_{x,ns} \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ \tau_{xx}u + \tau_{xy}v + \tau_{xz}w - (q_{tr,x} + q_{ve,x} - \sum (J_{x,s}h_s)) \\ -q_{ve,x} - \sum (J_{x,s}e_{ve,s}) \end{array} \right\} \qquad (3.3)$$

The inviscid fluxes are computed using a modified form of Steger-Warming flux vector splitting [89] that uses a pressure switch to decrease the numerical dissipation when computing boundary layers [60, 34]. MUSCL variable extrapolation [100] with a correction for unstructured grids is used to obtain second-order spatial accuracy for the inviscid fluxes. The viscous fluxes are computed using a second-order accurate stencil that was originally developed for unstructured meshes and which makes use of both nodal and cell center values for derivative calculations [50].

In order to solve the set of discrete equations, the residual, $\mathbf{R}_{cl}$, is linearized, and the resulting linear system is solved using either a point or line implicit method [87]. At each step in this approach, the left-hand side of the linear system is reduced to diagonal or tridiagonal form by moving any off-diagonal terms to the right-hand side of the equation. The diagonal form of the left hand side results from using the point implicit approach, and the tridiagonal form results from using the line implicit approach. In either case, once the off-diagonal terms have been moved, a series of relaxation steps is performed, with the right-hand side of the equation updated at each step using the solution of the diagonal system. Using the recommendation of Wright [109], four relaxation steps are used. While this approach provides good steady state results, it is not strictly time accurate, and so throughout this work all reported LeMANS solutions are for a steady state flow field.

Boundary conditions in LeMANS are implemented using the "ghost-cell" method. In this approach, the code generates extra layers of cells around the boundaries of the domain, and the variables in these extra cells are set so that the correct fluxes are obtained at the actual domain boundaries. This allows the same method of flux calculations to be used at the boundaries as is used in the interior of the domain. In order to allow for second-order accurate boundary flux calculations, two layers of ghost cells are created. The ghost cells are not part of the linear system to be solved, but they are only used to compute fluxes at the boundaries for use in Equation 3.1. Only one set of ghost cells is used for computing both the inviscid and viscous fluxes, but the cells are populated with different values depending on which type of flux is being computed.

In the case where there is an ablating surface, a blowing boundary condition is implemented [62]. This allows a mass blowing rate to be specified along the boundary, which allows ablation products to be introduced into the flow field. For more details on the implicit implementation of boundary conditions, as well as a more detailed discussion of the discrete equations and numerical methods used in LeMANS, the reader is directed to Reference [87]. The effect of different boundary condition implementations on the order of accuracy of LeMANS is also discussed in Chapter IV.

3.3   Material Response

The following sections detail the implementation of the material response code. First, an overview is given of the spatial discretization technique used for the governing equations. Then, the details of the thermal and structural discretizations are discussed along with the numerical approaches used to solve the resulting sets of nonlinear equations.

3.3.1 The Control Volume Finite Element Method

The material response code uses a discretization technique known as the control volume finite element method (CVFEM) [11, 104, 15]. Similar to a finite element method, the CVFEM discretizes the computational domain into a number of elements of a specific shape, uses interpolation functions based on the elements, and assembles the discrete equations on an element by element basis. Rather than using a variational principle or the Galerkin technique to derive the discrete equations, however, the CVFEM uses a control volume approach [10]. In this respect, the CVFEM is similar to a node-centered finite volume scheme. Use of the CVFEM has been demonstrated for both thermal [15, 5] and elastic structural problems [9, 88], and it can be coupled in a straightforward manner with a finite volume CFD code [106].

The CVFEM uses two different meshes to develop a spatial discretization. The first mesh, referred to here as the primary mesh, is created by grid generation software and consists of triangular elements in two dimensions and tetrahedral elements in three dimensions. The second mesh, called the dual mesh, is created in the material response code by connecting element centroids, edge midpoints, and face centroids for tetrahedral elements in order to form a closed circuit around a given node. This leads to node-centered control volumes, and it is these control volumes over which the discretized equations are integrated. Figure 3.1 shows an example of the primary and dual meshes for a two dimensional domain, and Figure 3.2 shows the control volume contributions of a single element to the node i control volume in two and three dimensions.

Quantities are only known at the centroids of the dual mesh (nodes of the primary mesh), so in order to integrate the equations an assumption must be made about how a quantity varies across a control volume. In the implementation of the CVFEM used in this work, it is assumed that quantities vary linearly within each element of the primary mesh, and linear shape functions are used to interpolate the nodal values
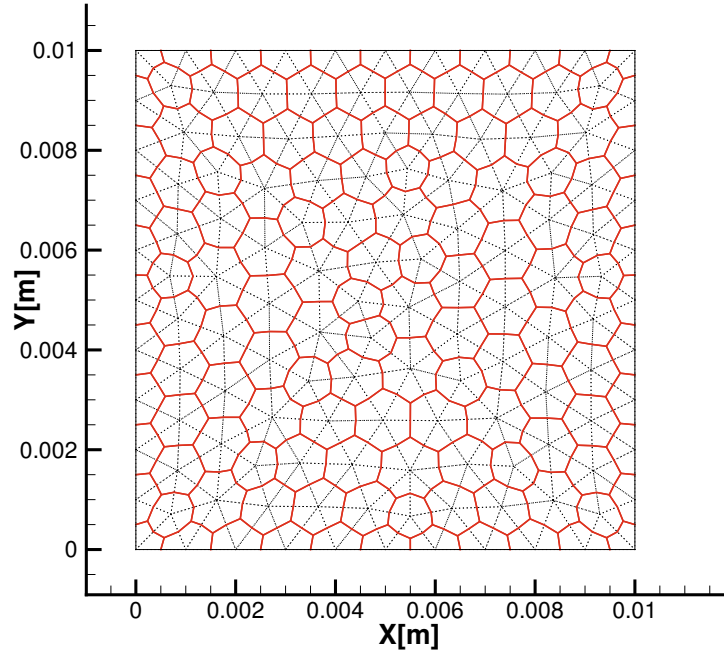
Figure 3.1: Primary (dotted lines) and dual (solid lines) meshes for the CVFEM in two dimensions.
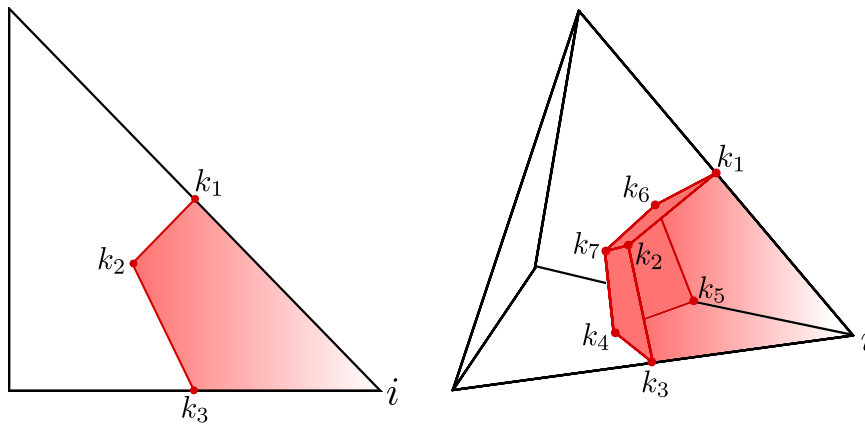
Figure 3.2: Control volume contribution of a single element to node i.

within an element. Interpolation for a quantity, θ, is performed within a reference element, and the result is mapped back to physical space. Equation 3.4 shows the mapping from reference to physical space for θ, where $\tilde{\theta}_i$ is the value at node i of an

element, ϕ_i is the shape function associated with node i, n is the number of nodes in an element, and ξ, η, and ζ are the coordinates of the reference element. The shape functions for triangular and tetrahedral elements are given by Equation 3.5. This process is shown schematically for a two dimensional element in Figure 3.3.

$$\theta(x,y,z) = \sum_{i=1}^{n} \tilde{\theta}_i \phi_i(\xi,\eta,\zeta) \tag{3.4}$$

$$triangles \begin{cases} \phi_1 = 1 - \xi - \eta & \phi_1 = 1 - \xi - \eta - \zeta \\ \phi_2 = \xi & \phi_2 = \xi \\ \phi_3 = \eta & \phi_3 = \zeta \\ & \phi_4 = \eta \end{cases} tetrahedra \tag{3.5}$$
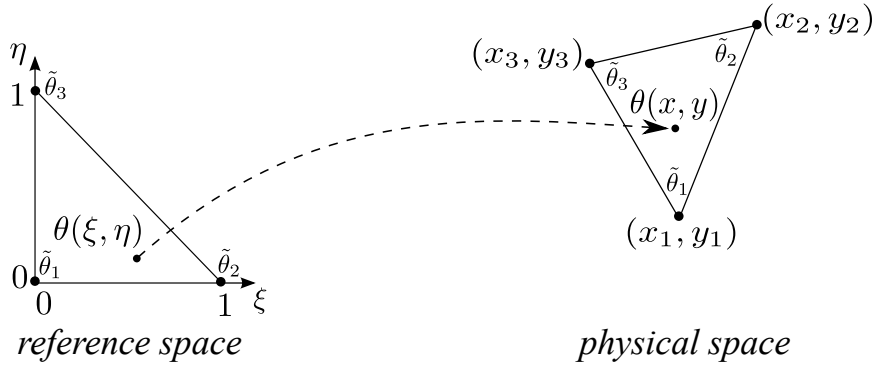
Figure 3.3: Interpolation and mapping of a quantity from reference space to physical space.

In order to compute the contributions of an element to a control volume, the element nodal values are interpolated to the corners of the dual mesh within each element, which correspond to the points labeled k in Figure 3.2. The necessary integrals over the dual mesh edges and volumes are then carried out analytically in reference element space and mapped back to the physical space using Equation 3.6 where J is the determinant of the Jacobian, which is constant for linear elements.

31

The physical volume is denoted by Ω, and the reference volume by Ω_r. Equation 3.7 shows the corresponding mapping for derivatives. This approach assembles the global system of equations on an element by element basis, with each element that is connected to node i contributing a portion of the node i control volume.

$$\int_\Omega f(\mathbf{x}) \mathrm{d}\Omega = \int_{\Omega_r} f(\mathbf{x}(\boldsymbol{\xi})) J \mathrm{d}\Omega_\mathrm{r} = J \int_{\Omega_r} f(\mathbf{x}(\boldsymbol{\xi})) \mathrm{d}\Omega_\mathrm{r} \tag{3.6}$$

$$\frac{\partial f}{\partial \mathbf{x}} = \frac{\partial f}{\partial \boldsymbol{\xi}} \frac{\partial \boldsymbol{\xi}}{\partial \mathbf{x}} = \frac{\partial f}{\partial \boldsymbol{\xi}} \mathbf{J}^{-1} \tag{3.7}$$

3.3.2 Thermal Response

The structural thermal response is governed by Equation 2.11, repeated below for convenience.

$$\int_{\partial\Omega} \tilde{q}_i \hat{n}_i \ \mathrm{d}\partial\Omega \ - \ \int_{\partial\Omega} \rho h v_{cs_i} \hat{n}_i \ \mathrm{d}\partial\Omega \ + \ \frac{d}{dt} \int_\Omega \rho e \mathrm{d}\Omega \ = \ \int_\Omega Q \mathrm{d}\Omega \tag{3.8}$$

Using the CVFEM approach described in the previous section and making use of Fourier's law leads to the discrete form of Equation 3.8 shown in Equation 3.9. In this equation, $\mathbf{H_t}$ contains the shape functions, $\mathbf{L_t}$ contains the differential operators, $\hat{\mathbf{n}}$ is the normal vector of the control surface, $\underline{\mathbf{k}}$ is the thermal conductivity tensor, and variables with a tilde represent nodal values. The term $\tilde{\mathbf{v}}_\mathbf{n}$ in the grid convection integral is the vector of nodal values of grid velocity in the direction normal to the control surface, and it is computed as $\mathbf{v_{cs_i}} \cdot \hat{\mathbf{n}}$ where $\mathbf{v_{cs_i}}$ is the grid velocity vector at node i. Equations 3.10 and 3.11 show the $\mathbf{L_t}$ vector and a vector of generic nodal values, $\tilde{\boldsymbol{\theta}}$, and Equation 3.12 shows the $\mathbf{H_t}$ matrix. The $\mathbf{H_t}$ and $\tilde{\boldsymbol{\theta}}$ arrays are written for an element containing n nodes.

The integral domains in Equation 3.9 are the full control volumes around each node, i, however, they are computed on an element by element basis so the integral over each control volume is actually split into several segments corresponding to the

contributions from each element that is connected to node i. The element integrals are computed analytically in reference space and then mapped back to physical space. More details on integration, including the analytic results, can be found in Appendix B.

$$\underbrace{\int_{\partial\Omega_i} -\hat{\mathbf{n}}^{\mathbf{T}}\underline{\mathbf{k}}\mathbf{L_t}\mathbf{H_t}^{T}\tilde{\mathbf{T}}\ \mathrm{d}\partial\Omega}_{\text{conduction}} - \underbrace{\int_{\partial\Omega_i} \left(\mathbf{H}^{T}\tilde{\mathbf{h}}\right)\left(\mathbf{H}^{T}\tilde{\mathbf{v}_{\mathbf{n}}}\right)\ \mathrm{d}\partial\Omega}_{\text{grid convection}}$$

$$+ \frac{d}{dt}\underbrace{\int_{\Omega_i} \rho\mathbf{H}^{T}\tilde{\mathbf{e}}\ \mathrm{d}\Omega}_{\text{energy content}} = \underbrace{\int_{\Omega_i} \mathbf{H}^{\mathbf{T}}\tilde{\mathbf{Q}}\mathrm{d}\Omega}_{\text{source}} \qquad (3.9)$$

$$\mathbf{L}_t = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{pmatrix} \quad (3.10) \qquad \tilde{\boldsymbol{\theta}} = \begin{pmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{pmatrix} \quad (3.11) \qquad \mathbf{H}_t = \begin{pmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_n \end{pmatrix} \quad (3.12)$$

Equation 3.9 represents a set of nonlinear equations since the material properties are functions of temperature. The Newton-Raphson method is used to solve this nonlinear system, and therefore it is necessary to linearize each term in Equation 3.9. To accomplish this, each term is linearized using a Taylor series expansion in iteration space. If the conduction term in Equation 3.9 is denoted by Q_{cond_i}, then it can be linearized as shown in Equation 3.13.

$$^{\nu+1}Q^{n+1}_{cond_i} = {}^{\nu}Q^{n+1}_{cond_i} + \mathbf{J}_{cond_i}\Delta\tilde{\mathbf{T}}_i \qquad (3.13)$$

In this equation $\mathbf{J}_{cond_i}$ is the vector of partial derivatives of Q_{cond_i} with respect to the temperature at node j, where j represents all of the nodes that are connected to node i. The time level is denoted by n, the nonlinear iteration number is given by ν, and the incremental change in temperature is given by $\Delta\tilde{\mathbf{T}}_{\mathbf{i}}$. In a similar manner,

33

the grid convection and source terms can be linearized as shown in Equations 3.14 and 3.15.

$$^{\nu+1}Q_{gc_i}^{n+1} = {}^{\nu}Q_{gc_i}^{n+1} + \mathbf{J}_{gc_i}\Delta\tilde{\mathbf{T}}_i \tag{3.14}$$

$$^{\nu+1}Q_{src_i}^{n+1} = {}^{\nu}Q_{src_i}^{n+1} + \mathbf{J}_{src_i}\Delta\tilde{\mathbf{T}}_i \tag{3.15}$$

For the energy content term, the time derivative must also be taken into account. This is done using a backward Euler scheme, which leads to first-order accurate implicit time integration. The time derivative can be written as shown in Equation 3.16.

$$\frac{\mathrm{d}Q_{ec_i}}{\mathrm{dt}} = \frac{Q_{ec_i}^{n+1} - Q_{ec_i}^{n}}{\Delta t} \tag{3.16}$$

Expanding this equation in iteration space leads to the linearized version shown in Equation 3.17.

$$\frac{\mathrm{d}Q_{ec_i}}{\mathrm{dt}} = \frac{1}{\Delta t}\left({}^{\nu}Q_{ec_i}^{n+1} + {}^{\nu}\mathbf{J}_{ec_i}\Delta\tilde{\mathbf{T}}_i - Q_{ec_i}^{n}\right) \tag{3.17}$$

Equations 3.13 - 3.17 are written for a control volume, i, but all of the individual control volumes can be written as an $N \times N$ system of equations, where N is the number of nodes in the primary mesh. Doing this, and combining Equations 3.13, 3.14, 3.15, and 3.17, leads to Equation 3.18.

$$\left(\mathbf{J}_{cond} - \mathbf{J}_{gc} - \mathbf{J}_{src} + \frac{1}{\Delta t}\mathbf{J}_{ec}\right)\Delta\tilde{\mathbf{T}} = \mathbf{Q}_{cond} - \mathbf{Q}_{ec} - \mathbf{Q}_{src} + \frac{1}{\Delta t}\left(\mathbf{Q}_{ec} - \mathbf{Q}_{ec}^{n}\right) \tag{3.18}$$

This further simplifies to Equation 3.19, which is the $N \times N$ sparse linear system that is solved at each iteration, ν, of the Newton-Raphson method. In order to reduce the amount of memory needed to store this system, the compressed sparse row (CSR) storage format [86] is used. Equation 3.19 is solved using the Generalized Minimal Residual (GMRES) method [85]. To improve the convergence properties of the GMRES solver, restarting is used along with either a Gauss-Seidel or ILU(0)

preconditioner [86].

$$^{\nu}\mathbf{J}_{all}\Delta\tilde{\mathbf{T}} = {}^{\nu}\mathbf{Q}_{all} \tag{3.19}$$

The thermal material response code supports several different boundary condition types including specified temperature, specified heat flux, radiation, aerodynamic heating, and thermochemical ablation. In all cases, the necessary boundary values are prescribed at the boundary nodes, and values are assumed to vary linearly along the boundary. Therefore, the boundary condition implementation is second-order accurate and consistent with the internal discretization.

Specified temperature boundary conditions are applied at a node, i, by modifying the ith row of both $\mathbf{J}_{all}$ and $\mathbf{Q}_{all}$. The diagonal term of row i in $\mathbf{J}_{all}$ is set to 1.0, and all other terms in row i are set to 0.0. In $\mathbf{Q}_{all}$, row i is set to $\Delta T = T_{bnd} - T_w$ where T_{bnd} is the desired boundary temperature and T_w is the current boundary temperature at iteration ν of the Newton-Raphson method.

The specified heat flux boundary condition is applied by adding the desired heat flux to $\mathbf{Q}_{all}$, so that the right hand side of Equation 3.18 becomes:

$$\text{RHS} = \mathbf{Q}_{cond} - \mathbf{Q}_{gc} - \mathbf{Q}_{src} + \frac{1}{\Delta t}\left(\mathbf{Q}_{ec} - \mathbf{Q}_{ec}^n\right) + \mathbf{Q}_{sf} \tag{3.20}$$

where $\mathbf{Q}_{sf} = \int_{\partial\Omega}\tilde{\mathbf{q}}_{bnd}\mathrm{d}\partial\Omega$. No changes are made to $\mathbf{J}_{all}$. Also, due to how the heat flux boundary condition is implemented, an adiabatic condition where $\tilde{\mathbf{q}}_{bnd} = 0.0$ requires no modification of the linear system.

The aerodynamic heating, radiation, and thermochemical ablation boundary conditions are typically applied simultaneously at a given boundary, and they can be used to couple the thermal code with a CFD code. The surface energy balance for the coupled case is shown in Equation 3.21.

$$\underbrace{-\underline{\mathbf{k}}\nabla T \cdot \hat{\mathbf{n}}}_{\text{Conduction}} = \underbrace{q_{ah}}_{\text{Convection}} - \underbrace{\epsilon\sigma(T_w^4 - T_\infty^4)}_{\text{Radiation}} - \underbrace{\rho_s\dot{s}h_w\hat{\mathbf{n}}}_{\text{Ablation}} \tag{3.21}$$

The aerodynamic heating boundary condition is implemented in two different ways depending on whether or not the thermal code is coupled to a CFD code. In the uncoupled case, the external flow conditions are user-defined, and the aerodynamic heat flux, q_{ah}, is given by Equation 3.22

$$q_{ah} = C_h \left(h_r - h_w \right) \tag{3.22}$$

where C_h is a convective heat transfer coefficient, h_w is the specific enthalpy of the gases adjacent to the wall, and h_r is the recovery enthalpy of the flow. The wall enthalpy is computed by an equilibrium chemistry code for various pressures and temperatures, and then input as a thermochemistry table to the thermal code. The density, boundary layer edge velocity, and Stanton number are also user inputs that rely on an assumed flow field and can vary in time. More details on the aerodynamic heating boundary condition for the uncoupled case can be found in Amar [5]. For the coupled case, the heat flux computed by the CFD code is applied directly in the thermal solver, so $q_{ah} = q_{CFD}$. The CFD heat flux can include both conduction and mass diffusion heat fluxes and is given by Equations 2.8 - 2.10 for LeMANS.

The radiation boundary condition accounts for the radiative heat flux from the material to the flow field, and is given by Equation 3.23.

$$q_{rad} = \epsilon \sigma \left(T_w^4 - T_\infty^4 \right) \tag{3.23}$$

In this equation, ϵ is the emissivity of the material, σ is the Stefan-Boltzmann constant, T_w is the current boundary temperature, and T_∞ is the temperature of the environment to which energy is radiating.

When modeling a problem that includes ablation, it is also necessary to include the thermochemical ablation boundary condition, which is shown in Equation 3.24

where ρ_s is the density of the solid material at the wall, $\dot{s}$ is the surface recession rate, and h_w is the same wall enthalpy as is used for the uncoupled aerodynamic heating boundary condition.

$$q_{abl} = \rho_s \dot{s} h_w \qquad (3.24)$$

The surface recession rate is computed based on a mass blowing rate, $\dot{m}_{abl}$, as shown in Equation 3.25.

$$\dot{s} = \frac{\dot{m}_{abl}}{\rho_s} \qquad (3.25)$$

The mass blowing rate and wall enthalpy are obtained from a thermochemical table [5] that is computed assuming an equilibrium chemistry solution of the ablation materials and the flow species. It is also possible, and more physically realistic, to use a finite rate surface chemistry (FRSC) module as part of LeMANS [4] to compute the mass flux directly based on a nonequilibrium surface chemistry solution. Use of the FRSC module is beyond the scope of this work, however, and all reported results that include ablation use the thermochemical, or B', table approach.

In cases where there is surface recession, it is necessary to deform the mesh so that the numerical problem domain is coincident with the physical problem domain. In order to move an unstructured mesh in a consistent and general manner, the mesh is treated as a linear elastic solid, and the elastic solid mechanics equations are solved with zero body forces [59, 48, 29]. The equations for a two-dimensional domain written in terms of the nodal displacements u and v are shown in Equation 3.26.

$$
\begin{aligned}
&\frac{1}{1-2\nu}\frac{\partial}{\partial x}\left[(1-\nu)\frac{\partial u}{\partial x} + \nu\left(\frac{\partial v}{\partial y}\right)\right] + \frac{1}{2}\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) = 0 \\
&\frac{1}{2}\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) + \frac{1}{1-2\nu}\frac{\partial}{\partial y}\left[(1-\nu)\frac{\partial v}{\partial y} + \nu\left(\frac{\partial u}{\partial x}\right)\right] = 0
\end{aligned}
\qquad (3.26)
$$

Poisson's ratio, ν, is the only material property that is needed to solve these equations. Since the solution of Equation 3.26 is only used to move the mesh and

the stresses in the fictitious mesh "material" are not important, Poisson's ratio can be arbitrarily chosen. Following the work of Lynch and O'Neill [59], ν is taken to be 0.0 for this study.

The solid mechanics equations are solved using the GMRES method with specified nodal displacements at the boundaries. At ablating boundaries, the nodal displacement over a time step, Δt, is found using the surface recession rate calculated in Equation 3.25, and is given by Equation 3.27 where n_x, n_y are the components of the surface normal vector, $\hat{\mathbf{n}}$.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \dot{s}\Delta t n_x \\ \dot{s}\Delta t n_y \end{bmatrix} \tag{3.27}$$

All non-ablating boundaries are assumed to have zero displacement. Non-ablating boundary nodes near an ablating surface, however, are allowed to slide along the boundary in order to prevent elements near the ablating surface from becoming highly skewed.

Once q_{ah}, q_{rad}, and q_{abl} have been computed for each of the boundary nodes, they are added to $\mathbf{Q}_{all}$ in the same way as the specified heat flux boundary condition. In the case of the radiation and thermochemical ablation boundary conditions, and the uncoupled aerodynamic heating boundary condition, it is also necessary to modify $\mathbf{J}_{all}$ by adding $\mathbf{J}_{rad}$, $\mathbf{J}_{abl}$, and $\mathbf{J}_{ah}$, which are the Jacobians of the different boundary heat fluxes. The Jacobian terms for each of the boundary conditions can be found in Amar [5].

3.3.3 Elastic Response

The structural elastic response is governed by Equation 2.25, repeated below for convenience.

$$\int_{\Omega_0} \rho \ddot{u}_i \mathrm{d}\Omega_0 - \int_{\Omega_0} \frac{\partial}{\partial x_{0_j}} \left(F_{ik} \tilde{\sigma}_{jk} \right) \mathrm{d}\Omega_0 = 0 \tag{3.28}$$

38

Equation 3.28 can be written in a more convenient form for discretization by the CVFEM by applying the divergence theorem to the second integral to obtain Equation 3.29.

$$\int_{\Omega_0} \rho \ddot{u}_i \mathrm{d}\Omega_0 - \int_{\partial\Omega_0} F_{ik} \tilde{\sigma}_{jk} \hat{n}_j \mathrm{d}\partial\Omega_0 = 0 \tag{3.29}$$

Equation 3.29 can then be discretized using the CVFEM, which leads to the discrete equation shown in Equation 3.30.

$$\int_{\Omega_0} \rho \mathbf{H}_s \ddot{\tilde{\mathbf{u}}} \, \mathrm{d}\Omega_0 - \int_{\partial\Omega_0} \mathbf{F}\hat{\mathbf{N}}\mathbf{D} \left(\mathbf{L_s H_s} + \frac{1}{2}\mathbf{AGH_s} \right) \tilde{\mathbf{u}} \, \mathrm{d}\partial\Omega_0 =$$
$$- \int_{\partial\Omega_0} \mathbf{F}\hat{\mathbf{N}}\mathbf{D}\tilde{\boldsymbol{\epsilon}}^0 \, \mathrm{d}\partial\Omega_0 - \int_{\partial\Omega_0} \mathbf{F}\hat{\mathbf{N}}\mathbf{D}\tilde{\boldsymbol{\epsilon}}^{th} \, \mathrm{d}\partial\Omega_0 + \int_{\partial\Omega_{b0}} \mathbf{F}\hat{\mathbf{N}}\mathbf{t_p} \, \mathrm{d}\partial\Omega_{b0} \tag{3.30}$$

Similar to Equation 3.9, $\mathbf{H_s}$ contains the shape functions, $\mathbf{L_s}$ contains the differential operators, and variables with a tilde are nodal values. The constitutive relations from Equation 2.27 are contained in $\mathbf{D}$, $\mathbf{F}$ is the deformation gradient tensor from Equation 2.23, and the product $\mathbf{AG}$ contains the nonlinear portion of the Green-Lagrange strain tensor. The $\hat{\mathbf{N}}$ matrix contains the components of the control surface normal vector, and t_p is the vector of prescribed traction values for applying a traction boundary condition. Integration is again performed analytically in reference elements and then mapped back to physical space, however, for the elastic response a total Lagrangian formulation is used, so the integration is always over the initial undeformed geometry. The definitions of $\mathbf{H_s}$, $\mathbf{L_s}$, $\hat{\mathbf{N}}$, $\mathbf{A}$, $\mathbf{G}$, and the nodal displacement vector, $\tilde{\mathbf{u}}$, are shown in Equations 3.31 - 3.36.

$$\mathbf{H_s} = \begin{bmatrix} \phi_1 & 0 & 0 & \phi_2 & 0 & 0 & \cdots & \phi_n & 0 & 0 \\ 0 & \phi_1 & 0 & 0 & \phi_2 & 0 & \cdots & 0 & \phi_n & 0 \\ 0 & 0 & \phi_1 & 0 & 0 & \phi_2 & \cdots & 0 & 0 & \phi_n \end{bmatrix} \tag{3.31}$$

$$\mathbf{L_s} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \\ \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \\ 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \end{bmatrix} \quad (3.32)$$

$$\hat{\mathbf{N}} = \begin{bmatrix} n_x & 0 & 0 & n_y & n_z & 0 \\ 0 & n_y & 0 & n_x & 0 & n_z \\ 0 & 0 & n_z & 0 & n_x & n_y \end{bmatrix} \quad (3.33)$$

$$\mathbf{A} = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial v}{\partial x} & \frac{\partial w}{\partial x} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\partial u}{\partial y} & \frac{\partial v}{\partial y} & \frac{\partial w}{\partial y} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{\partial u}{\partial z} & \frac{\partial v}{\partial z} & \frac{\partial w}{\partial z} \\ \frac{\partial u}{\partial y} & \frac{\partial v}{\partial y} & \frac{\partial w}{\partial y} & \frac{\partial u}{\partial x} & \frac{\partial v}{\partial x} & \frac{\partial w}{\partial x} & 0 & 0 & 0 \\ \frac{\partial u}{\partial z} & \frac{\partial v}{\partial z} & \frac{\partial w}{\partial z} & 0 & 0 & 0 & \frac{\partial u}{\partial x} & \frac{\partial v}{\partial x} & \frac{\partial w}{\partial x} \\ 0 & 0 & 0 & \frac{\partial u}{\partial z} & \frac{\partial v}{\partial z} & \frac{\partial w}{\partial z} & \frac{\partial u}{\partial y} & \frac{\partial v}{\partial y} & \frac{\partial w}{\partial y} \end{bmatrix} \quad (3.34)$$

$$\mathbf{G} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial x} & 0 \\ 0 & 0 & \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} & 0 & 0 \\ 0 & \frac{\partial}{\partial z} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \end{bmatrix} \quad (3.35)$$

$$\tilde{\mathbf{u}} = \begin{pmatrix} u_1 \\ v_1 \\ w_1 \\ u_2 \\ v_2 \\ w_2 \\ \vdots \\ u_n \\ v_n \\ w_n \end{pmatrix} \quad (3.36)$$

Equation 3.30 can be rewritten more simply using Equations 3.37 - 3.39 which

leads to Equation 3.40.

$$\mathbf{M} = \int_{\Omega_0} \rho \mathbf{H_s} \, \mathrm{d}\Omega_0 \tag{3.37}$$

$$\mathbf{K} = - \int_{\partial\Omega_0} \mathbf{F\hat{N}D} \left(\mathbf{L_s H_s} + \frac{1}{2} \mathbf{AGH_s} \right) \mathrm{d}\partial\Omega_0 \tag{3.38}$$

$$\mathbf{R} = - \int_{\partial\Omega_0} \mathbf{F\hat{N}D}\tilde{\boldsymbol{\epsilon}}^0 \, \mathrm{d}\partial\Omega_0 - \int_{\partial\Omega_0} \mathbf{F\hat{N}D}\tilde{\boldsymbol{\epsilon}}^{th} \, \mathrm{d}\partial\Omega_0 + \int_{\partial\Omega_0^t} \mathbf{F\hat{N}t_p} \, \mathrm{d}\partial\Omega_0^t \tag{3.39}$$

$$\mathbf{M}\tilde{\ddot{u}} + \mathbf{K}\tilde{u} = \mathbf{R} \tag{3.40}$$

Due to the use of the nonlinear Green-Lagrange strain displacement relation, Equation 3.40 must be solved iteratively in a similar fashion to the thermal equation. In order to develop the iterative version of Equation 3.40, the acceleration term can be neglected and a Taylor series expansion of Equation 3.41 is taken about u^* as shown in Equation 3.42 assuming that $\mathbf{R}$ is not a function of displacement.

$$\left. \begin{aligned} f\left(u^*\right) &= \mathbf{R} - \mathbf{K}\hat{u} = 0 \\ &= \mathbf{R} - \Psi\left(u^*\right) = 0 \end{aligned} \right\} \tag{3.41}$$

$$\left. \begin{aligned} f\left({}^{\nu+1}u^*\right) &= f\left({}^{\nu}u\right) + \frac{\partial f}{\partial u}|_{{}^{\nu}u}\left(u^* - {}^{\nu}u\right) \\ \Rightarrow \frac{\partial f}{\partial u}|_{{}^{\nu}u}\Delta u &= f\left(u^*\right) - f\left({}^{\nu}u\right) \\ &= -R + \Psi\left({}^{\nu}u\right) \\ \Rightarrow \frac{\partial \Psi}{\partial u}|_{{}^{\nu}u}\Delta u &= R - \Psi\left({}^{\nu}u\right) \\ \Rightarrow \mathbf{K}_T \Delta u &= R - \Psi\left({}^{\nu}u\right) \end{aligned} \right\} \tag{3.42}$$

The matrix $\mathbf{K}_T$ is known as the tangent stiffness matrix and contains the derivatives of the nodal point forces with respect to the nodal point displacements [13]. Following the approach of Fallah [39], the tangent stiffness matrix can be written as

41

shown in Equation 3.43.

$$\frac{\partial \Psi}{\partial \mathbf{u}} = \mathbf{K}_T = -\int_{\partial \Omega_0} \bar{\mathbf{N}} \mathbf{P} \mathbf{G} \, \mathrm{d}\partial \Omega_0 - \int_{\partial \Omega_0} \mathbf{F} \hat{\mathbf{N}} \mathbf{D} \mathbf{B} \, \mathrm{d}\partial \Omega_0 \tag{3.43}$$

The matrix, $\mathbf{B}$, is given by $\mathbf{B} = \mathbf{L_s} \mathbf{H_s} + \mathbf{A} \mathbf{G} \mathbf{H_s}$, and the matrices $\mathbf{P}$ and $\bar{\mathbf{N}}$ are shown in Equations 3.44 and 3.45 where $\mathbf{I}$ is the 3×3 identity matrix.

$$\mathbf{P} = \begin{bmatrix} \sigma_{xx}\mathbf{I} & \sigma_{xy}\mathbf{I} & \sigma_{xz}\mathbf{I} \\ \sigma_{xy}\mathbf{I} & \sigma_{yy}\mathbf{I} & \sigma_{yz}\mathbf{I} \\ \sigma_{zx}\mathbf{I} & \sigma_{zy}\mathbf{I} & \sigma_{zz}\mathbf{I} \end{bmatrix} \tag{3.44}$$

$$\bar{\mathbf{N}} = \begin{bmatrix} n_x & 0 & 0 & n_y & 0 & 0 & n_z & 0 & 0 \\ 0 & n_x & 0 & 0 & n_y & 0 & 0 & n_z & 0 \\ 0 & 0 & n_x & 0 & 0 & n_y & 0 & 0 & n_z \end{bmatrix} \tag{3.45}$$

Using the tangent stiffness matrix, Equation 3.40 can be rewritten as shown in Equation 3.46. While this formulation accounts for the nonlinearity of the governing equation, it is still necessary to write the acceleration term in terms of incremental displacements in order to solve the equation. This is performed by selecting a time integration scheme, which assumes a relationship between the nodal acceleration and displacement.

$$\mathbf{M}\ddot{\tilde{\boldsymbol{u}}} + \mathbf{K}_T \Delta \tilde{\mathbf{u}} = \mathbf{R} - \mathbf{K}\tilde{\boldsymbol{u}} \tag{3.46}$$

Time integration is performed using the Newmark-beta method [73], which makes use of the assumptions shown in Equations 3.47 and 3.48, written for a node, i. The Newmark-beta method describes a family of integration techniques depending on the values of β and γ. In this work, $\beta = \frac{1}{2}$ and $\gamma = \frac{1}{4}$, which leads to a second order

accurate implicit time integration scheme.

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n + \dot{\mathbf{u}}_i^n \Delta t + \left[(1 - 2\beta) \, \ddot{\mathbf{u}}_i^n + 2\beta \ddot{\mathbf{u}}_i^{n+1} \right] \frac{\Delta t^2}{2} \tag{3.47}$$

$$\dot{\mathbf{u}}_i^{n+1} = \dot{\mathbf{u}}_i^n + \left[(1 - \gamma) \, \ddot{\mathbf{u}}_i^n + \gamma \ddot{\mathbf{u}}_i^{n+1} \right] \Delta t \tag{3.48}$$

Solving Equation 3.47 for $\ddot{\mathbf{u}}^{n+1}$, and using the fact that $^{\nu+1}\mathbf{u}_i^{n+1} = {}^{\nu}\mathbf{u}_i^{n+1} + \Delta \mathbf{u}_i$ allows the acceleration at time $n + 1$ to be written as shown in Equation 3.49.

$$^{\nu+1}\ddot{\mathbf{u}}_i^{n+1} = \frac{1}{\beta \Delta t^2} \left({}^{\nu}\mathbf{u}_i^{n+1} - \mathbf{u}_i^n + \Delta \mathbf{u}_i - \Delta t \dot{\mathbf{u}}_i^n \right) - \frac{1 - 2\beta}{2\beta} \ddot{\mathbf{u}}_i^n \tag{3.49}$$

Substituting Equation 3.49 into Equation 3.46 and rearranging terms leads to Equation 3.50, which is the linear system that needs to be solved for the time accurate solution of the nonlinear Cauchy's equation. This can be written in a simplified form similar to the thermal equation as shown in Equation 3.51.

$$\left(\mathbf{M} \frac{1}{\beta \Delta t^2} + \mathbf{K}_T \right) \Delta \tilde{\mathbf{u}} = \mathbf{R} - \mathbf{K}^{\nu+1} \tilde{\mathbf{u}}^{n+1} +$$
$$\mathbf{M} \left(\frac{1 - 2\beta}{2\beta} \ddot{\tilde{\mathbf{u}}}^n + \frac{1}{\beta \Delta t} \dot{\tilde{\mathbf{u}}}^n + \frac{1}{\beta \Delta t^2} \tilde{\mathbf{u}}^n - \frac{1}{\beta \Delta t^2} {}^{\nu} \tilde{\mathbf{u}}^{n+1} \right) \tag{3.50}$$

$$\hat{\mathbf{K}} \Delta \tilde{\mathbf{u}} = \hat{\mathbf{R}} \tag{3.51}$$

There are two different types of boundary conditions that can be applied to the above equations: specified displacement and specified traction. These can be written as shown in Equations 3.52 and 3.53 where u_{p_i} and t_{p_i} are the specified displacement and traction values and $\mathbf{u_i}$ is the current displacement. The $F_{ik} \tilde{\sigma}_{jk} \hat{n}_j$ term represents the current force vector at the boundary expressed in terms of the initial undeformed geometry. The initial geometry is used in order to be consistent with the total Lagrangian formulation. Boundary segments where the different boundary conditions

are applied are indicated by Ω_0^u and Ω_0^t.

$$u_i = u_{p_i} \quad \text{on } \Omega_0^u \tag{3.52}$$

$$t_{p_i} = F_{ik}\tilde{\sigma}_{jk}\hat{n}_j \quad \text{on } \Omega_0^t \tag{3.53}$$

Specified displacement boundary conditions are implemented in the same manner as the specified temperature boundary conditions for the thermal equation. The $\Delta\tilde{\mathbf{u}}$ needed to produce the desired displacement at node i is specified in the ith row of $\hat{\mathbf{R}}$, and the corresponding row in $\hat{\mathbf{K}}$ is modified to be 1.0 on the diagonal and 0.0 off the diagonal.

The specified traction boundary condition is implemented by prescribing the terms of the stress tensor along the boundary. This, along with the surface normal vector and the deformation gradient tensor, allows the traction vector at the boundary to be computed. The traction vector is then added into the $\mathbf{R}$ vector as shown in Equation 3.39. If the structural code is coupled with a CFD code, then a specified traction boundary condition is used at the fluid-solid interface, and the pressure and viscous shear stress components computed in the CFD code are combined into a stress tensor and applied as the boundary conditions.

The Newton-Raphson method is used to iteratively solve the nonlinear governing equations. For the elastic response, the standard Newton-Raphson method is combined with a line search and backtracking technique [79] in order to improve the global convergence and robustness of the method. This approach proved very useful for the solution of the elastic equations, but it was not found to be necessary for the thermal equations.

In a similar manner to the thermal code, the linear system in Equation 3.51 is stored in CSR format. However, since the displacements in each direction are part of the solution, the size of the system is now $(N_D N) \times (N_D N)$, where N_D is the number

of spatial dimensions in the problem. Once the linear system is assembled and the boundary conditions applied, a preconditioned, restarted GMRES method is used to solve the system.

3.4   Aerothermoelastic Coupling

In order to solve an aerothermoelastic problem, it is necessary to couple the flow, material thermal response, and material structural response solutions. The coupling process can in general take either a monolithic or a partitioned approach. In the monolithic approach all of the governing equations for the different physical processes (flow, thermal, structural) are combined into a single framework and solved simultaneously. This approach is well suited to problems where there is very strong interaction between the flow and solid, and it can often be more stable than a partitioned scheme [44]. There are several drawbacks, however, including the need to develop a highly specialized code, the necessity of using the same time step for the different physical subsystems, and the possibility of ill-conditioned system matrices [49]. The partitioned approach, on the other hand, solves the fluid and structural problems separately and links them through boundary conditions. One advantage of the partitioned approach is that separate off-the-shelf codes can be used for each component problem, and a single specialized code does not need to be developed [51]. Also, the partitioned approach allows for the most appropriate time step, discretization techniques, and numerical methods to be used for each component problem. The partitioned approach is adopted in this work since the problems being considered have widely varying time scales between the fluid and structure, and because it is desirable to use the already existing CFD code, LeMANS.

The coupling framework that is developed assumes a quasi-static relationship between the fluid and structural codes, where the flow is considered to be steady relative to the structure. This allows steady state solutions from LeMANS to be

used, but it does limit the applicability of the coupling procedure to cases where the quasi-static assumptions are valid. In particular, it is assumed that the characteristic time scales of the flow problem are much faster than the thermal and elastic response times of the structure, and therefore the flow can be assumed steady relative to the structure. This assumption greatly reduces the computational load since a time-accurate flow solution is not required. In some cases it is also possible to assume that the elastic response time of a structure is much faster than the thermal response time, but still much slower than the fluid response time. In this case the elastic response can be solved assuming a static rather than dynamic problem.

The overall coupling procedure between the CFD and computational structural dynamics (CSD) codes can be summarized as shown in Figure 3.4. This method splits
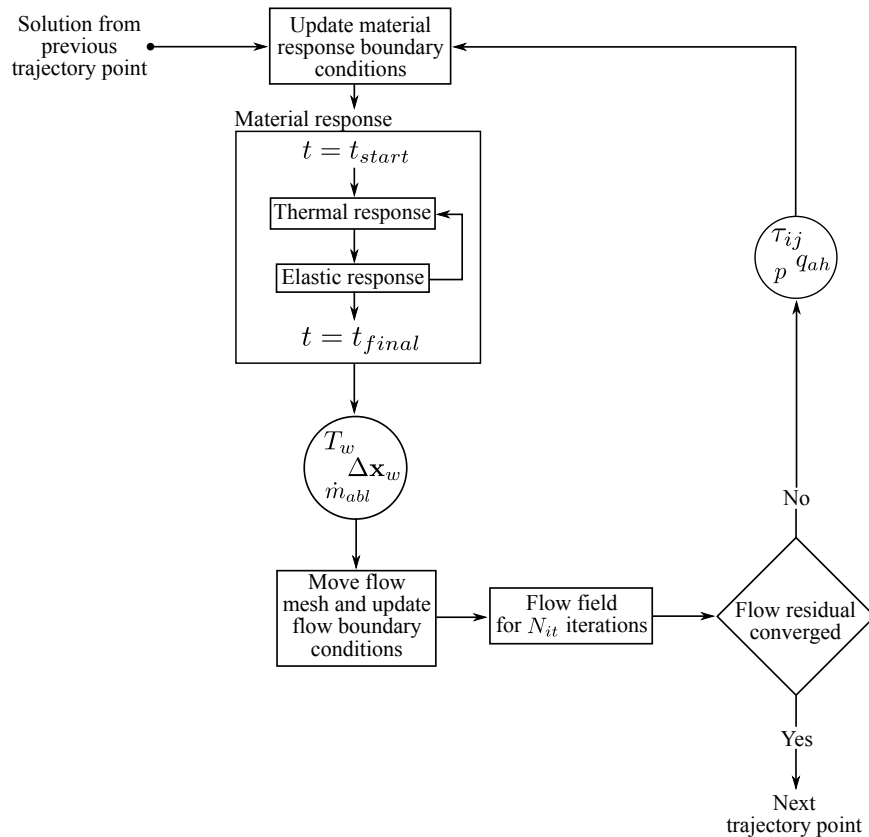
Figure 3.4: Flowchart for the aerothermoelastic coupling procedure.

a coupled problem into a series of trajectory points. These trajectory points may be

46

actual points in time along a vehicle's flight path, or they may simply be user-defined points that are chosen in order to split a problem with a long run time into a series of shorter length problems. In either case, the first trajectory point consists of a flow field only solution, and no material response is included. The next trajectory point uses the flow-only solution from the first point to initialize the flow field, and the solid material is initialized to a constant temperature and zero stress and strain. The material response is then called as a subroutine from LeMANS, and is run in a time accurate fashion from the start time to the end time of the current trajectory point.

The necessary boundary conditions for the structural thermal and elastic solvers are passed from LeMANS to the material response code. The updated wall temperature, T_w, wall displacement $\Delta \mathbf{x}_w$, and mass blowing rate, $\dot{m}_{abl}$, are passed back to LeMANS as new flow boundary conditions, and the flow field is advanced for some number of iterations, N_{it}. After N_{it} flow iterations, the material response is called again using the updated flow solution to provide new boundary conditions for the thermal and elastic solvers. This coupling procedure is continued until the flow field has reached a steady state and the flow residual has converged. If there are more points in the trajectory being modeled, then the next point, $i+1$, is initialized using the flow and solid solutions from the previous point, i. In all cases the flow is assumed to be steady, and the thermal material response is time accurate. The elastic material response may be static or dynamic depending on the problem. The thermal structural boundary conditions are assumed to vary linearly in time during each trajectory point with the initial value being the flow field solution from the previous trajectory point and the final value being the current flow field solution. The elastic structural boundary conditions vary linearly in time for a dynamic simulation, but for a static simulation only the current flow solution is used. The static boundary conditions are applied incrementally in order to improve convergence [22].

The boundary conditions at the coupled interface are shown in Equations 3.54

and 3.55.

$$T_f = T_s \tag{3.54}$$

$$\boldsymbol{\sigma}_f \cdot \hat{\mathbf{n}}_f + \boldsymbol{\sigma}_s \cdot \hat{\mathbf{n}}_s = 0 \tag{3.55}$$

where the f and s subscripts indicate flow and solid quantities, and the flow field pressure and viscous stresses are combined into a single stress tensor as shown in Equation 3.56.

$$\boldsymbol{\sigma}_f = \sigma_{f_{ij}} = \tau_{ij} - p\delta_{ij} \tag{3.56}$$

These equations state that the flow and solid temperatures must match at the interface, and that the flow and solid surface forces must also match. The flow is computed in an Eulerian reference frame, so the flow stress tensor represents Cauchy stresses, or stresses that are expressed in the deformed configuration in terms of the deformed geometry [38]. The elastic structural response is computed in a total Lagrangian reference frame, however, so the flow stresses must be converted to the 2$^{\text{nd}}$ Piola-Kirchhoff stress tensor before they can be applied as a boundary condition. This is achieved using Equation 2.22 and results in the specified traction boundary condition for the elastic solver shown in Equation 3.57.

$$\mathbf{t}_p = \mathbf{F}\left[\det(\mathbf{F})\mathbf{F}^{-1}\boldsymbol{\sigma}_f(\mathbf{F}^{-1})^T\right] \cdot \hat{\mathbf{n}}_s \tag{3.57}$$

The meshes in the flow and solid domains can be generated independently, and the surface nodes in each domain are not required to be coincident. Therefore, to pass boundary values between meshes it is necessary to interpolate. Interpolation is performed by linking boundary nodes in the solid mesh with boundary faces in the flow mesh, and flow face centroids in the flow mesh with boundary element edges in the solid mesh. Values are then interpolated to the solid nodes and flow face centroids

by assuming a linear variation across the linked flow face or solid edge. The reason for interpolating to nodes in the solid mesh and face centroids in the flow mesh is that those are the most convenient locations at which to apply the boundary conditions for the different discretization techniques used in the flow and solid domains. Figure 3.5 shows an example of the linking process. This figure indicates the flow face centroids as squares, the flow field boundary nodes as triangles, and the solid boundary nodes as circles. As an example, to interpolate a value from the solid mesh to the flow centroid, f_1, Equation 3.58 is used, where θ is the interpolated value, s is the length of the solid boundary edge formed by nodes j and $j+1$, and d is the distance between f_1 and node j.

$$\theta_{f1} = \left(1.0 - \frac{d}{s}\right)\theta_j + \frac{d}{s}\theta_{j+1} \tag{3.58}$$

Likewise, to interpolate from the flow mesh to the solid node $j+2$, Equation 3.59 would be used where s is now the distance between flow mesh nodes $i+1$ and $i+2$, and d is the distance between solid node $j+2$ and flow node $i+1$.

$$\theta_{j+2} = \left(1.0 - \frac{d}{s}\right)\theta_{i+1} + \frac{d}{s}\theta_{i+2} \tag{3.59}$$

LeMANS is a cell-centered finite volume code, but the nodal values in the flow mesh are computed as part of the viscous flux calculations, and therefore it is straightforward to also use the nodal values as part of the boundary interpolation scheme.

Since the elastic response of a structure is being modeled, the interface between the flow and solid domains will often move over the course of a simulation. This can be caused by ablation induced surface recession, or by structural deformation due to aerodynamic or thermal loads. As the fluid-solid interface moves, it becomes necessary to adjust the flow mesh so that the computational domains of the flow and solid remain connected.

The mesh motion approach adopted in LeMANS uses Bézier curves [41] to adjust
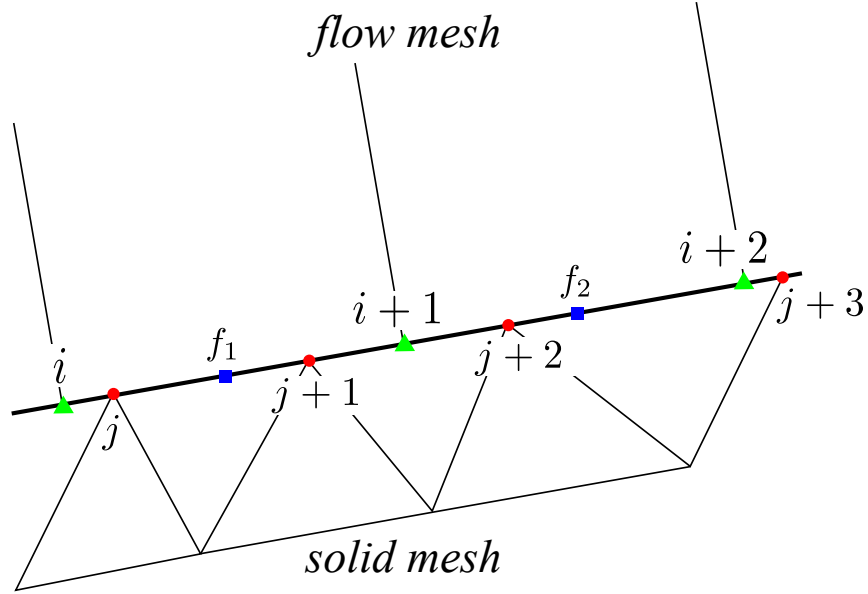
Figure 3.5: Interpolation points along the fluid-solid interface.

the location of nodes within the flow mesh, and the elastic material response solution to determine the interface location. The first step is to identify lines of nodes in the mesh that extend from the coupled interface into the flow field, typically to an inlet boundary. Using the start and end points of the node lines, a third order Bézier curve is defined that is normal to the interface boundary. The nodes that are identified on the mesh lines are then redistributed along the Bézier curves using the one-sided stretching function proposed by Vinokur [103] with a user-defined first cell length. As the elastic material response is updated, the new interface location is used to define new starting points for the Bézier curves, and the nodes are again distributed along the updated curves.

This mesh motion method is fairly simple to implement, and it maintains good mesh quality throughout a simulation since the desired first cell length and mesh clustering are maintained as the interface moves. The use of Bézier curves leads to a smoothly varying distribution of nodes and mesh lines that do not overlap one another. An additional benefit is that the mesh lines defined by the Bézier curves

are normal to the coupled interface, which is beneficial for the prediction of surface properties.

Since only steady-state solutions are sought in LeMANS, the mesh motion does not need to be accounted for in the governing equations or their discretization. The mesh is moved as the coupled interface location changes, and the flow solution is continued in the updated domain.

The number of iterations, N_{it}, between calls to the material response code can be determined in one of two ways. First, it can be a constant user-specified value. This approach is simple to implement and can produce good results [64, 65]. The second approach is to dynamically determine N_{it} based on the convergence of the parameters in LeMANS at the coupled interface that are passed as boundary conditions to the material response. In particular, it is useful to monitor the heat flux predicted by LeMANS at the coupled interface. When the average change in these values between flow iterations drops below a user-specified threshold, then the material response is called again. Additionally, if the average change in these values drops below a separate threshold value between calls to the material response code, then the material response code is not called anymore. This takes advantage of the fact that if the boundary conditions for the material response code are not changing from one call of the code to the next, then the material response solution is not changing, and there is no need to update the boundary conditions for the steady-state flow solution. This approach to setting N_{it} has the benefit of being more physically grounded than the first approach, and in general leads to faster and more robust convergence of the overall coupled problem [40].

The thermal and elastic modules are coupled via the temperature solution. The thermal solver is run first, with a time step of Δt_f and a temperature distribution is obtained. For a dynamic elastic problem the thermal solver is paused after a time step, and the elastic solver is run with a time step of Δt_s up to the time $t^{n+1} = t^n + \Delta t_f$,

where it is assumed that $\Delta t_s < \Delta t_f$. The nodal temperatures from the thermal solver are interpolated linearly in time across the structural time steps in order to account for the thermal load on the structure. This approach is known as multicycling [70]. The updated displacement solution is then used to generate updated control volumes for the thermal code, and thermal code is run for the next time step, Δt_f. Since thermo-mechanical coupling is neglected, the updated displacement field is considered fixed while the thermal code is running. The cycling of thermal and elastic time-stepping is continued until a solution is obtained for the length of the current trajectory point.

In the case of a static elastic problem, the thermal code is run with a time step of Δt_f for the full length of the current trajectory point. The elastic code is then called using the final temperatures computed by the thermal code as the thermal load on the structure. At the end of the material response solution, the final temperature, displacement, and mass blowing rate values are returned to LeMANS as updated flow boundary conditions. There is no need to interpolate the LeMANS boundary conditions in time since a steady state flow solution is sought.

The coupling methodology that is presented in this chapter is designed to work for a certain class of problems. Specifically, quasi-static problems are the focus of this method. The term "quasi-static" refers to the idea, discussed earlier in the chapter, that some of the physical processes in a coupled problem may have much shorter time scales than other processes, and so the "fast" portions of the problem can be treated as steady. In Chapter V, two different test cases are considered, and it is found that the coupling framework is robust, as long as the quasi-static assumption remains valid.

CHAPTER IV

Code Verification Studies

4.1 Introduction

As a code is developed, it is important to ensure that the discretization of the governing equations and the numerical approaches used to solve the equations are implemented correctly. The term that is given to this testing process is code verification. Verification is not concerned with how well a code's results match experimental data, or with whether or not the correct physical models are used for a particular problem; these concerns are related to the task of validation.

Verification and validation are terms that are sometimes used interchangeably when talking about code development, however, they refer to two very different, but equally important ideas. Validation is the process of ensuring that the models used in a code accurately reflect the physical processes that are being studied. Oberkampf [74] states the idea of validation as:

> "Verification does not address whether the conceptual model has any relationship to the real world. Validation, on the other hand, provides evidence (substantiation) for how accurately the computational model simulates reality."

Validation requires the comparison of computational results with experimental data.

53

For cases involving complex physics, including hypersonic or aerothermoelastic problems, the availability of experimental data is typically very limited, thus validation using experimental data is difficult or impossible.

Verification, however, is only concerned with whether or not the models used in a code have been correctly implemented. Roache [81] states this more formally as:

> "The code author defines precisely what partial differential equations are being solved and convincingly demonstrates that they are solved correctly, i.e., usually with some order of accuracy, and always consistently, so that as some measure of discretization Δ (e.g., the mesh increments) $\rightarrow 0$ the code produces a solution to the continuum equations; this is verification. Whether or not those equations and that solution bear any relation to a physical problem of interest to the code user is the subject of validation."

Therefore, verification is a purely mathematical exercise and does not require experimental data or even physically valid solutions to the governing equations. It only seeks to confirm the expected order of accuracy of a code's spatial and temporal discretization approaches.

The above definition of verification refers specifically to the idea of code verification. There is another verification procedure known as solution verification, which seeks to estimate the magnitude of the discretization error for a particular calculation rather than the order of the error [53]. Code verification only needs to be performed once for a given code, but solution verification should be applied to each new calculation that is performed [81]. This chapter is only concerned with code verification, and therefore the term "verification" is taken to refer to the code verification process and not solution verification.

There are several ways in which verification can be accomplished including: comparison with exact analytical solutions (Method of Exact Solutions), comparison with

highly refined solutions from numerical benchmarks, code-to-code comparisons, and the Method of Manufactured Solutions (MMS) [53]. Of these methods, the Method of Manufactured Solutions and the Method of Exact Solutions are the only two that can be considered rigorous order of accuracy verification procedures [84]. The use of manufactured solutions offers some advantages over the use of exact solutions in that it can be used in cases where analytical solutions are impossible to obtain or are very complicated to implement due to the use of infinite series, non-trivial integrals, or special functions [53]. Therefore, the Method of Manufactured Solutions allows for testing of the full, possibly nonlinear, set of governing equations in arbitrary domains with non-constant material properties and non-trivial boundary conditions.

In this chapter, verification results obtained using the Method of Manufactured Solutions are presented for LeMANS, and for both the thermal and structural components of the material response code for a number of different boundary conditions. In the case of LeMANS, the code has already been validated against experimental data and compared with other codes [87], but rigorous verification studies have not been performed for all aspects of the code relevant to the aerothermoelastic test cases considered in Chapter V. The boundary conditions and code options tested for the material response code are likewise targeted towards verifying the portions of the code that are relevant to the aerothermoelastic framework. Therefore, the verification results shown do not represent an exhaustive verification of every code option and boundary condition combination that is possible, but they demonstrate the basic functionality related to the aerothermoelastic coupling and provide confidence in each code's implementation. Tests of the inter-code interpolation process are also presented to ensure that the interpolation does not negatively impact code accuracy.

4.2 Order of Accuracy and Manufactured Solutions

In this section, an overview of the Method of Manufactured Solutions and how this method is used as a part of the code verification process is given. Specific details, including the manufactured solutions that are used, are given for LeMANS and the material response code. Next, a description is given of how the order of accuracy of a code is computed, and what it means for a code to be p^{th}-order accurate. In addition, the error metrics that are used to compute the code order of accuracy are outlined.

4.2.1 The Method of Manufactured Solutions

The Method of Manufactured Solutions is a powerful tool for performing order of accuracy verification of a numerical code. The idea of using manufactured solutions along with grid refinement to determine the order of accuracy is generally credited to Steinberg and Roache [90] and Roache [81]. Many details and practical examples of the MMS are given by Knupp and Salari [53], and a general overview is given by Roache [80]. This method has been used in several different verification exercises including verification of a material thermal response code [6], Euler and Navier-Stokes codes [101, 19, 20, 21, 82], and even a monolithic fluid-structure interaction code [37].

The basic idea of the Method of Manufactured Solutions is to choose, or "manufacture," a solution for each dependent variable in a set of governing equations, and then compute analytical source terms to balance the governing equations. The source terms are necessary since the manufactured solutions do not in general satisfy the governing equations. Source terms are found by substituting the manufactured solutions and any constitutive relations into the governing equations and evaluating them directly. Depending on the complexity of the governing equations and manufactured solutions, the resulting source terms can be very complicated and difficult to work with. To alleviate the risk of introducing errors while adding the source term functions into a code, Mathematica [108] is used to analytically compute the source

terms, and then generate C and Fortran functions which can be directly implemented in LeMANS and the material response code.

A manufactured solution can, in theory, be almost any function of the spatial coordinates and time. In practice, however, there are several guidelines given by Knupp and Salari [53] to ensure that the chosen manufactured solution provides a suitable test of a code. A manufactured solution should consist of smooth, analytic functions such as polynomials or trigonometric functions that are easy to evaluate and allow the theoretical order of accuracy to be obtained. In addition, solutions should contain enough non-trivial derivatives to fully test the code, and be general enough so that all terms of interest in the governing equations are exercised. Finally, the solution should be chosen so that the code will run reliably, and so that assumptions made by the code (e.g. positive temperatures) are not violated. This does not mean that the chosen solution has to be physically valid, i.e. a valid solution to the set of governing equations, but it should not prevent the code from running. The testing of code robustness is not part of the verification process, and cannot be accomplished using manufactured solutions.

One of the key considerations when developing manufactured solutions is the boundary conditions that will be tested. Typically, the standard Dirichlet and Neumann conditions are simple to implement, but more specialized boundary conditions, such as wall boundaries in a CFD code, require extra care. For this reason it is often useful to test the interior equation set first by using Dirichlet or other simple boundary conditions on all boundaries so that errors caused by specialized boundary conditions can be more easily identified and isolated [19].

There are two ways in which boundary conditions can be handled using the Method of Manufactured Solutions. The first method requires the development of a manufactured solution that exactly satisfies a given boundary condition [24]. For instance, when testing a no-slip boundary condition in a Navier-Stokes code, the

57

manufactured solution that is used must have zero velocity at the no-slip boundary. This method is straightforward and requires no modification of the MMS procedure outlined above, however, it requires a different manufactured solution each time a new boundary condition type is tested. The second method of implementing boundary conditions defines an additional source term that is only computed at the boundary, and that depends on the boundary condition type. The additional source term is used to drive the boundary residual to zero [42]. This method is more flexible than the first method in that it allows any variety of boundary conditions to be tested using the same manufactured solution simply by modifying the boundary source term. The drawback is that it requires an additional source term to be computed and enforced at the boundaries. In this work, the first method of implementing boundary conditions is adopted, so different manufactured solutions are generated for each type of boundary condition that is tested beyond the simple Dirichlet and Neumann type boundary conditions.

A common approach used in CFD to obtain a stable higher-order solution for flows that contain shocks is to use a limiter [96]. A limiter locally decreases the order of accuracy of a code in the vicinity of strong flow gradients, such as near a shock, but allows for a higher-order of accuracy to be obtained in smooth regions of the flow. In the case of LeMANS, a minmod limiter is used within the second-order accurate spatial discretization.

A disadvantage to using the Method of Manufactured Solutions is that it requires smooth solutions, which means that the use of the MMS only tests the limiter for the case where no shocks are present in the solution, and higher-order accuracy is expected. The use of the MMS on non-smooth solutions is still an open area of research [83]. Therefore, the performance of LeMANS on flows where shocks are present is not tested in this work.

4.2.1.1 Manufactured Solutions: LeMANS

The governing equations for LeMANS are given by Equations 2.1 - 2.4. In the MMS tests, however, vibrational energy is not considered, so Equation 2.4 is not needed. A two species flow consisting of atomic and molecular nitrogen in chemical equilibrium is used for most of the presented cases, so there are two species conservation equations, and the chemical source terms are zero for both equations. It is therefore necessary to construct manufactured solutions for 5 different variables that are independent of one another. The chosen set of variables for which solutions are manufactured is: $\rho_{N_2}, \rho_N, u, v, T_{tr}$. Note that this is not the only valid set of variables that could be chosen, but it represents a convenient set for use in LeMANS. In addition to the governing equations, the necessary auxiliary relations for the thermodynamic and transport properties can be found in Scalabrin [87]. These relations also make use of the manufactured variables for computing the MMS source terms.

Taking into account the guidelines listed in the previous section, the manufactured solutions for LeMANS are shown in Equation 4.1. This set of equations defines a two-dimensional steady flow-field for a free stream test case designed to use the supersonic inlet and supersonic outlet boundary conditions that are available in LeMANS. These are the two simplest boundary conditions that can be applied in LeMANS, and therefore are useful for testing the interior equations before including more complicated boundary conditions. The manufactured solution contains a number of constants that are defined by the code user, and can be used to make the solution more or less

complicated in order to test specific aspects of the code.

$$\rho_s(x, y) = \rho_{s_0} + \rho_{x_s}\cos\left(\frac{a_{x_s}\pi x}{L}\right) + \rho_{y_s}\sin\left(\frac{a_{y_s}\pi y}{L}\right) + \rho_{xy_s}\cos\left(\frac{a_{xy_s}\pi xy}{L^2}\right)$$

$$u(x, y) = u_0 + u_x\sin\left(\frac{a_{x_u}\pi x}{L}\right) + u_y\cos\left(\frac{a_{y_u}\pi y}{L}\right) + u_{xy}\cos\left(\frac{a_{xy_u}\pi xy}{L^2}\right)$$

$$v(x, y) = v_0 + v_x\sin\left(\frac{a_{x_v}\pi x}{L}\right) + v_y\cos\left(\frac{a_{y_v}\pi y}{L}\right) + u_{xy}\cos\left(\frac{a_{xy_u}\pi xy}{L^2}\right) \quad (4.1)$$

$$T_{tr}(x, y) = T_0 + T_x\cos\left(\frac{a_{x_T}\pi x}{L}\right) + T_y\cos\left(\frac{a_{y_T}\pi y}{L}\right) + T_{xy}\cos\left(\frac{a_{xy_T}\pi xy}{L^2}\right)$$

In addition to the free stream boundary conditions, the inviscid isothermal wall boundary condition and the no-slip, isothermal wall boundary condition are tested in LeMANS. For these boundary conditions the manufactured solution must be modified so that it satisfies the constraints implied by the different wall boundaries. At the inviscid isothermal wall, the normal flow velocity must be zero, and the temperature must be constant. Equation 4.2 shows the modified manufactured solution for the inviscid wall case. The $f(x, y) - C_s$ term defines a surface in the flow domain, and when $f(x, y) = C_s$ the v-velocity will be zero and the temperature will be T_0 to satisfy the wall boundary conditions. For the cases that are considered, $f(x, y) = y$ and $C_s = 0.0$ which enforces zero v-velocity and constant temperature at a surface located at $y = 0.0$. The wall boundary for the domains tested in Section 4.3 is flat, and lies along $y = 0.0$, so it is not necessary to modify the u-velocity from Equation 4.1 in order to set the normal velocity to zero. Similarly, there is no need to change the density solutions from Equation 4.1 to satisfy the boundary conditions.

$$v(x, y) = (f(x, y) - C_s)\left\{v_0 + v_x\sin\left(\frac{a_{x_v}\pi x}{L}\right) + v_y\cos\left(\frac{a_{y_v}\pi y}{L}\right)\right.$$
$$\left. + u_{xy}\cos\left(\frac{a_{xy_u}\pi xy}{L^2}\right)\right\}$$
$$\quad (4.2)$$
$$T_{tr}(x, y) = T_0 + (f(x, y) - C_s)\left\{T_x\cos\left(\frac{a_{x_T}\pi x}{L}\right) + T_y\cos\left(\frac{a_{y_T}\pi y}{L}\right)\right.$$
$$\left. + T_{xy}\cos\left(\frac{a_{xy_T}\pi xy}{L^2}\right)\right\}$$

The only change that needs to be made to the inviscid isothermal wall boundary manufactured solution for use with the viscous isothermal wall boundary is that the u-velocity must also go to zero at the wall. This is accomplished in the same way as for the v-velocity in Equation 4.2, and the result is shown in Equation 4.3.

$$u(x,y) = (f(x,y) - C_s)\left\{u_0 + u_x\sin\left(\frac{a_{x_u}\pi x}{L}\right) + u_y\cos\left(\frac{a_{y_u}\pi y}{L}\right) \right.$$
$$\left. + u_{xy}\cos\left(\frac{a_{xy_u}\pi xy}{L^2}\right)\right\} \tag{4.3}$$

4.2.1.2 Manufactured Solutions: Material Response

The governing equation for the thermal material response code is Equation 2.11. The only variable that needs to be analytically defined is the temperature, for which the manufactured solution is shown in Equation 4.4. In addition to the temperature, analytical forms of the specific heat and thermal conductivity are constructed as shown in Equation 4.5. The only boundary conditions that are tested with the thermal code are the specified temperature and specified heat flux conditions, which correspond to Dirichlet and Neumann boundary conditions, so there is no need to develop an additional manufactured solution to test the boundary conditions. Note that the manufactured solution in this case is dependent on time as well as the spatial coordinates, so it is possible to test the temporal order of accuracy of the code.

$$T(x,y,t) = T_0\cos\left(B_x x^2 + B_y y^2 + \omega_T t\right) + \epsilon_T \tag{4.4}$$

$$\left.\begin{aligned}\kappa_{ij}(T) &= \kappa_{0_{ij}} + \kappa_{1_{ij}}T + \kappa_{2_{ij}}T^2 \\ c_p(T) &= c_{p_0} + c_{p_1}T\end{aligned}\right\} \tag{4.5}$$

The implementation of manufactured solutions for the structural material response code is very similar to the thermal code. The governing equation in this case is Equation 2.17 and the manufactured variables are the displacements in the x and y directions. The necessary material properties: Young's modulus, shear modulus, Poisson's

ratio, and thermal expansion coefficients are all temperature dependent, and have the same form as the thermal conductivity shown in Equation 4.5. The manufactured solution for displacement is shown in Equation 4.6. Once again, only Dirichlet and Neumann boundary conditions are tested, so there is no need to develop additional manufactured solutions to satisfy specialized boundary conditions. Temperature is also used in the structural code to compute thermal stresses and material property values, but it is not one of the solution variables. Therefore, the temperature distribution shown in Equation 4.7 is specified in the code, and is used in the construction of the manufactured source term.

$$u(x, y, t) = u_0 + u_t\cos\left(\frac{a_{ut}\pi t}{T_0}\right)\left\{u_x\sin\left(\frac{a_{ux}\pi x}{L}\right) + u_y\cos\left(\frac{a_{uy}\pi y}{L}\right)\right.$$
$$\left. + u_{xy}\sin\left(\frac{a_{uxy}\pi xy}{L^2}\right)\right\}$$

$$v(x, y, t) = v_0 + v_t\cos\left(\frac{a_{vt}\pi t}{T_0}\right)\left\{v_x\sin\left(\frac{a_{vx}\pi x}{L}\right) + v_y\cos\left(\frac{a_{vy}\pi y}{L}\right)\right.$$
$$\left. + v_{xy}\sin\left(\frac{a_{vxy}\pi xy}{L^2}\right)\right\}$$

(4.6)

$$T_s(x, y, t) = T_0 + T_t\sin\left(\frac{a_{Tt}\pi t}{T_0}\right)\left\{T_x\cos\left(\frac{a_{Tx}\pi x}{L}\right) + T_y\sin\left(\frac{a_{Ty}\pi y}{L}\right)\right\} \qquad (4.7)$$

4.2.2 Order of Accuracy

The order of accuracy of a code refers to the rate at which the error in the solution decreases as the spatial or temporal discretization is refined. The discretization error is generally assumed to have the form shown in Equation 4.8, neglecting higher order terms. The time discretization is given by Δt and the spatial discretization by Δh, and a and b are constants. The superscripts m and q are the orders of accuracy of the spatial and temporal discretizations.

$$E_D = a\Delta t^q + b\Delta h^m \qquad (4.8)$$

So, if a code is expected to have second-order spatial accuracy, then $m = 2$, and the spatial error in the solution, given by $E_{D_s} = \Delta h^m$, should decrease by a factor of four as the mesh is refined by a factor of two. If, after a series of systematic grid or time step refinements, the error in the solution does not match the expected trend, then there is a strong possibility that either some portion of the code is implemented incorrectly, or is implemented in such a way that the theoretical order of accuracy is decreased.

The order of accuracy, p, is determined from Equation 4.9 where $\hat{\mathrm{E}}(mesh_1)$ and $\hat{\mathrm{E}}(mesh_2)$ are measures of error on two different meshes, and r is the refinement ratio between the meshes. The refinement may be spatial or temporal depending on whether the spatial or temporal order of accuracy is being determined. The theoretical spatial order of accuracy for LeMANS is second-order. The temporal order of accuracy for LeMANS is not considered since only steady state solutions are sought. The expected spatial order of accuracy for both the thermal and structural modules of the material response code is second-order. The thermal material response code is expected to be first-order accurate in time, and the structural material response code should be second-order accurate in time. Results are not presented for the time accuracy of the structural code, however, as only static solutions are considered in the Chapter V test cases.

$$p = \frac{\log\left(\frac{\hat{\mathrm{E}}(mesh_1)}{\hat{\mathrm{E}}(mesh_2)}\right)}{\log(r)} \tag{4.9}$$

The spatial and temporal orders of accuracy are often tested independently, however, it is possible to simultaneously refine the grid spacing and the time step in order to verify both orders of accuracy at the same time. The benefit of this approach is that it can lead to substantial computational time savings since it is not necessary to solve a problem using either very small mesh spacing or time steps. The main drawback is that if the observed order of accuracy is different from what is expected,

then it is not necessarily obvious whether an error is present in the spatial or temporal discretization [6]. The theory for how to simultaneously refine the grid spacing and time step is outlined below, and is based on the work shown in Amar [6].

By factoring out the quantity $b\Delta h^m$, Equation 4.8 can be rewritten as shown in Equation 4.10, where the th subscripts represent the theoretical orders of accuracy, which are known for a given discretization technique.

$$E_D = \left(1 + \frac{a\Delta t^{q_{th}}}{b\Delta h^{m_{th}}}\right) b\Delta h^{m_{th}} \tag{4.10}$$

The grid spacing and time step can be refined simultaneously as long as the ratio of $\frac{\Delta t^{q_{th}}}{\Delta h^{m_{th}}}$ is kept constant. If the ratio is constant, then taking the log of both sides of Equation 4.10, and taking the derivative of $\log(E_D)$ with respect to $\log(\Delta h)$, leads to Equation 4.11, which is the continuous form of Equation 4.9. Therefore, if p-order accuracy is observed over the course of a grid refinement study, then it is valid to say that both the q_{th}-order of accuracy of the temporal discretization and the m_{th}-order of accuracy of the spatial discretization are verified.

$$\frac{\partial \left[\log\left(E_D\right)\right]}{\partial \left[\log\left(\Delta h\right)\right]} = p \tag{4.11}$$

If $\log(E_D)$ is plotted as a function of $\log(\Delta h)$, then the slope of the resulting line is the observed order of accuracy. The observed order of accuracy is often compared to the theoretical by plotting the observed error norms vs. mesh size on a log-log plot, and then plotting an additional line on the same plot that represents the theoretical order of accuracy. If the two lines have similar slopes, then the expected order of accuracy is confirmed. An example of this comparison is shown in Figure 4.1a for a code that is expected to be second-order accurate. As can be seen, the error computed from a coarse mesh does not necessarily lead to the expected order of accuracy, but as the mesh is refined the observed order of accuracy approaches the theoretical. This is

(a) Error plot   (b) Order of accuracy plot
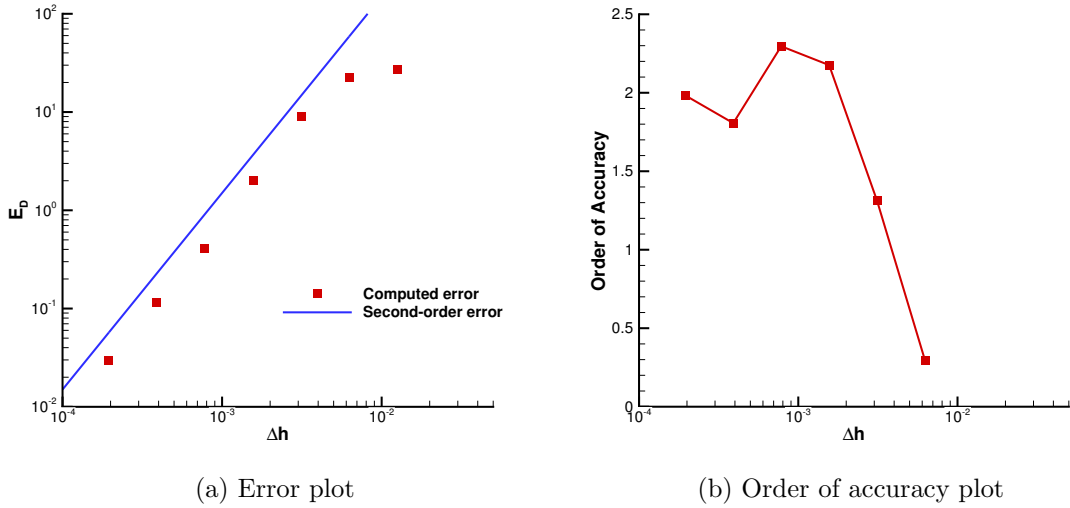
Figure 4.1: Example order of accuracy plots for a theoretically second-order accurate code.

known as the asymptotic error behavior, and it is important that the grid is refined sufficiently to reach the asymptotic error regime.

A different method of displaying the order of accuracy is the approach of Bond et al. [18], where the computed order of accuracy is directly plotted against the log of the grid spacing. Figure 4.1b shows an example of this type of plot using the same data as Figure 4.1a. Plotting the order of accuracy directly makes it easier to quickly identify when the order of a code is less than expected, and therefore the results presented in this chapter use this style of plot.

As part of the verification process it is necessary to systematically refine the time step and grid spacing. While the definition of a time step is relatively unambiguous, the choice of a grid spacing metric is somewhat harder to define, especially for unstructured meshes. In this work, the number of elements in one dimension, computed as $\sqrt[d]{N_e}$, is used to define the characteristic mesh length scale, Δh, as shown in Equation 4.12. In this equation d is the number of dimensions of the problem, N_e is the number of elements in the mesh, and L is the length of the domain. This definition

works well for the problems that are considered in the present work since all of the domains are initially square.

$$\Delta h = \frac{L}{\sqrt[d]{N_e}} \tag{4.12}$$

The results presented in Section 4.3 present the order of accuracy as a function of the normalized grid spacing, h_{norm}, which is defined as $h_{norm} = \frac{\Delta h}{\Delta h_n}$, where Δh_n is the grid spacing for the most refined mesh. As h_{norm} approaches one, the grid spacing becomes smaller.

The error values used in Equation 4.9 are based on error norms of the solutions at each mesh refinement. Two different error norms, the L_∞ norm and the L_2 norm, are used to compute the observed order of accuracy. Definitions of each norm are shown in Equation 4.13 where $\mathbf{e}$ is the vector of nodal errors.

$$\|\mathbf{e}\|_\infty = \max\left(|e_1|, ..., |e_n|\right)$$
$$\|\mathbf{e}\|_2 = \sqrt{\frac{1}{N_e} \sum_{i=1}^{n} (e_i)^2} \tag{4.13}$$

The reason for computing the order of accuracy using different norms is that each norm highlights different aspects of the error. The L_2 norm provides a more "averaged" view of the error, which means that it gives a good global picture of the order of accuracy, but may not detect localized regions of lower order accuracy. If first-order accurate errors are being generated at one specific boundary or at a small number of points within the domain, then the L_2 norm may still indicate second-order accuracy even though that is not the order of accuracy present throughout the entire domain. The L_∞ norm is much more sensitive, however, and will detect even highly localized regions of first-order accurate errors [18]. Looking at the order of accuracy using both of these norms helps to provide a more complete picture of how the discretization error is behaving across a series of mesh refinements.

4.3 Verification Test Cases

In this section, the results of applying the Method of Manufactured Solutions to LeMANS and the material response code are presented. In cases where the observed order of accuracy does not match the theoretical order, details on why this occurred are presented, and, when possible, changes made to the codes which allow the theoretical order to be obtained are described as well. In addition to the verification of the individual flow and material response codes, verification studies for some portions of the coupled aerothermal code are presented, and the verification of the full aerothermoelastic coupling procedure is discussed.

4.3.1 LeMANS: Free Stream Boundary Conditions

The first MMS test case in LeMANS uses the supersonic inlet and supersonic outlet boundary conditions in order to model a free stream flow. Testing these boundary conditions first allows for errors in the interior equations to be identified more easily before using more complicated boundary conditions. Due to the hyperbolic nature of the inviscid portion of the governing flow equations, at a supersonic inlet information can only propagate into the flow domain, and at a supersonic outlet information can only propagate out of the domain [46]. This means that at an inlet, all of the flow variables must be specified from the manufactured solutions. At an outlet, the boundary values are set by extrapolating the solution from the interior cells. The order of extrapolation used to set the boundary values can have an impact on the observed order of accuracy of the code, and can influence the level of error in the solution. This effect is demonstrated by using two different extrapolation orders.

Figure 4.2 shows the manufactured variables for the free stream case. Flow is from the lower left corner of the domain to the upper right corner with supersonic inlet boundary conditions set at the $x = 0.0$ and $y = 0.0$ boundaries, and supersonic outlet boundary conditions used at the $x = 1.0$ and $y = 1.0$ boundaries. The flow is

supersonic everywhere, so it is valid to use the supersonic inlet and outlet boundary conditions. The series of meshes listed in Table 4.1 is used for the grid refinement study. Not all of the mesh levels listed in Table 4.1 are used for every case, but only as many as are needed in order to demonstrate the asymptotic order of accuracy. This series of meshes is used for all of the LeMANS MMS test cases, not just the free stream case. Uniform meshes are used for all LeMANS verification test cases. The

Refinement Level	Number of Elements	Refinement Ratio (r)	h_{norm}
0	64	–	32
1	256	2	16
2	1024	2	8
3	4096	2	4
4	16384	2	2
5	65536	2	1

Table 4.1: Meshes used for the grid refinement studies of LeMANS.

constants used in the manufactured solutions for the free stream test case are listed in Table C.1 of Appendix C.
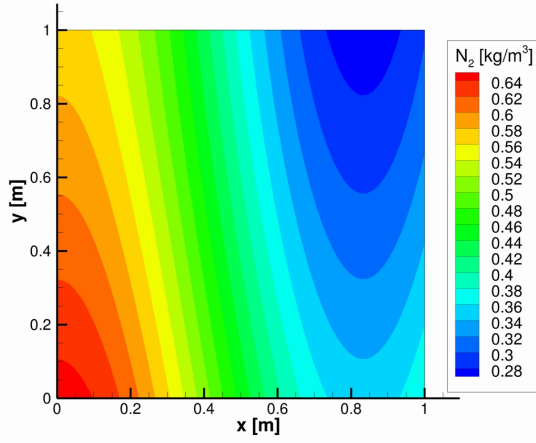
(a) Density of N_2
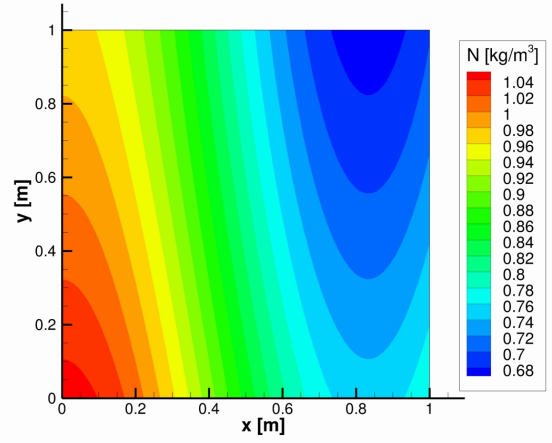
(b) Density of N
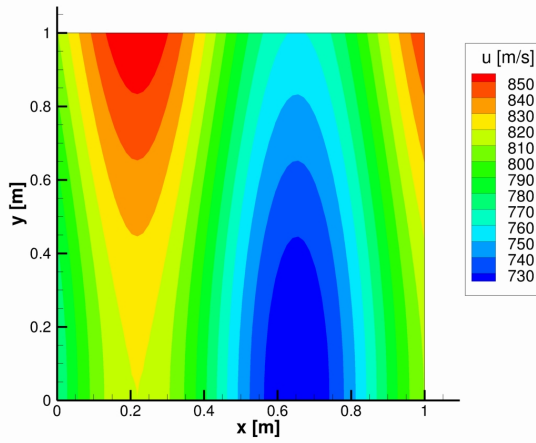
(c) u-velocity
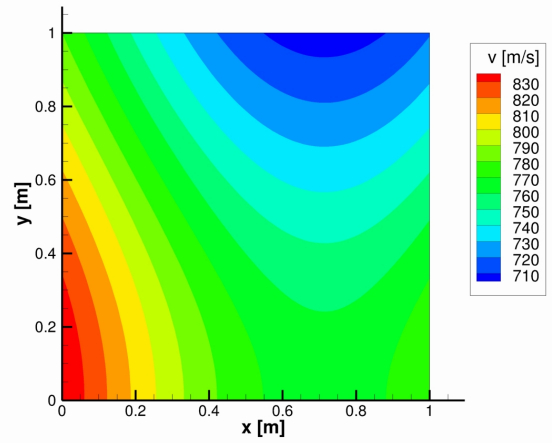
(d) v-velocity
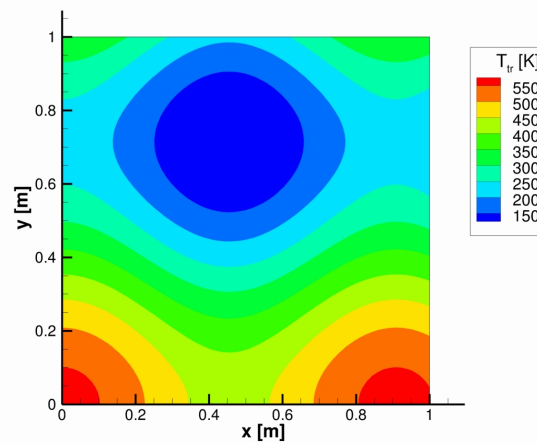
(e) Temperature

Figure 4.2: Contours of manufactured quantities for the free stream test case.

Results for two variations of the free stream test case are presented in this section. The first case uses only the Euler equations to model a single species flow. The second case uses the full Navier-Stokes equations and models a two species flow in chemical equilibrium. The manufactured solution constants are the same in both cases, with the exception that the Euler case has only one density equation.

Figure 4.3 shows the observed order of accuracy results for the single species Euler equation test case using zero-order extrapolation to set the supersonic outlet boundary values. Zero-order extrapolation sets the ghost cell values equal to the interior cell values at the boundary, and therefore enforces a zero-gradient condition at the outlet. It is clear that the observed order of accuracy in both norms is close to first-order rather than the theoretically expected second-order accuracy.
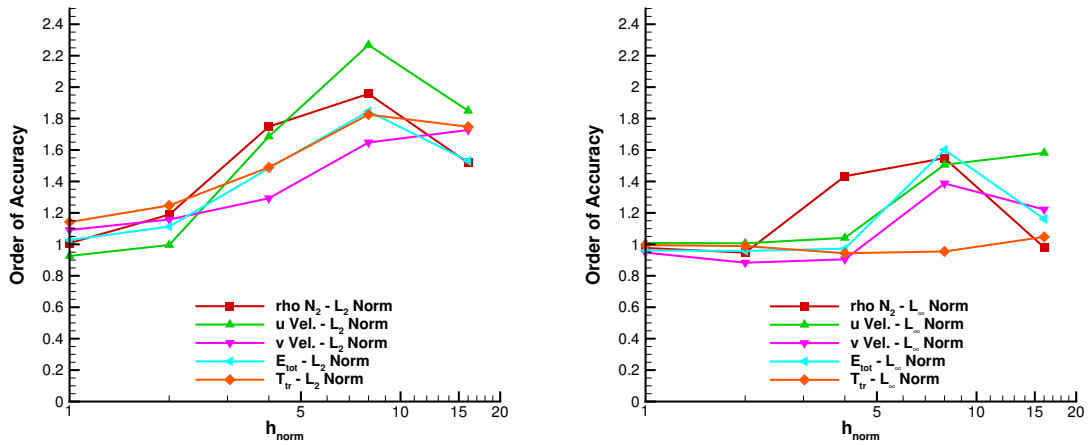
(a) Order of accuracy based on L_2 norms

(b) Order of accuracy based on L_∞ norms

Figure 4.3: Initial observed order of accuracy for the free stream test case of the Euler equations.

Looking at the error contours for this case, which are shown in Figure 4.4, there are clearly larger errors clustered near the outlet boundaries. This is due to two different implementation choices in LeMANS that produce correct results, but cause the free stream boundary conditions to be first-order accurate. The first implementation

issue is the number of ghost cells that are generated around the domain. The results shown in Figure 4.3 use only a single layer of ghost cells, but the boundary flux is still computed using MUSCL variable extrapolation, which requires two values on either side of a face to construct a second-order flux. Since there is only a single ghost cell on the outside of each boundary face, the MUSCL extrapolation is effectively truncated, and the reconstructed boundary flux is no longer fully second order accurate. This is an issue for all of the boundary conditions, not just the supersonic inlet and outlet conditions.

The second implementation issue is the use of zero-order extrapolation at the outlet boundaries. This is appropriate when the flow field properties actually have zero gradients at an outlet, however, this is not the case for the manufactured solution. The effect of this extrapolation approach is to compute a boundary flux that decreases in error more slowly than the rest of the flow field since the ghost cell value is not computed with the same level of accuracy as the rest of the flow domain.

(a) Percent error in N_2 density

(b) Percent error in v-velocity
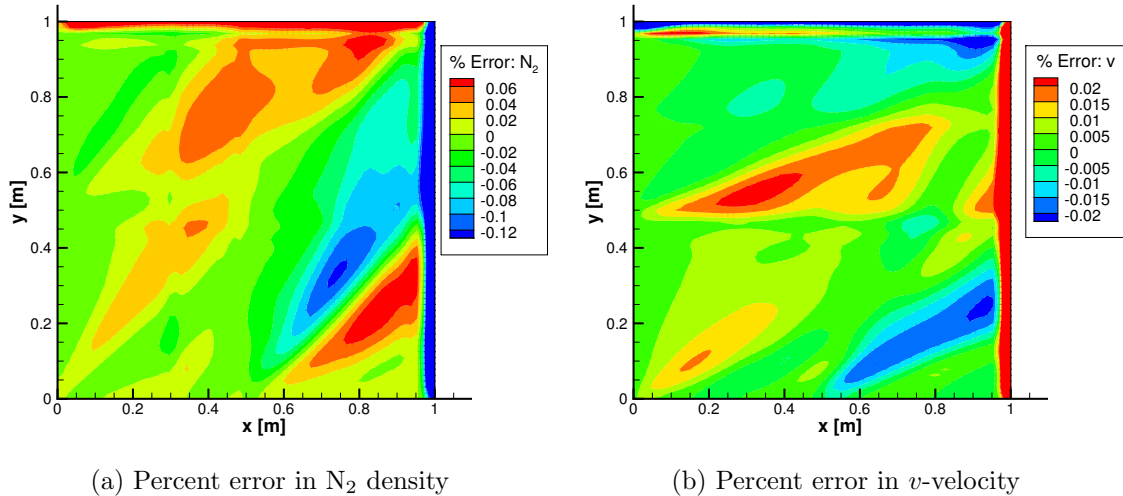
Figure 4.4: Error contours for the Euler free stream test case with one layer of ghost cells and zero-order outlet boundary extrapolation.

Figure 4.5 shows error contours for the same test case, but with a second layer

of ghost cells added around the boundary and with linear, rather than zero-order, extrapolation used to set the ghost cell values at the outlet boundaries. After these two changes, the large boundary error that appeared in Figure 4.4 has decreased to the level of the rest of the flow field. Figure 4.6 shows the new order of accuracy for this case, and, as expected, both the L_∞ and L_2 norms show second-order accuracy.

(a) Percent error in N_2 density

(b) Percent error in v-velocity
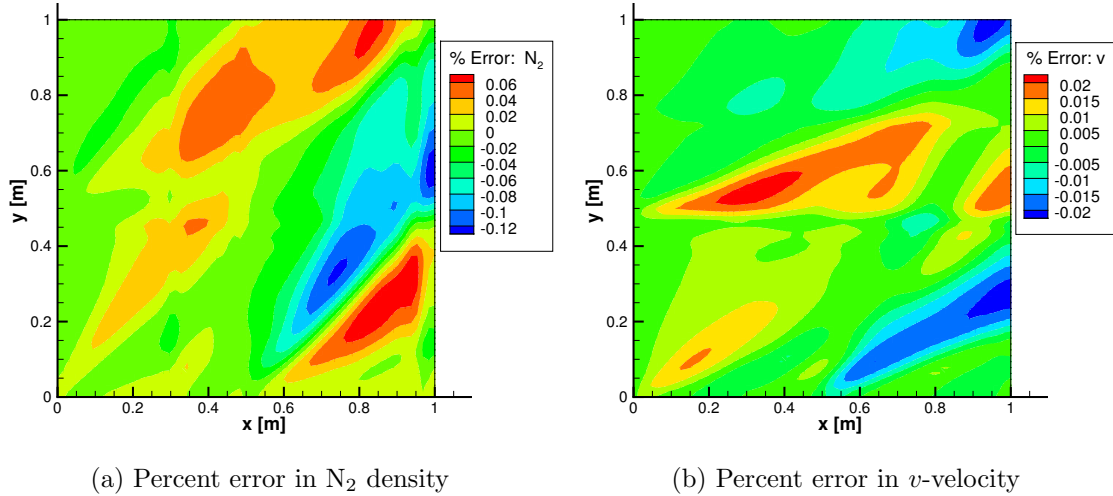
Figure 4.5: Error contours for the Euler free stream test case with two layers of ghost cells and linear extrapolation at the outlet boundaries.

The remaining free stream case is to test the full Navier-Stokes equations with a two species flow in chemical equilibrium. Figure 4.7 shows the order of accuracy results for this case using the updated boundary condition implementation. Second-order accuracy is observed in both the L_2 and L_∞ norms. Based on these results, the inlet boundary conditions, outlet boundary conditions, and interior equations of LeMANS are verified.
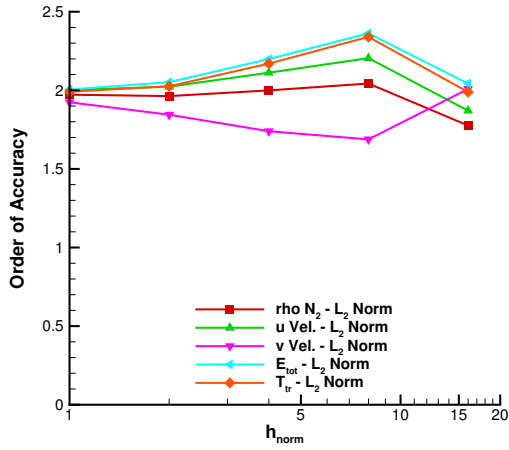
(a) Order of accuracy based on L_2 norms
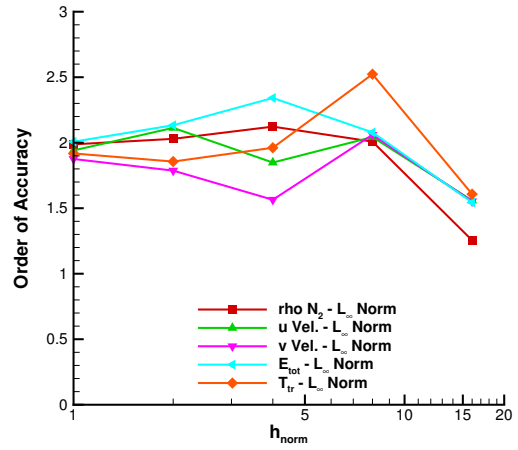
(b) Order of accuracy based on L_∞ norms

Figure 4.6: Observed order of accuracy for the Euler free stream test case with two layers of ghost cells and linear extrapolation at the outlet boundaries.
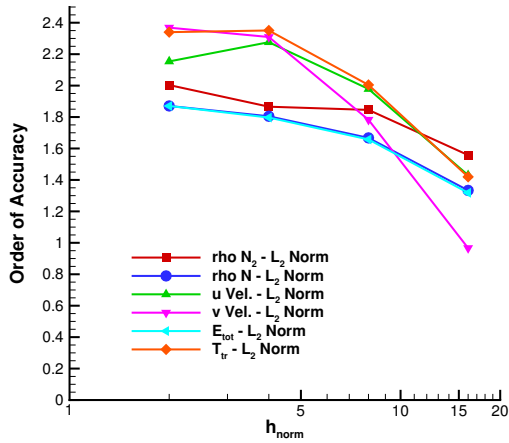
(a) Order of accuracy based on L_2 norms
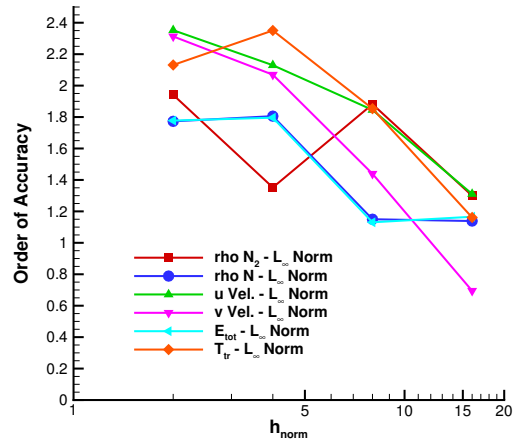
(b) Order of accuracy based on L_∞ norms

Figure 4.7: Observed order of accuracy for the Navier-Stokes free stream test case.

4.3.2 LeMANS: Wall Boundary Conditions

Two different wall boundary conditions are considered: an inviscid, isothermal wall, and a no-slip, isothermal wall. The inviscid wall only tests the implementation

of the Euler equations, while the viscous wall requires the solution of the full Navier-Stokes equations. The reason for looking at two different wall boundary conditions is to isolate any errors that may be caused by the inviscid calculations before working with the full set of viscous equations.

(a) u-velocity

(b) v-velocity
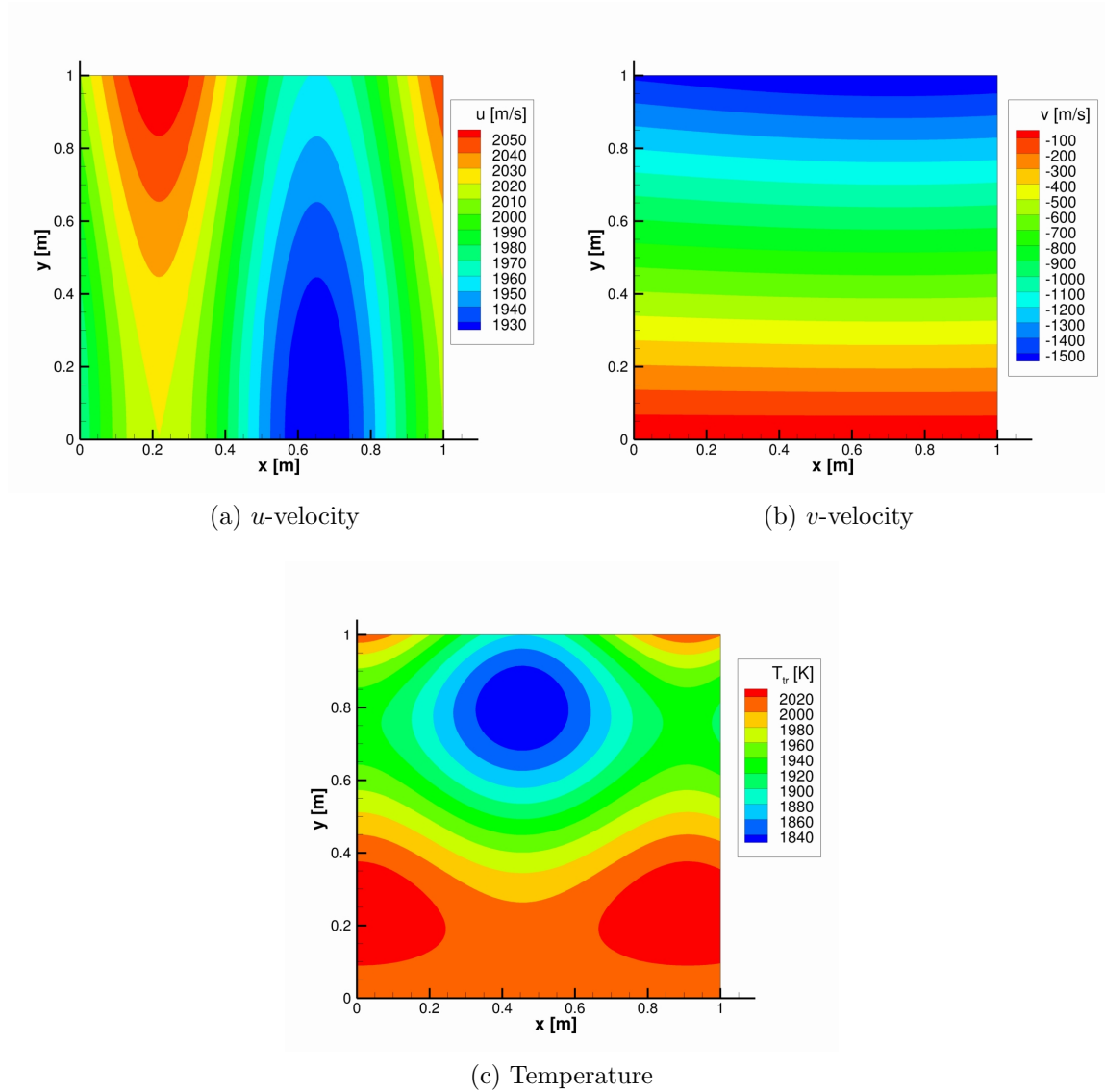
(c) Temperature

Figure 4.8: Contours of manufactured quantities for the inviscid wall test case.

The manufactured velocity and temperature contours for the inviscid wall case are shown in Figure 4.8. Flow is from left to right, and the manufactured solution is designed so that the flow is supersonic at the inlet and outlet boundaries so that

74

the supersonic boundary conditions that were tested in the previous section can be used again. The density contours are the same as for the free stream case, and the u-velocity contours are the same shape, but the velocity magnitude is increased to ensure that the flow remains supersonic at the inlet and outlet boundaries. The wall boundary is located at $y = 0.0$, inlet boundaries are at $x = 0.0$ and $y = 1.0$, and the outlet boundary is at $x = 1.0$. The meshes listed in Table 4.1 are again used for the grid refinement study, and the constants used in the manufactured solution are shown in Table C.2 of Appendix C.

The order of accuracy results for the inviscid wall test case are shown in Figure 4.9. In this case the errors for the velocity components show second-order convergence, but the error for all other variables converges at a rate that is closer to first-order. To understand why this is the case, it is necessary to look at how the inviscid wall boundary conditions are implemented in LeMANS.
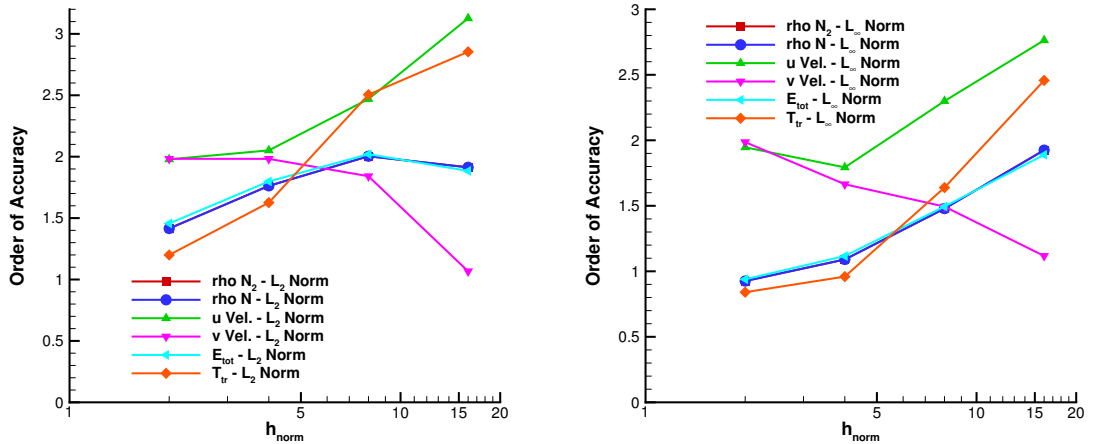
(a) Order of accuracy based on L_2 norms   (b) Order of accuracy based on L_∞ norms

Figure 4.9: Initial observed order of accuracy for the inviscid wall test case.

Equation 4.14 shows how the ghost cell values are populated for each variable, with the g subscript indicating a ghost cell value, the s subscript denoting the species, and the cl subscript denoting the interior cell adjacent to the wall boundary. These

equations are written in the reference frame of the mesh boundary face, so the n and t subscripts indicate components in the directions normal and tangential to the face. Implementing the boundary conditions in this fashion correctly sets the zero normal velocity condition, however, it also forces the other variables to have zero gradient at the wall. This is not the case for the manufactured solution, and is also not necessarily the case for a general non-MMS problem.

$$\rho_{s,g} = \rho_{s,cl}$$
$$\rho_g u_{n,g} = -\rho_{cl} u_{n,cl}$$
$$\rho_g u_{t,g} = \rho_{cl} u_{t,cl} \qquad (4.14)$$
$$E_{tot,g} = E_{tot,cl}$$

To improve the accuracy of the inviscid wall boundary condition, linear extrapolation from the interior domain to the wall is implemented for the density and energy variables. The implementation of the velocity components at the boundaries is unchanged since the normal velocity condition is already being properly enforced. Figure 4.10 shows the results of these changes on the observed order of accuracy. It can be seen that now all variables show the expected second-order accuracy in both norms. The error contours for temperature are shown in Figure 4.11 for the original and updated wall boundary implementations. Even after implementing linear extrapolation, the maximum error is still concentrated near the wall boundary, but the magnitude of the maximum error has decreased significantly by using linear extrapolation. While not shown, the error in the other flow variables shows similar behavior.

At this point, the interior equations, inviscid wall boundary conditions, and inlet and outlet boundary conditions have all been verified. The next case to consider is the viscous wall boundary. Manufactured flow quantities are shown for this case in Figure 4.12. The temperature and density contours are the same as for the inviscid wall case, and the wall boundary is again located at $y = 0.0$. Inlet boundary conditions are now
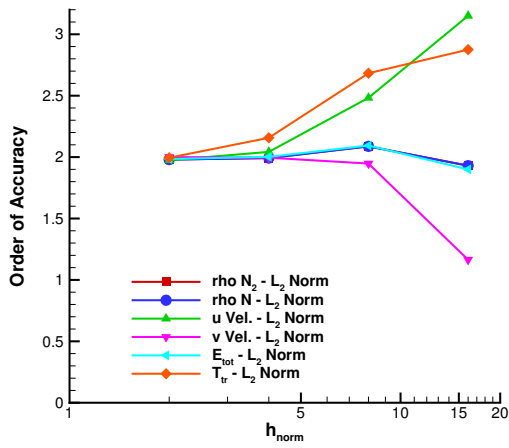
(a) Order based on L_2 norms
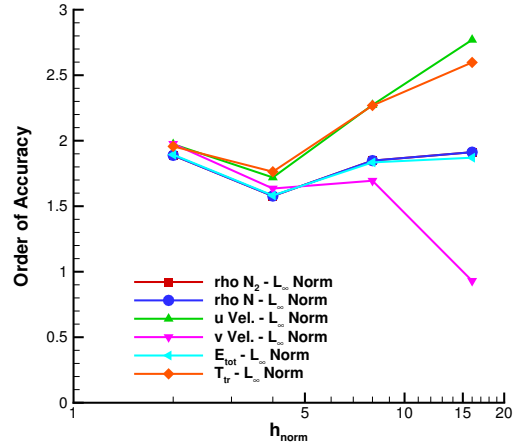
(b) Order based on L_∞ norms

Figure 4.10: Observed order of accuracy for the inviscid wall test case.
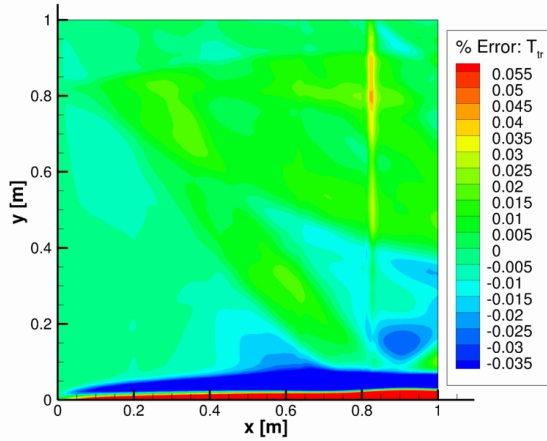
(a) Percent error in temperature, original boundary condition
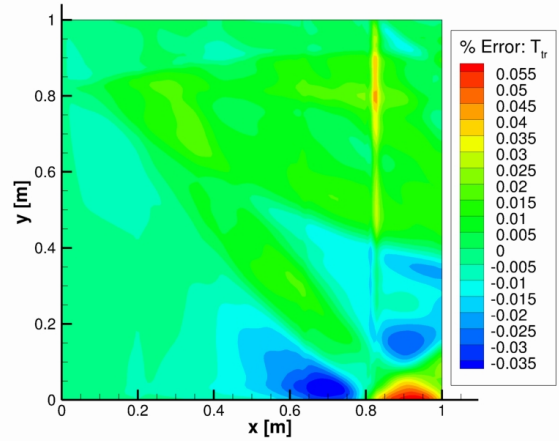
(b) Percent error in temperature, updated boundary condition

Figure 4.11: Error contours for the inviscid wall test case for the two different wall boundary implementations.

applied at all boundaries except the wall, however, since the cells near the wall will contain subsonic flow due to the no-slip condition. The supersonic outlet boundary condition is therefore not valid immediately adjacent to the wall. It is valid to use

an inlet boundary condition on all of the non-wall boundaries because it specifies the full state vector, and can be used within LeMANS as a Dirichlet boundary condition to exactly set the ghost cell values [82]. The constants used in the manufactured solution are found in Appendix C in Table C.3.

(a) u-velocity   (b) v-velocity
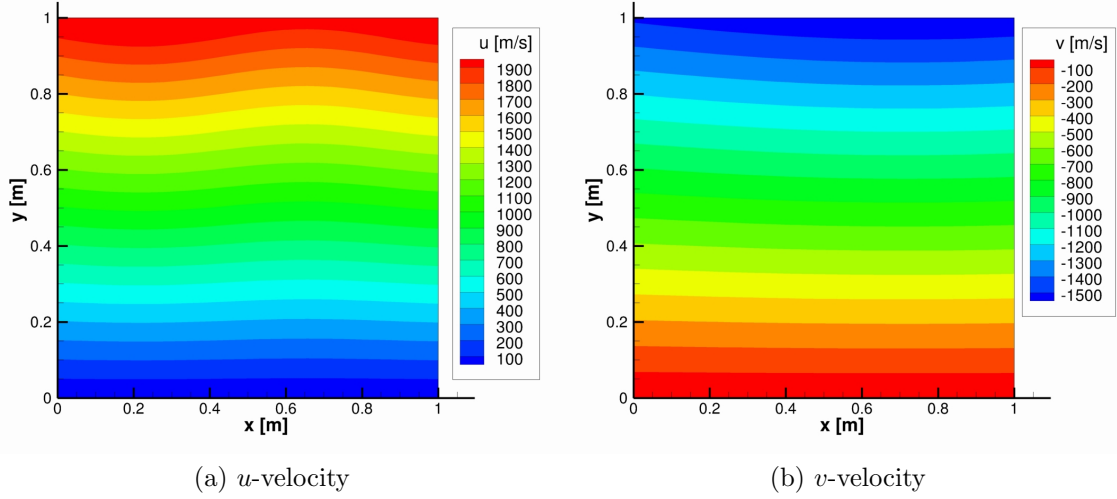
Figure 4.12: Contours of manufactured velocities for the no-slip wall test case.

The order of accuracy results for the viscous case are shown in Figure 4.13. The observed order of accuracy is near two based on the L_2 norms for all variables except the u-velocity, but the L_∞ norms show an order of accuracy near one for all variables. The exact reason for this is not clear, however there are several possibilities.

One possible reason is the way in which the viscous fluxes are computed. In order to compute the derivatives necessary for the viscous flux calculations, LeMANS requires the values of the flow quantities at the mesh nodes. These values are obtained by averaging the cell centered values from all of the cells that share a node. At the domain boundaries the averaging process includes values from the ghost cells, and, especially at the corners of a domain where different boundary conditions meet, this averaging can lead to less accurate values being computed at the boundary nodes. This could result in localized errors that decrease at a slower rate than the errors in

the remainder of the flow field, and which would decrease the order of accuracy based on the L_∞ norms.

Regardless of the reason for the observed order of accuracy, it is important to remember that lower order of accuracy does not mean that the results generated by LeMANS are incorrect. The errors in the solution are still decreasing as the mesh is refined; the rate of decrease is just less than theoretically expected. Therefore, while the implementation of the viscous wall boundary condition cannot be considered rigorously verified based on the results presented, the use of LeMANS within the aerothermoelastic framework is still valid.
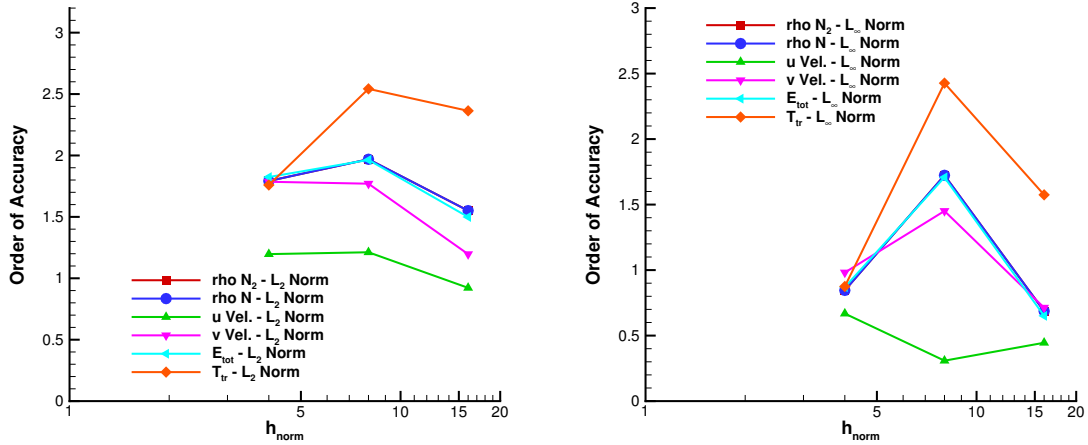
(a) Order of accuracy based on L_2 norms

(b) Order of accuracy based on L_∞ norms

Figure 4.13: Observed order of accuracy for the no-slip wall test case.

Based on the manufactured solution tests, several code options and boundary conditions in LeMANS can be considered formally verified. Specifically, the supersonic inlet and outlet boundary conditions, the inviscid wall boundary condition, and the interior equations for a multi-species gas mixture in thermal and chemical equilibrium have been shown to be correctly implemented within the limits of the MMS. This does not necessarily mean that there are no errors in the code, but, in conjunction with previously performed validation studies and code-to-code comparisons, it provides

increased confidence that the results produced by LeMANS are correct within the limits of the models that are used.

4.3.3 Thermal Material Response

The thermal material response code has two main boundary condition options: specified temperature and specified heat flux. Both of these boundary conditions can be tested along with the interior equations using the same manufactured solution, which is given by Equation 4.4. The boundary conditions for the material response code are applied directly at the boundary nodes, so unlike LeMANS there are no ghost cells. The main difference in applying the MMS to the thermal material response code rather than LeMANS is that the temporal order of accuracy is considered in addition to the spatial order of accuracy. As described in Section 4.2.2, both orders of accuracy are verified simultaneously by refining both the time step and the grid spacing at each refinement level in the grid refinement study. Table 4.2 shows the grid levels and time steps that are used for the specified temperature boundary condition cases.

Refinement Level	Number of Elements	Refinement Ratio (r)	h_{norm}	Time Step [s]
0	72	–	67.68	1.0
1	304	2.05	32.94	2.3684×10^{-1}
2	1282	2.05	16.04	5.616×10^{-2}
3	5154	2.01	8.00	1.397×10^{-2}
4	20658	2.00	4.00	3.49×10^{-3}
5	82950	2.00	1.99	8.7×10^{-4}
6	329754	1.99	1.00	2.2×10^{-4}

Table 4.2: Meshes used for the grid refinement studies of the thermal material response code with specified temperature boundary conditions.

The first boundary condition to be tested is the specified temperature case. This acts mainly as a check on the interior equations since it represents a Dirichlet boundary condition that is exactly enforced at the boundary. The thermal conductivity

(a) Temperature
(b) Thermal conductivity, κ_{xx}
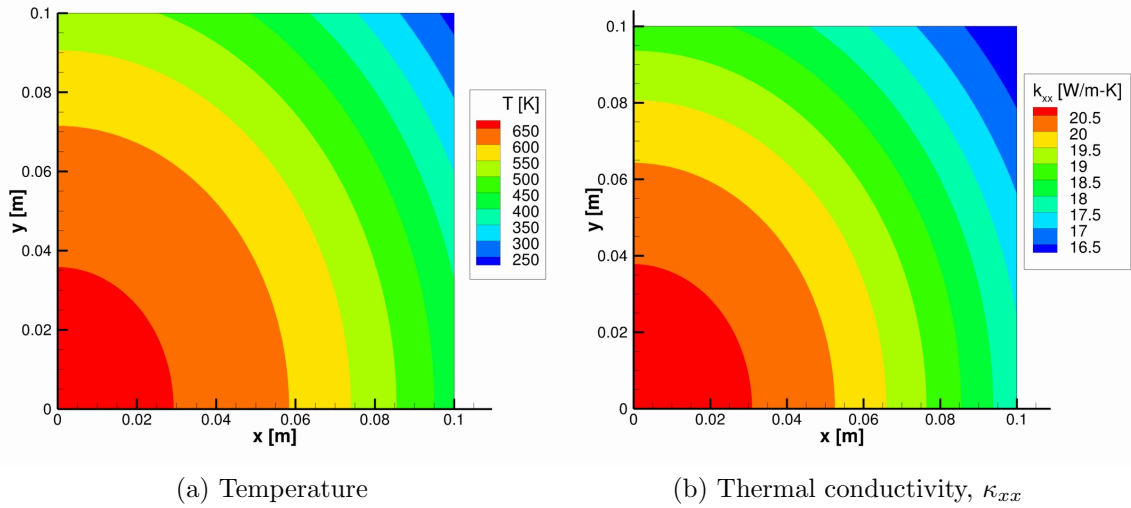
Figure 4.14: Contours of manufactured quantities for the specified temperature test case.

and specific heat are given by Equation 4.5 and are temperature dependent. The thermal conductivity is also anisotropic. In keeping with the assumptions outlined in Chapter II, density is constant and for these tests has a value of $\rho = 1000.0 \ [kg/m^3]$. The manufactured temperature contours and thermal conductivity, κ_{xx}, contours are shown in Figure 4.14, and the constants for the manufactured solution and manufactured properties can be be found in Appendix C in Tables C.4 and C.5. The other components of the thermal conductivity tensor have similar variation, but different values. Specified temperature boundary conditions are applied at all boundaries.

Figure 4.15 shows the observed order of accuracy for this test case at two different times: 1.0 seconds and 4.0 seconds. The order of accuracy that is shown represents the combined spatial and temporal orders of accuracy. Since the observed values are showing second order accuracy in both norms, this confirms that the code is second-order accurate spatially and first-order accurate temporally. Therefore, the interior equations, the time integration, and the specified temperature boundary condition are verified for the thermal material response code.

(a) Order of accuracy after 1.0 seconds   (b) Order of accuracy after 4.0 seconds
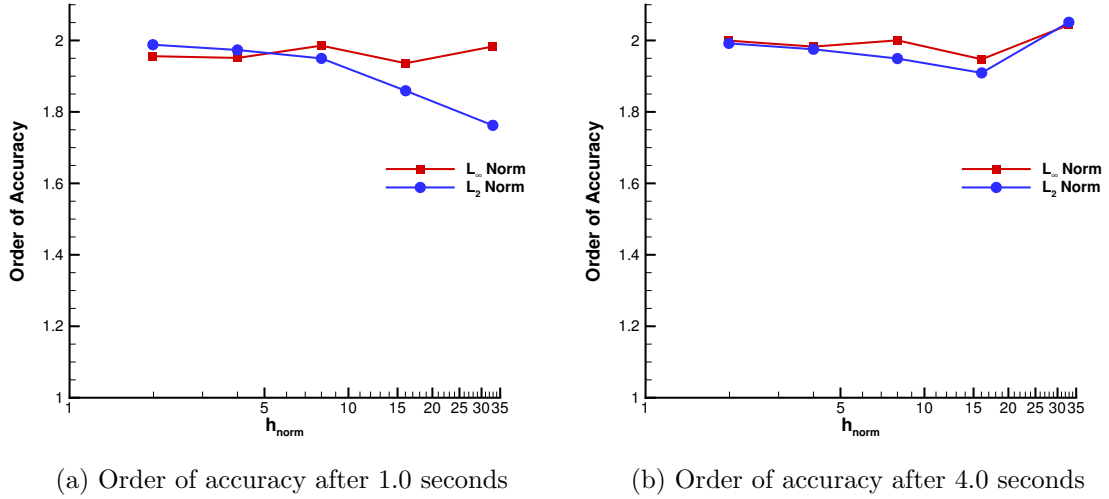
Figure 4.15: Observed order of accuracy for the specified temperature test case.

The specified flux boundary condition case makes use of a different domain geometry in order to test the calculation of normal vectors at the boundaries. The mesh refinement levels used for this case are shown in Table 4.3. The material properties

Refinement Level	Number of Elements	Refinement Ratio (r)	h_{norm}	Time Step [s]
0	42	–	67.68	1.0
1	178	2.05	32.94	2.3684×10^{-1}
2	746	2.05	16.04	5.616×10^{-2}
3	3186	2.01	8.00	1.397×10^{-2}
4	13636	2.00	4.00	3.49×10^{-3}
5	56542	2.00	1.99	8.7×10^{-4}
6	224626	1.99	1.00	2.2×10^{-4}

Table 4.3: Meshes used for the grid refinement studies of the thermal material response code with specified flux boundary conditions.

and temperature solution are the same as the specified temperature boundary condition case, and the manufactured solution constants can once again be found in Table C.4. The manufactured temperature solution is shown in the new geometry in Figure 4.16. Specified flux boundary conditions are applied at all of the boundaries.

The observed order of accuracy for this case is shown in Figure 4.17. Only one simulation time is tested since the time integration is already verified in the specified temperature case. Second order accuracy is again obtained for both norms, so the specified flux boundary condition for the thermal material response code is verified.
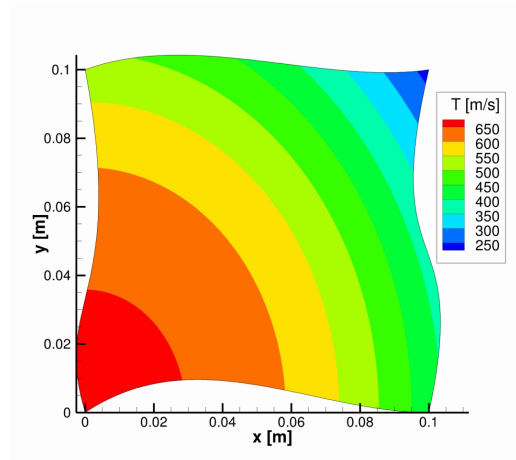
Figure 4.16: Contours of manufactured temperature for the specified flux test case.
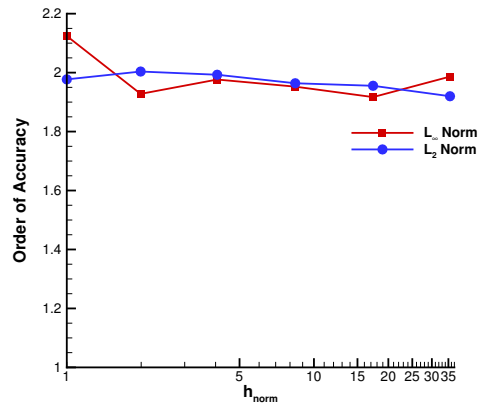
Figure 4.17: Observed order of accuracy for the specified flux test case after 1.0 seconds.

4.3.4   Structural Material Response

Similar to the thermal material response code, the structural response code has two different boundary conditions to be tested: specified displacement and specified traction, which represent a Dirichlet and a Neumann condition, respectively. Table 4.4 shows the meshes used for the grid refinement study. The structural response tests that are conducted using manufactured solutions are static, so unlike the thermal response case it is not necessary to refine both the time step and grid spacing since the temporal accuracy is not tested. The same manufactured solution is used for both boundary conditions, and Figure 4.18 shows the manufactured variables in the deformed configuration. The domain is initially a square with zero displacement everywhere. Similar to the thermal case, the material properties tested are temperature dependent, but in this case they are isotropic. Thermal stresses are considered, and the temperature distribution is also shown in Figure 4.18. This temperature distribution is imposed directly in the code using Equation 4.7 with $t = 1.0$ seconds, and is not computed by the thermal response code. Figure 4.19 shows contours of the temperature dependent material properties. The constants needed for the manufactured solution, temperature distribution, and manufactured material properties are shown in Tables C.6 - C.9 of Appendix C.

Refinement Level	Number of Elements	Refinement Ratio (r)	h_{norm}
0	74	–	67.68
1	290	1.98	32.94
2	1130	1.97	16.04
3	4504	2.00	8.00
4	18088	2.00	4.00
5	72514	2.00	1.99
6	290592	2.00	1.00

Table 4.4: Meshes used for the grid refinement studies of the structural material response code.

(a) x-displacement

(b) y-displacement

(c) Temperature
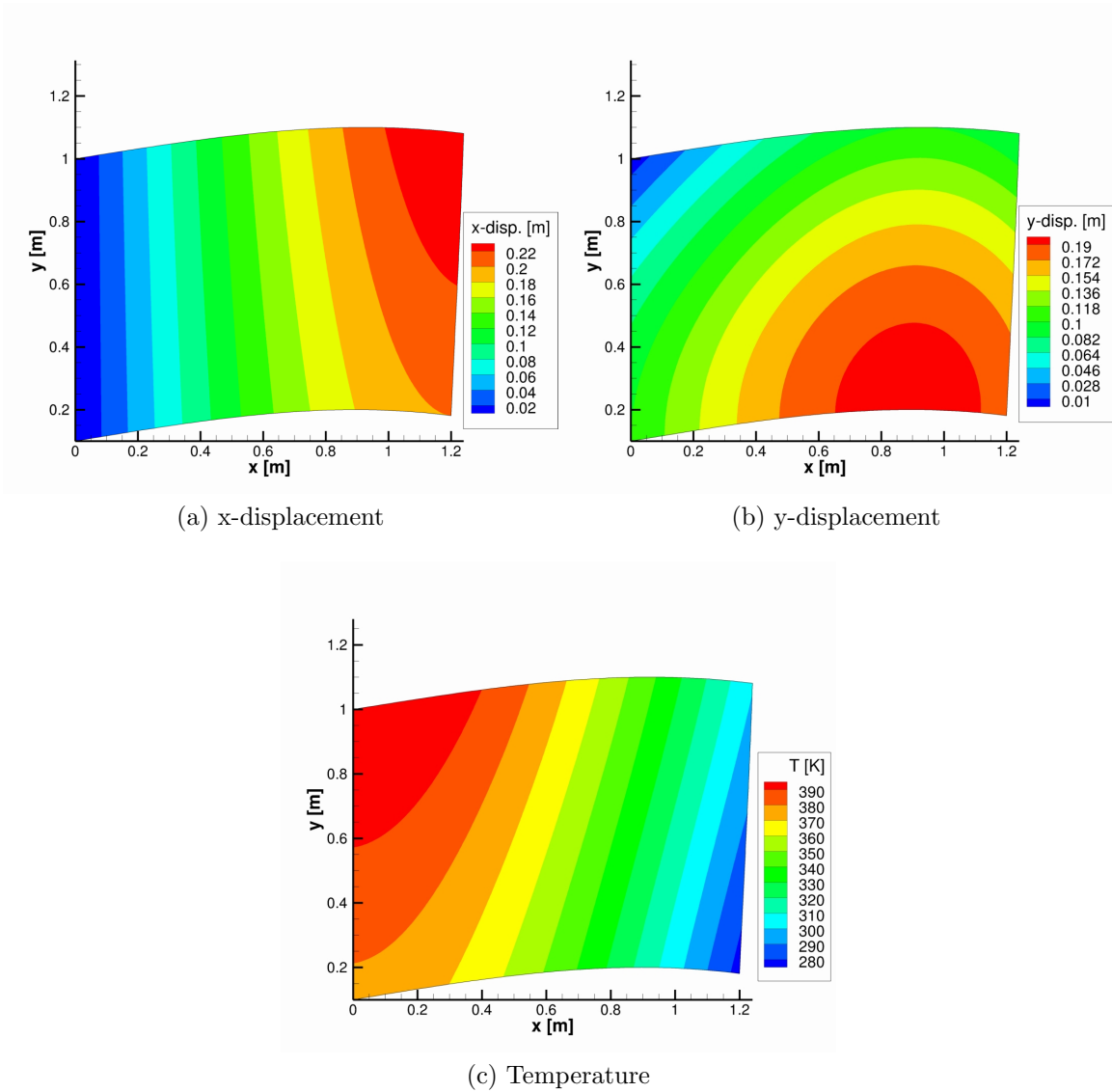
Figure 4.18: Contours of manufactured quantities for the structural test cases.

The specified displacement case serves as a way to test the interior equations, and results for this case are shown in Figure 4.20. The displacement values, which are the dependent variables in the structural simulations, show the expected second-order accuracy in both error norms. The stress values, however, which are derived quantities, show only first-order accuracy. A lower order of accuracy for the stress values is expected for the numerical approach used in this work because stresses are computed using the derivatives of displacement. The displacement is assumed to

(a) Young's modulus

(b) Poisson's ratio

(c) Coefficient of thermal expansion
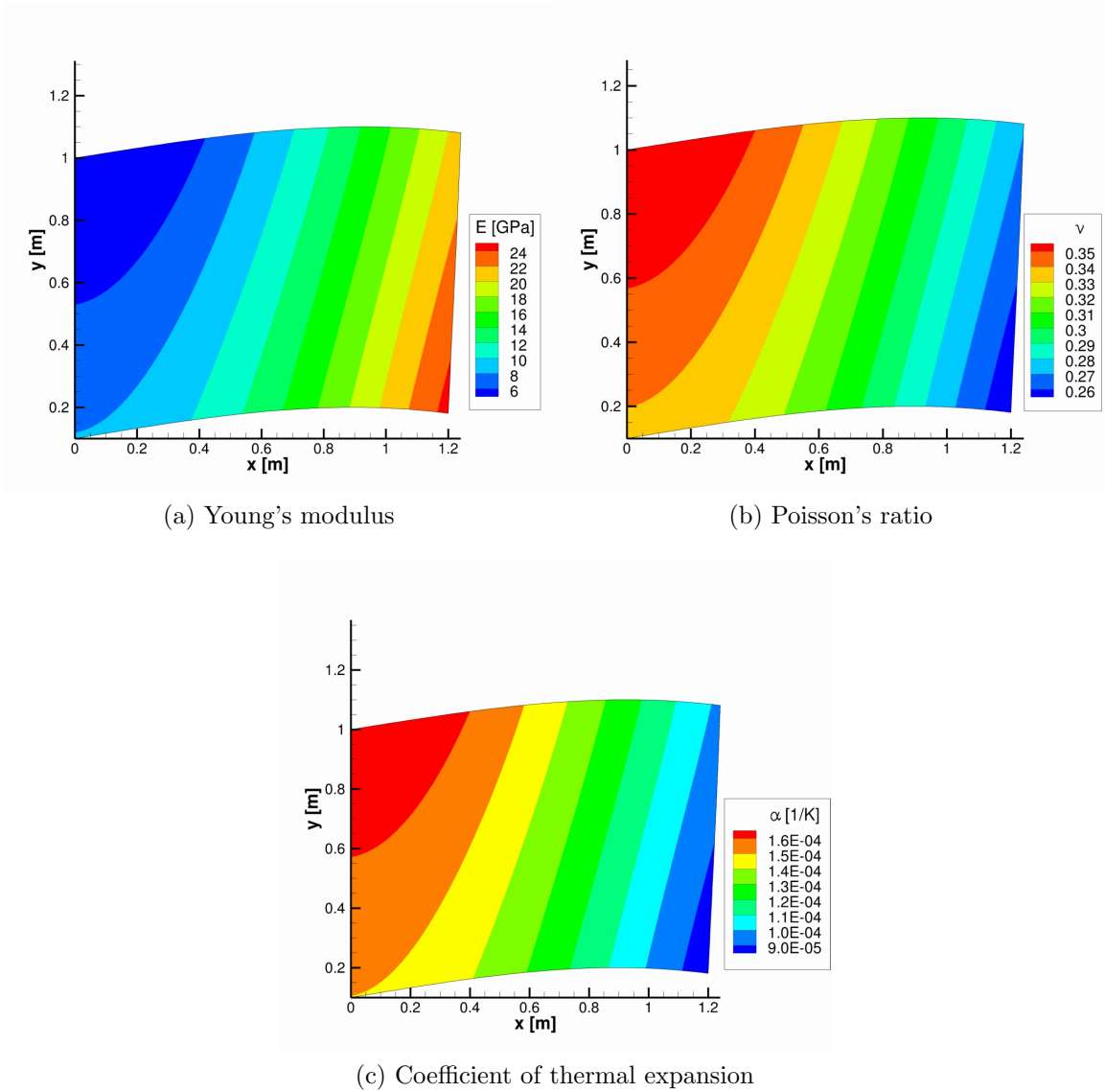
Figure 4.19: Contours of manufactured properties for the specified temperature test case.

vary linearly across an element, which means that the derivatives of displacement are constant across an element. Having derivatives that are constant within an element leads to first-order accurate stress calculations. While not shown here, the error in the strain calculations is also of first-order accuracy for the same reason. Therefore, observing first-order accuracy in the stress calculations and second-order accuracy in the displacement results verifies the implementation of the interior equations and also
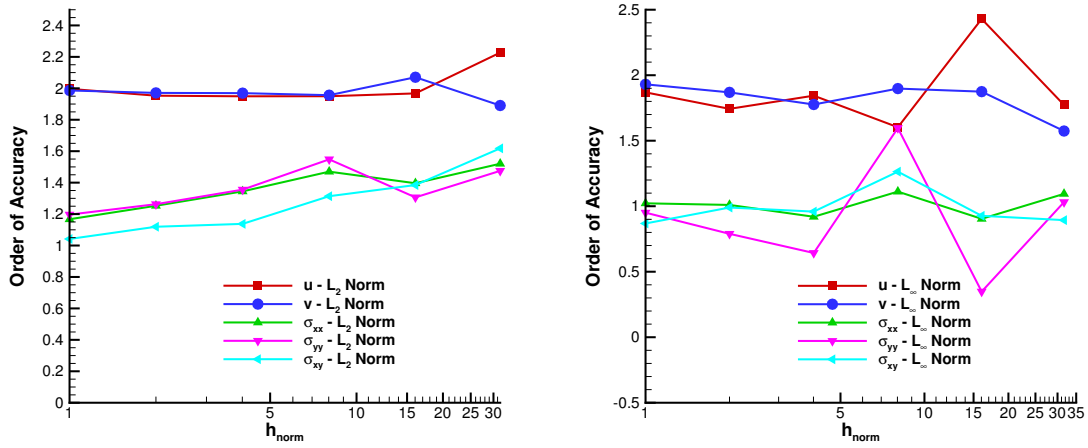
(a) Order of accuracy based on L_2 norms (b) Order of accuracy based on L_∞ norms

Figure 4.20: Observed order of accuracy for the specified displacement test case.

the specified displacement boundary conditions.

For the specified traction case, displacement boundary conditions are applied on the left boundary at $x = 0.0$, and traction boundary conditions are used on the remaining boundaries. The reason for this is that applying traction boundary conditions at all boundaries results in a numerical problem that is not well-posed. Fixing one edge of the boundary allows the problem to be solved computationally, and still provides an adequate test of the specified traction boundary condition. The results for this case are shown in Figure 4.21. The L_2 norms indicate the expected order of accuracy for both the displacements and the stresses. The L_∞ norms show the expected first-order accuracy for the stresses, but seem to be converging to an order of accuracy between first and second-order for the displacements.

The slightly lower than expected observed order of accuracy is due to localized errors in the corners of the domain that decrease at a slightly slower rate than the errors in the rest of the domain. This is the result of the way that the control volumes are constructed at the domain corners, and is not indicative of an error in the code. Therefore, the specified traction boundary condition can be considered verified.

87
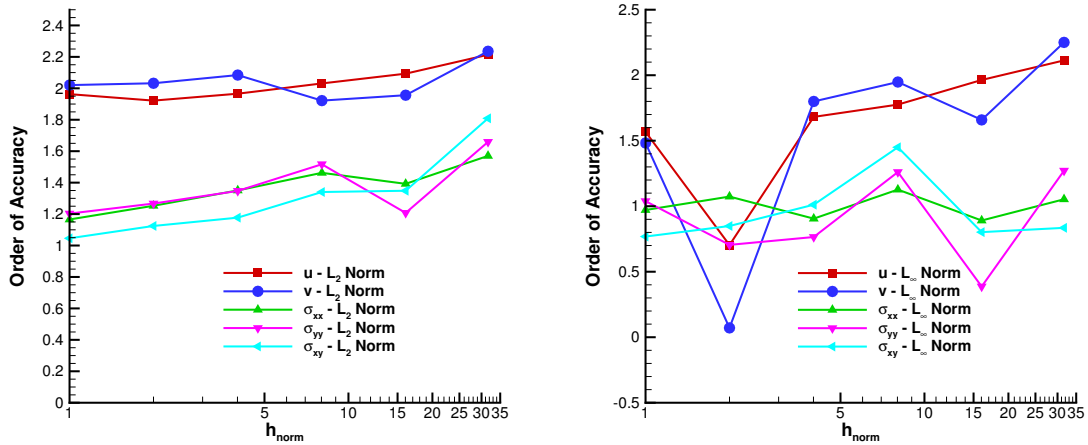
(a) Order of accuracy based on L_2 norms

(b) Order of accuracy based on L_∞ norms

Figure 4.21: Observed order of accuracy for the specified traction test case.

The L_∞ order of accuracy could be improved, however, by nonuniform refinement of the mesh in the corners of the domain. This is a technique that is often used in structural analysis for dealing with problems that contain singularities [110]. It is worth noting, though, that the local error in the domain corners does not seem to impact the solution in the rest of the domain. Therefore, depending on the quantities of interest, it may not be necessary to overly refine the mesh corners in order to get a practical solution.

4.3.5   Aerothermoelastic Coupling Framework

As discussed in Chapter III, a partitioned approach is used in this work to couple the CFD and material response codes. Since the partitioned approach uses completely separate codes for each sub-problem and the coupling is enforced through boundary conditions, verification of each code can be performed individually as long as the boundary conditions that are relevant to the coupling procedure are also tested. The boundary conditions that are needed for a fully coupled simulation include the no-slip wall boundary in LeMANS, specified heat flux in the thermal response code, and

specified traction in the structural response code. All of these boundary conditions were tested in the previous sections. The only remaining aspect of the procedure that is relevant to the test cases considered in Chapter V of this thesis is the spatial interpolation of boundary values between the fluid and material response codes.

Testing the interpolation is accomplished by defining manufactured solutions for temperature in both the fluid and solid domains that have identical temperature values and temperature gradients at the fluid-solid interface. If the same values for thermal conductivity are used in both the fluid and solid, then the heat flux computed by LeMANS at the interface and interpolated to the thermal response code will act as the manufactured boundary heat flux for the thermal code. While it is possible to test the code-to-code interpolation in a simpler manner, the approach used here is chosen for two reasons. First, it provides a test of all of the code that deals with interpolating variables from one mesh to another exactly as it is used in a real case. Second, even though the aerodynamic heating boundary condition for the coupled case uses the same implementation as the specified heat flux boundary condition, it corresponds to a different branch of the thermal response code, which will be tested using this approach.

It is only necessary to test the inter-code interpolation in one direction (i.e. Le-MANS $\rightarrow$ material response or material response $\rightarrow$ LeMANS) since the two cases are the same, only differing in the position at which the interpolating values are located. Additionally, the interpolation of pressures and stresses is identical to the interpolation of temperature, and the specified traction boundary condition used in the coupled framework is the same code as for the uncoupled case, so interpolation between the fluid and structural material response codes does not need to be tested in addition to the thermal response interpolation.

The coupled manufactured solution case uses a single mesh for the fluid domain and the series of meshes shown in Table 4.5 for a grid refinement study in the solid

domain. The mesh used in the fluid domain is the same as was used in the verification tests of LeMANS and corresponds to refinement level 4 in Table 4.1. This mesh is chosen so that the number of flow boundary nodes at the fluid-solid interface is higher than the number of solid boundary nodes for the coarse solid meshes, and lower than the number of solid boundary nodes for the more refined solid meshes. The thermal response code uses specified temperature boundary conditions at the non-coupled boundaries, and LeMANS uses inlet boundary conditions at the non-coupled boundaries in the same way as for the viscous wall test case.

Refinement Level	Number of Elements	Refinement Ratio (r)	h_{norm}	Time Step [s]
0	64	–	36.31	1.0
1	288	2.12	17.12	2.5×10^{-1}
2	1258	2.09	8.19	6.25×10^{-2}
3	5110	2.02	4.06	1.5625×10^{-2}
4	20874	2.02	2.01	3.90625×10^{-3}
5	84370	2.01	1.00	9.765625×10^{-4}

Table 4.5: Meshes used for the grid refinement studies of the thermal response code coupled to LeMANS.

Equations 4.15 and 4.16 give the manufactured solutions for the flow and solid domains. The value of C_s is 0.25, and the other necessary constants for these equations are listed in Table C.10 in Appendix C. Both the fluid and solid domains use a constant, isotropic thermal conductivity equal to 15.0 $[W/m\text{-}K]$. The density values, which are shown in Figure 4.22 are chosen so that there is no mass flux to the wall. This eliminates the mass diffusion heat flux, and makes matching the fluid and solid manufactured solutions much simpler, while still providing a good test of the coupling procedure. The manufactured temperature for the flow and solid domains is shown in Figure 4.23. At the fluid-solid interface, the temperature and heat flux are identical in both domains, however, away from the interface the solid temperature distribution has an additional time dependent component that is not present in the flow solution.

Note that the manufactured velocity components in LeMANS are not important for this test case, since only the thermal solution is sought in the solid domain.

$$\left.\begin{aligned}
\rho_s(x,y) &= \rho_{s_0} + \rho_{x_s}\cos\left(\frac{a_{x_s}\pi x}{L}\right) + \rho_{y_s}\sin\left(\frac{a_{y_s}\pi y}{L}\right) + \rho_{xy_s}\cos\left(\frac{a_{xy_s}\pi xy}{L^2}\right) \\[6pt]
u(x,y) &= (f(x,y) - C_s)\left\{u_0 + u_x\sin\left(\frac{a_{x_u}\pi x}{L}\right) + u_y\cos\left(\frac{a_{y_u}\pi y}{L}\right)\right. \\[4pt]
&\qquad\left. + u_{xy}\cos\left(\frac{a_{xy_u}\pi xy}{L^2}\right)\right\} \\[6pt]
v(x,y) &= (f(x,y) - C_s)\left\{v_0 + v_x\sin\left(\frac{a_{x_v}\pi x}{L}\right) + v_y\cos\left(\frac{a_{y_v}\pi y}{L}\right)\right. \\[4pt]
&\qquad\left. + u_{xy}\cos\left(\frac{a_{xy_u}\pi xy}{L^2}\right)\right\} \\[6pt]
T_{tr}(x,y) &= T_0 + T_x\cos\left(\frac{a_{x_T}\pi x}{L}\right) + T_y\cos\left(\frac{a_{y_T}\pi y}{L}\right) + T_{xy}\cos\left(\frac{a_{xy_T}\pi xy}{L^2}\right)
\end{aligned}\right\} \tag{4.15}$$

$$\begin{aligned}
T_s(x,y,t) &= T_t\sin\left(\frac{a_t\pi t(y-C_s)^2}{T_0}\right) + T_0 + T_x\cos\left(\frac{a_{x_T}\pi x}{L}\right) + T_y\cos\left(\frac{a_{y_T}\pi y}{L}\right) \\[4pt]
&\quad + T_{xy}\cos\left(\frac{a_{xy_T}\pi xy}{L^2}\right) \\[8pt]
&= T_t\sin\left(\frac{a_t\pi t(y-C_s)^2}{T_0}\right) + T_{tr}(x,y)
\end{aligned} \tag{4.16}$$

The procedure for the coupled tests is as follows. First, the flow domain and coupled variables are initialized to the exact manufactured solution. Then, LeMANS is run as normal and interpolates the flow values to the solid domain boundaries, and then calls the material response code. The material response is run to a simulation time of 1.0 seconds, and the results are returned to LeMANS. At this point, since the order of accuracy of LeMANS is not being tested, the case is finished. This is repeated for each of the solid mesh refinements shown in Table 4.5.

The order of accuracy results for the coupled test case are shown in Figure 4.24. The L_2 norms show second-order accuracy, however, the L_∞ norms only indicate first-order accuracy. The lower order of accuracy is due to how the thermal response

(a) Density of N$_2$ (b) Density of N
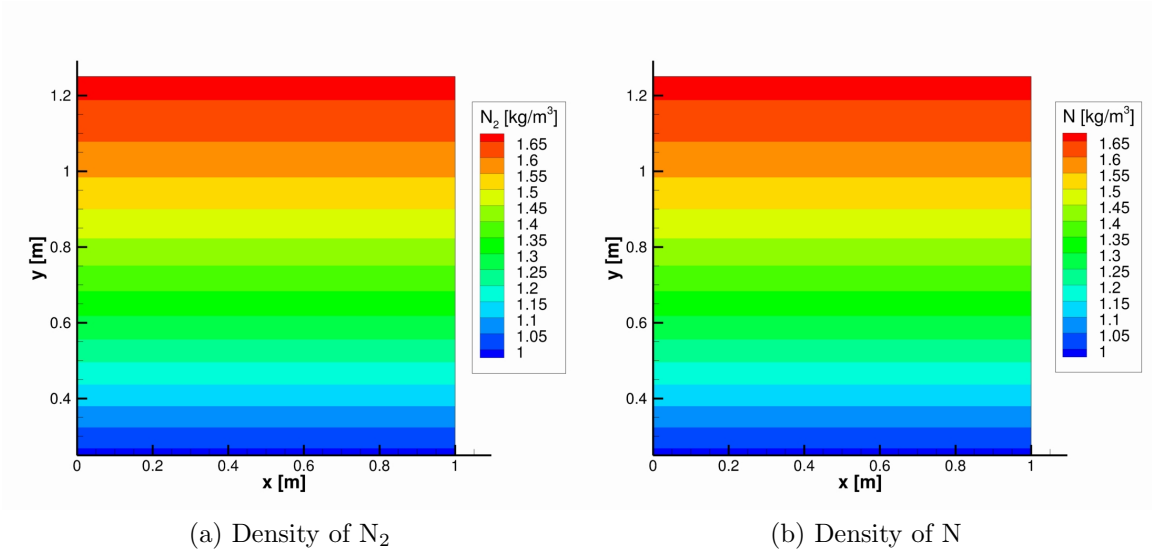
Figure 4.22: Contours of manufactured density in LeMANS for the coupled test case.
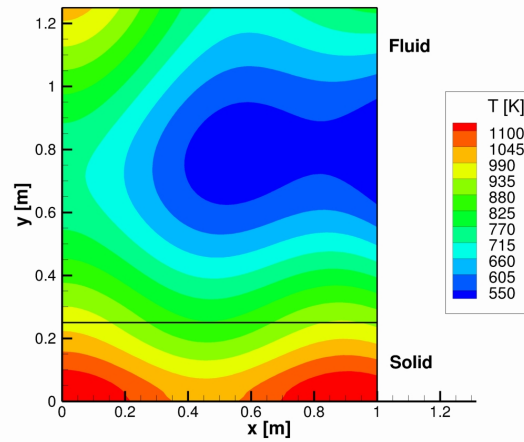
Figure 4.23: Contours of manufactured temperature in the fluid and solid domains for the coupled test case.

code handles the corners of the domain at $(x, y) = (0.0, 0.25)$ and $(x, y) = (1.0, 0.25)$ where the different boundary condition types meet. This is a result of how the control volumes at the intersection of two boundary conditions are constructed, and is not indicative of an error within either the thermal response code or the coupling procedure.
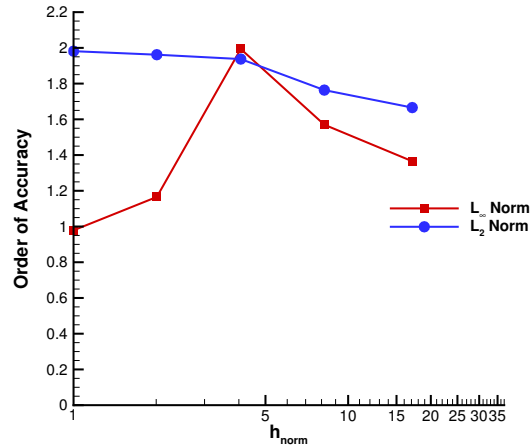
Figure 4.24: Observed order of accuracy of the thermal response code for the coupled test case at $t = 1.0$ seconds.

Figure 4.25 shows the error contours on two different meshes for the corner of the solid domain at $(x, y) = (1.0, 0.25)$. The boundary condition types used on the different sides of the domain are also shown. From these plots it is clear that there is a highly localized region of larger error present where the different boundary conditions meet. This is due to the fact that the error at the corner of the domain decreases at a slower rate than the error in the rest of the domain, which leads to regions of larger error being present on the more refined meshes. Similar to the traction boundary condition case shown earlier, these local regions of error could be decreased to the level of the rest of the domain by refining the mesh in the corners. Based on both the observed order of accuracy and the error contours, however, the locally large errors do not propagate into the rest of the solution domain, and so in practice it may not be necessary to use a highly refined mesh in these regions.

Based on the results presented in this chapter, the interior equations of LeMANS, the thermal material response code, and the structural material response code are verified. The boundary conditions in each of these codes that are relevant to the aerothermoelastic framework are also found to be correctly implemented, and the in-

(a) Refinement level 4   (b) Refinement level 5
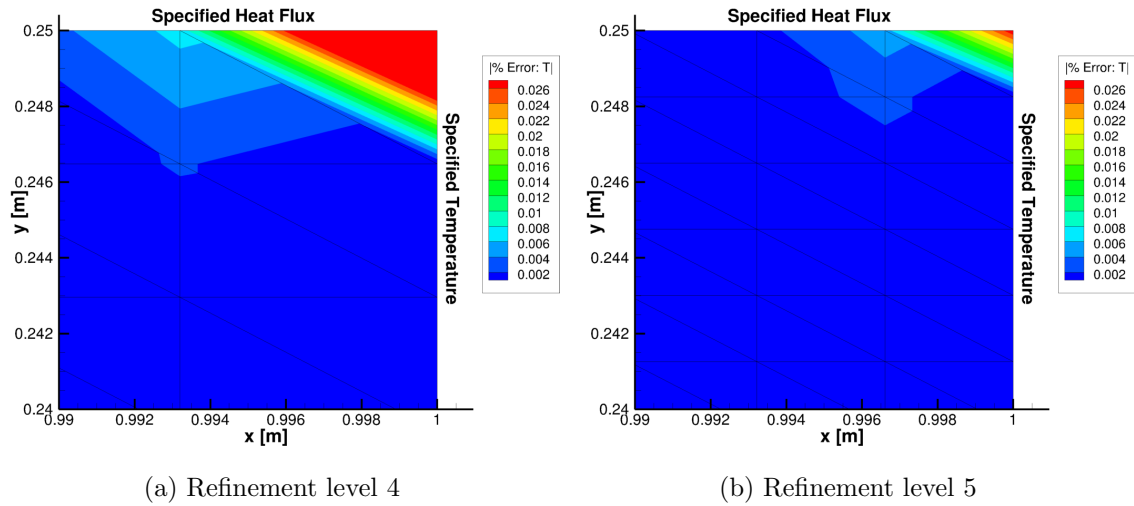
Figure 4.25: Contours of percentage temperature error for the corner of the coupled solid domain at two different mesh refinements.

terpolation of boundary values between the codes is found to not reduce the observed order of accuracy. These results provide confidence that the aerothermoelastic framework is correctly implemented, and that it provides valid results within the limits of the models that are used.

CHAPTER V

Aerothermal and Aerothermoelastic Test Cases

5.1 Introduction

The previous chapter presents a series of code verification exercises which help to build confidence in the implementation of the various pieces of the aerothermoelastic framework. Verification exercises are important, but it is also desirable to test the framework on physically realistic problems. In particular, it is useful to study problems where experimental data is available, or where it is possible to compare results with other codes. When dealing with hypersonic flows and coupled material response, there is very limited experimental data that is available and useful for comparison with a numerical code. So, in the present work, two test cases are studied which allow for comparison with other codes and with lower fidelity models in order to build confidence in the aerothermoelastic framework, and to demonstrate the code capabilities.

In this chapter, two different test cases are considered in order to investigate the performance of the aerothermoelastic framework on realistic problems. The first test case is of the IRV-2 vehicle, which is an unmanned reentry vehicle. The vehicle's reentry trajectory is discretized into a series of points, and a coupled aerothermal solution is found at a number of these points. The structural material response is not considered for this problem, so this is an aerothermal problem rather than a full

aerothermoelastic problem. Non-charring ablation is modeled at the IRV-2 surface, and the associated surface recession is taken into account. The results from the current work are compared with results from a one-dimensional material response code [63, 64], and also with the ABRES Shape Change Code (ASCC) [52].

The second test case consists of a two-dimensional wedge in hypersonic flow, with a compliant panel located over a portion of the wedge. This is a full aerothermoelastic problem involving LeMANS and both the thermal and structural material response modules. The geometry and test conditions for this case come from a study [27, 26] that uses a similar aerothermoelastic coupling strategy as the present work, but simplified flow and structural models. Some of the differences in results due to the different models are discussed, and a number of studies are presented on how changing various parameters of the coupling procedure affects the results. The purpose of this test case is to gain insight into how the aerothermoelastic coupling procedure performs, and to test whether or not reasonable results are generated by assuming a steady state flow field and quasi-static thermo-structural response.

5.2 The IRV-2 Vehicle

The IRV-2 test case is used to investigate the aerothermal coupling portion of the aerothermoelastic framework, along with a surface ablation boundary condition. Structural material response is not considered. Results from the present work are compared with results from a one-dimensional material response code, and with results from ASCC. An overview of the test case, as well as a discussion of the results is presented.

5.2.1 IRV-2 Geometry and Test Conditions

The IRV-2 vehicle, shown in Figure 5.1, has a sphere-biconic geometry, and a total length of 1.386 m. The nose radius is 0.01905 m, and the two cone angles are 8.42° and 6.10° [55]. The only portion of the vehicle that is modeled is the nosetip, which consists of the sphere and the first cone, and extends axially 0.1488 m from the stagnation point. This geometry is modeled as an axisymmetric sphere-cone, and a schematic of the computational domain is shown in Figure 5.2 along with the boundary conditions that are used. Figure 5.3 shows the initial flow and solid meshes that are used. The flow mesh contains 8448 cells, and the solid mesh contains 1106 nodes, which corresponds to the number of control volumes using the CVFEM for spatial discretization. These meshes will evolve slightly in shape as the vehicle surface recesses, however, the recession for this particular test case is not large enough to greatly alter the shape of the vehicle, and the number of mesh cells does not vary. All of the results presented in this chapter are computed on a single processor for both the fluid and solid domains. Computational times for each case considered are on the order of 50 hours of CPU time.
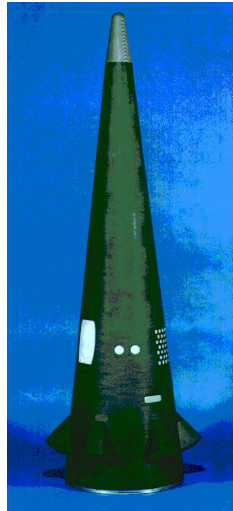
Figure 5.1: The IRV-2 vehicle [55].

1. Supersonic inlet
2. Supersonic outlet
3. Symmetry
4. Coupled, no-slip wall
5. Aerodynamic heating, thermochemical ablation
6. Adiabatic
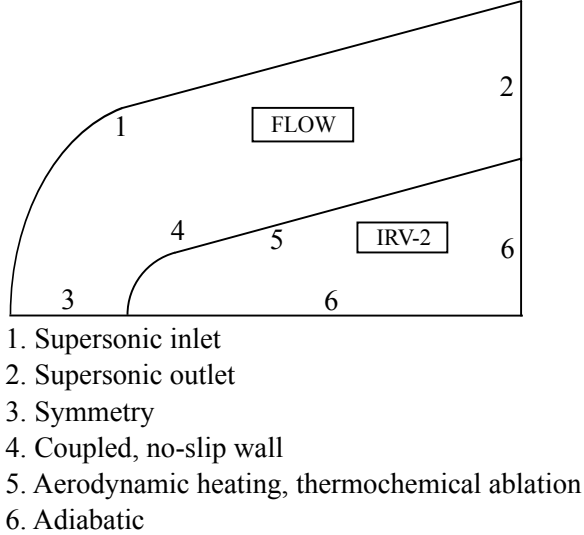
Figure 5.2: The computational domain and boundary conditions for the IRV-2 simulations.

The IRV-2 vehicle follows a ballistic trajectory, which Kuntz et al. [55] split into a series of 23 discrete points, with each trajectory point representing a flight time at which a coupled solution is found. In the present work, only the first four trajectory points are considered. The free stream conditions for these points are listed in Table 5.1. Trajectory point 0 is a flow-only point, so no material response is computed, and the flow solution is only used as the initial conditions for trajectory point 1. The remaining trajectory points use the coupled approach presented in Chapter III, with the flow boundary conditions given in Table 5.1, and the initial conditions provided by the solution at the previous trajectory point. The solid domain is initialized to a constant temperature of 300.0 K at trajectory point 1, and the following trajectory points are initialized with the temperature distribution found at the previous trajectory point. In order to allow for comparison with the one-dimensional material response results found in [64], a 5-species air model consisting of N_2, O_2, NO, N, and O is used in LeMANS. So, while the mass flux of ablation species into the flow field is considered, the chemistry between the ablation products and the flow species is neglected.

98
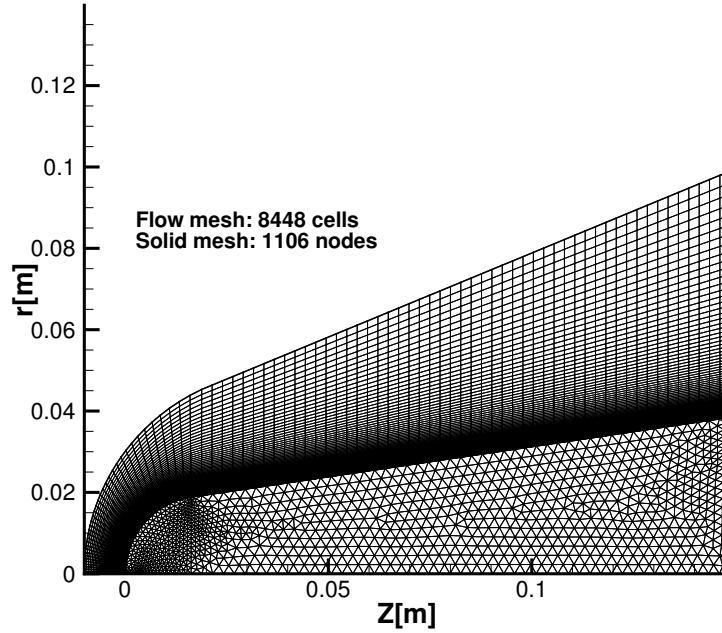
Figure 5.3: Initial flow and solid meshes for the IRV-2 simulations.

The material properties used to model the IRV-2 carbon-carbon nosetip are for generic non-charring carbon and are taken from Amar [5] and listed in Appendix D. Carbon-carbon is a non-decomposing ablator, so no pyrolysis gas is generated and chemical ablation occurs only at the material surface. The B' table approach is used to determine the wall enthalpy and char mass blowing rate. The necessary surface thermochemistry data is generated using the Aerotherm Chemical Equilibrium (ACE) code [78].

For this test case, a fixed iteration coupling procedure is used, and the material response code is called every 100 flow field iterations. The value of 100 was chosen based on experimentation with this test case, and was found to lead to a stable coupling scheme. The somewhat arbitrary process for choosing the flow coupling parameter highlights one of the drawbacks of the fixed iteration coupling procedure. The method proposed by Hogan et al. [48] is used to iteratively solve for the surface

Trajectory point	Reentry time [s]	Altitude [km]	Mach number	Temperature [K]	Density [kg/m^3]
0	0.00	66.7	22	227.8	1.25×10^{-4}
1	4.25	56.0	21	258.0	5.05×10^{-4}
2	6.75	49.3	21	270.7	1.13×10^{-3}
3	8.75	44.0	21	261.4	2.26×10^{-3}

Table 5.1: Free stream conditions for the first four IRV-2 trajectory points.

recession rate as a function of temperature and pressure. A constant time step of $\Delta t = 0.1$ seconds is used for the thermal material response. At each of the trajectory points presented, the partitioned aerothermal coupling approach is robust and leads to converged flow field results.

5.2.2 Comparison of One-Dimensional and Multi-Dimensional Material Response

The one-dimensional material response results are obtained using the MOPAR code [62] that has been developed at the University of Michigan. MOPAR uses the same governing heat conduction equation and ablation boundary condition as the code developed in this work. The one-dimensional results are obtained at each boundary face in the flow mesh along lines that are normal to the faces of the flow mesh which lie along the fluid-solid interface. Each line has a constant length equal to the IRV-2 nose radius, and an adiabatic boundary condition is set at the end of the line that lies within the solid domain. Once one-dimensional results are obtained for each mesh face along the boundary, the contours shown in Figures 5.5 and 5.6 are generated by combining all of the one-dimensional solutions together using the interpolation routines available in Tecplot [92].

Figure 5.4 shows the temperature contours for the axisymmetric material response at trajectory points 1 and 3. The nosetip heats up very rapidly from the initial temperature of 300.0 K, and there is very little surface recession over the course of

the trajectory points that are considered. At the $t = 8.75$ s trajectory point, the shock becomes thinner, and also moves slightly closer to the vehicle surface. The temperature of the IRV-2 vehicle increases over all of the trajectory points, with the highest temperatures found in the nose region. The flow of heat is much greater in the axial direction than in the radial direction due to the much larger aerodynamic heating that is present over the nose of the vehicle.

(a) Full view

(b) Close-up of stagnation region
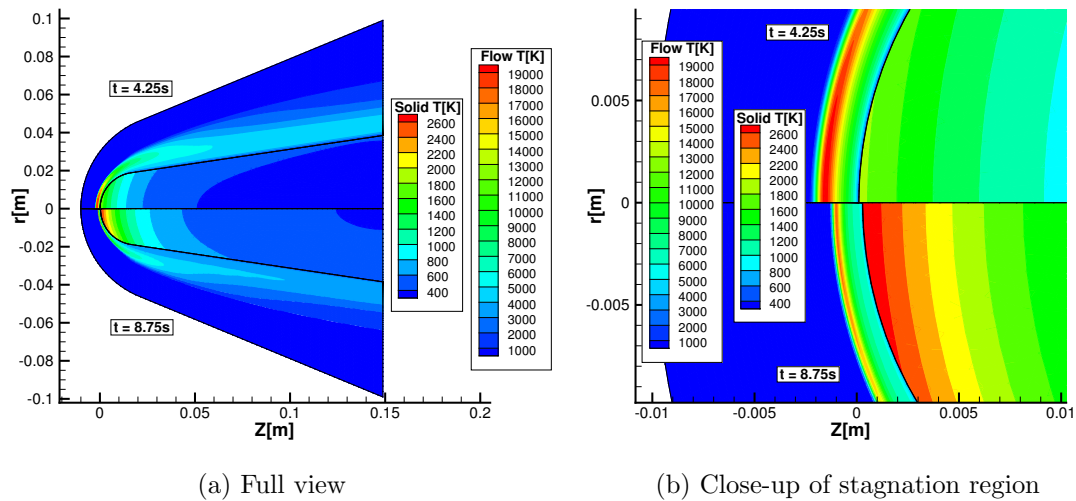
Figure 5.4: Temperature contours for the axisymmetric solutions at $t = 4.25$ s and $t = 8.75$ s.

Figure 5.5 shows a comparison of temperature contours between the one-dimensional and axisymmetric codes at trajectory point 1. The flow field results are nearly identical to one another, which is not surprising, since in both cases the flow solution is axisymmetric and the surface values do not change drastically between the different models. The in-depth solid solutions are noticeably different, however, especially in the shape of the temperature contours. The one-dimensional material response results do not adequately capture the multi-dimensional nature of the heat flow, especially in the highly curved nose region. The effect of using one-dimensional lines of constant length is also evident in how the temperature penetrates into the material. Looking along the centerline of the material it can be seen that using lines of length equal to

the nose radius (0.01905 m) causes an adiabatic boundary condition to be set at a location where it is not physically accurate to do so. At a location of $z = 0.01905$ m, it is evident from the axisymmetric solution that there are still significant temperature gradients, so using an adiabatic boundary condition at that location impacts how the in-depth temperature profiles develop.
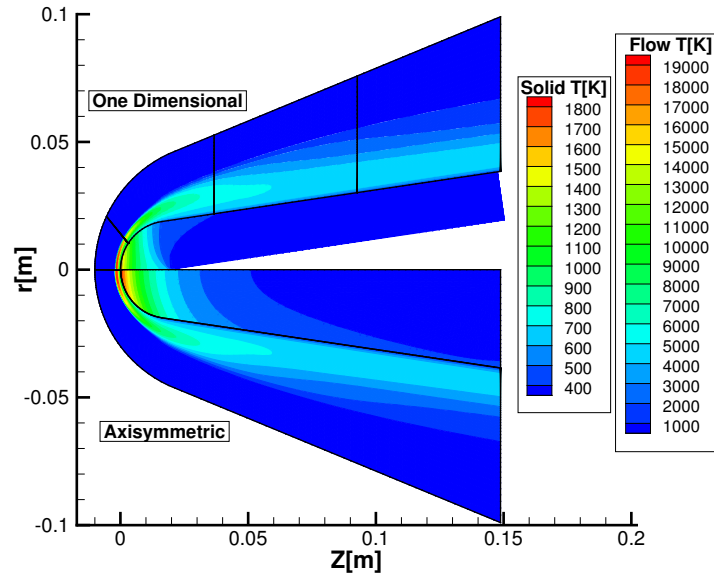
Figure 5.5: Comparison of one-dimensional and axisymmetric results at $t = 4.25s$.

To further examine the differences between the one dimensional and axisymmetric material response models, the temperature contours in the solid are shown for the last three trajectory points in Figure 5.6. The multi-dimensional nature of the heat flow is clearly evident from the temperature contours, and this leads to several observations. First, near the vehicle's surface, the axisymmetric contours are more curved. This causes higher temperatures to be found further aft along the surface than for the one-dimensional model. Second, along the stagnation line ($r = 0.0$ m) the one-dimensional contours become elongated and distorted as the back of the solution line is approached. This is due in part to setting an adiabatic boundary condition at a location where it is not physically accurate to do so, but it is also due to the fact that

102

(a) $t = 4.25s$

(b) $t = 6.75s$

(c) $t = 8.75s$
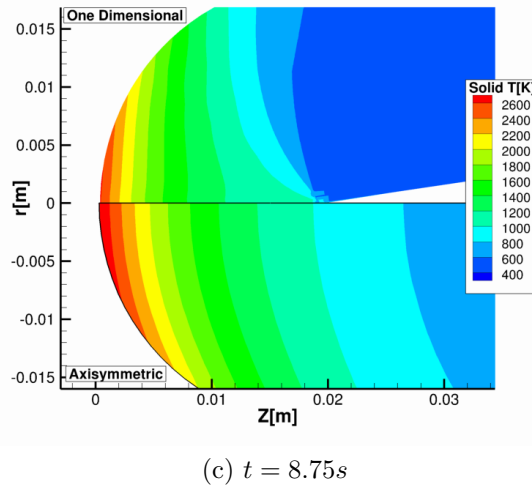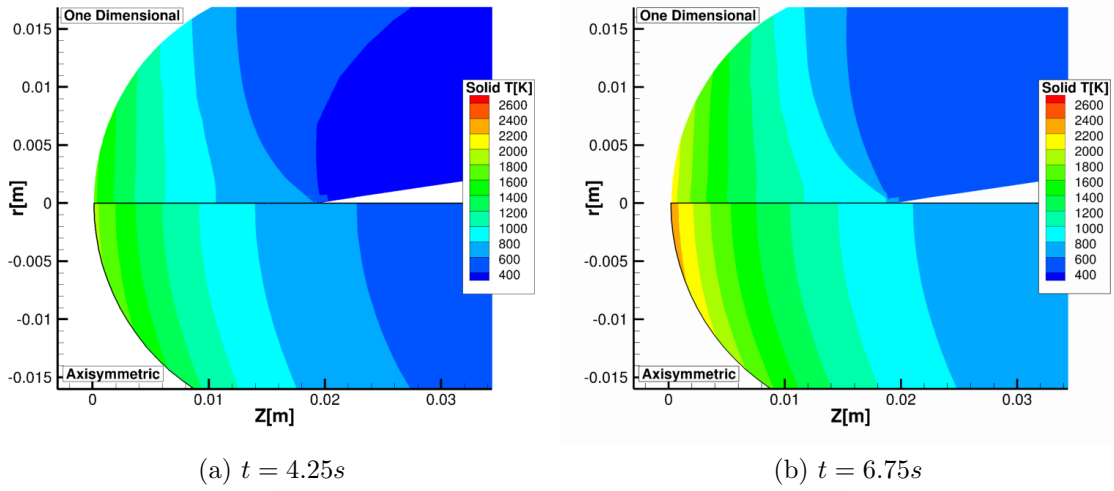
Figure 5.6: Comparison of one-dimensional and axisymmetric temperature contours.

the high temperatures along the stagnation line are unable to conduct radially into the material. In the axisymmetric case, these problems are alleviated, leading to a more physically accurate prediction of the in-depth material response.

Figure 5.7 shows a comparison of several different surface properties between the one-dimensional and axisymmetric results. The heat flux, ablation mass flux, and rate of recession predicted by the one-dimensional approach are all higher near the stagnation point than the corresponding axisymmetric results. However, the stagnation point temperature of the one-dimensional model is lower than that obtained

with the axisymmetric approach. The lower surface temperature leads to the higher heat flux from LeMANS due to the larger difference between the flow and solid temperatures. The higher ablation rate in the one-dimensional results leads to a larger surface recession in the stagnation region.

(a) Heat flux

(b) Temperature

(c) Mass blowing rate

(d) Surface shape
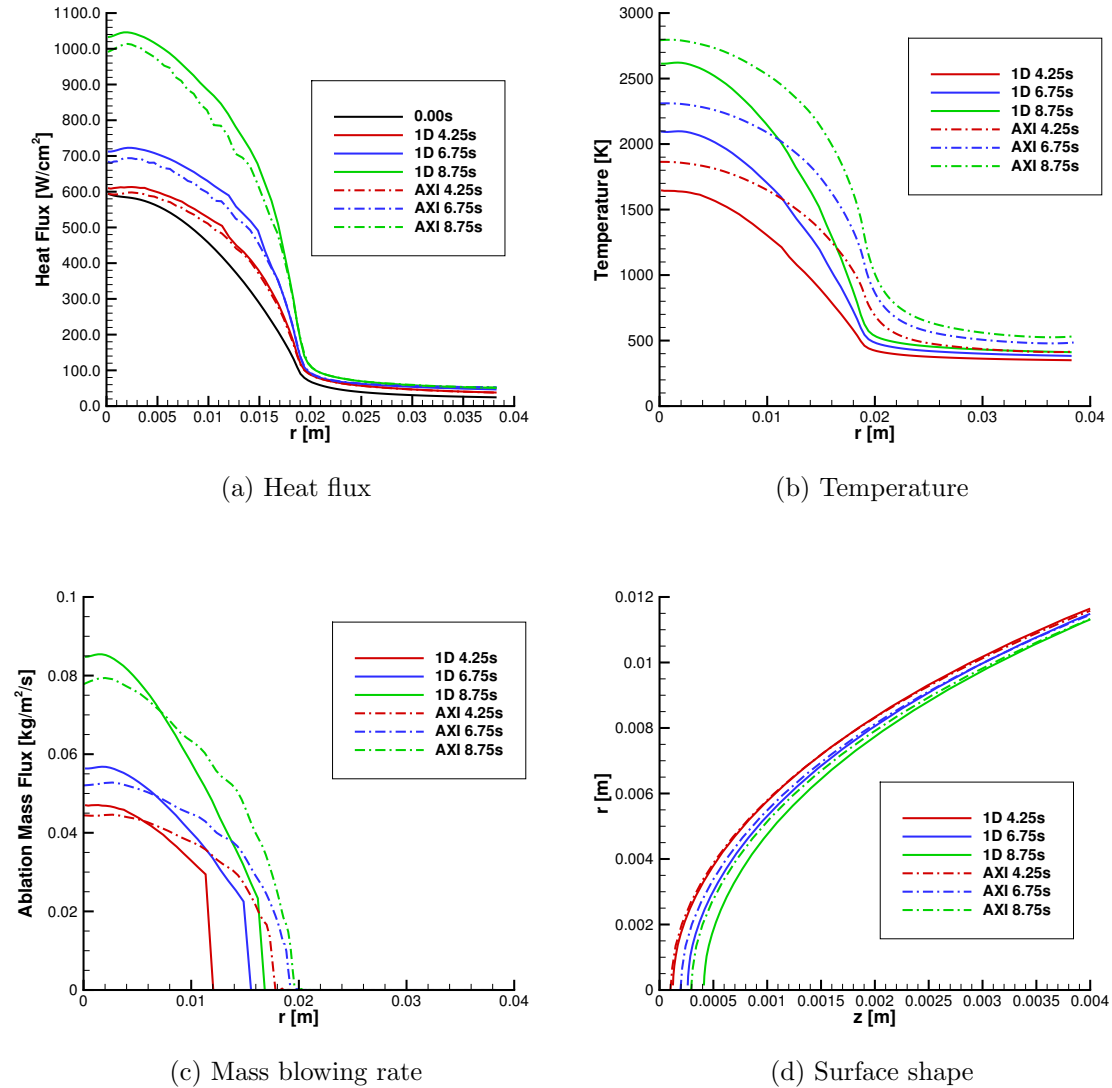
Figure 5.7: Comparison of several different surface properties between one-dimensional and axisymmetric results.

The axisymmetric temperature and mass flux results show broader profiles than the one-dimensional results, especially for the earlier trajectory times. This is the

same trend that was evident in Figure 5.5 where the contours in the multi-dimensional case were curved further aft, which indicates that higher temperatures were maintained further along the vehicle's surface. This is a result of the heat being able to conduct along the surface and into the material rather than just into the material as in the one-dimensional case.

5.2.3 Comparison With ASCC

In addition to comparing results between the one-dimensional and axisymmetric material response codes, the axisymmetric code is also compared against results obtained with ASCC [55]. Figure 5.8a compares the predicted stagnation point temperatures between ASCC and both the axisymmetric and the one-dimensional LeMANS/material response coupled results. Both models predict lower stagnation point temperatures than ASCC for all of the trajectory points that are considered, and the agreement with ASCC decreases for both codes at each successive trajectory point. Figure 5.8b shows a comparison of the stagnation point surface recession between the axisymmetric model and ASCC. ASCC predicts a higher surface recession rate with an increasing difference between LeMANS and ASCC for later trajectory points, which is consistent with the temperature comparison.

The differences in results between ASCC and the present work are likely due to several causes. First, the flow model used in this work does not include carbon-in-air chemistry. This was done to allow for comparison with the one-dimensional code, but it will impact the comparison with ASCC. Additionally, the surface chemistry model that is used can greatly impact the results [40], and differences between ASCC and the present work would contribute to the observed differences. The exact material properties used in ASCC are also not known, and the material model used in this work may have slightly different properties.

It is clear from the results presented that accounting for the multi-dimensional

105

(a) Stagnation point temperatures   (b) Surface recession
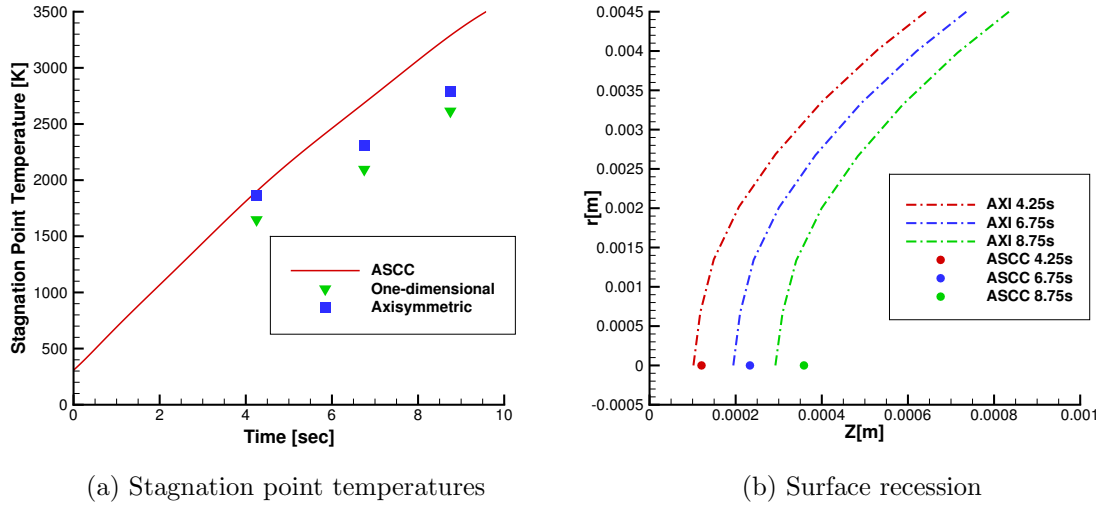
Figure 5.8: Comparison of stagnation point quantities between ASCC and the Le-MANS/material response coupled results.

heat conduction within a material can significantly alter the results when compared to a one-dimensional heat conduction model. This is especially true in geometries with relatively large curvature, such as the nose region of the IRV-2 vehicle. Additionally, when comparing the present results with ASCC, it is likely that accurate flow field and surface chemistry modeling play a larger role in matching the ASCC data than whether a one-dimensional or multi-dimensional material response model is used.

5.3 Aerothermoelastic Compliant Panel

The purpose of this case is to test the full aerothermoelastic framework, and to investigate the assumptions made by using a steady state flow model in conjunction with a quasi-steady structural material response model. An overview of the test case is presented, along with a discussion of the results. The impact of different boundary conditions on the compliant panel, as well as the effect of varying different coupling parameters is also investigated. Finally, the computational cost of the coupled approach is examined.

This test case differs significantly from the IRV-2 case, both in terms of geometry and flight conditions. The wide range of simulation parameters illustrates the wide range of applicability for the coupling procedure presented in this thesis.

5.3.1 Background

The aerothermoelastic test case consists of a two-dimensional wedge in an hypersonic flow. A portion of the wedge surface is a compliant panel that can deform due to the aerothermal and aerodynamic loads from the external hypersonic flow. The geometry and flight conditions used were proposed by Culler and McNamara [27], who studied the test case using considerably different flow and structural models compared to the present work. Culler and McNamara used inviscid shock relations to determine the local ambient flow properties over the deforming panel based on the oblique shock that forms at the leading edge of the wedge. A third-order expansion of piston theory [8, 69] was used to compute pressures over the panel, and Eckert's reference enthalpy method [36] is used to compute the aerodynamic heating. The flow was assumed to be fully turbulent over the panel. To model the elastic response of the structure, a simply supported, semi-infinite panel was assumed, and the equation of motion was found by using von Kármán plate theory [33, 94]. A number of other modeling assumptions were made with respect to the thermo-structural model, and a full description can be found in Culler and McNamara [27].

The flow model used in this thesis represents one of the most significant differences with the methods outlined above. In particular, the results from LeMANS do not use a turbulence model, so the aerodynamic heating predictions are expected to be considerably lower than those obtained using Eckert's reference enthalpy method for a turbulent flow. The use of a full CFD model also allows for higher fidelity in modeling any flow changes that are caused by panel deformation.

A partitioned coupling method is used by Culler and McNamara, and the coupling

approach taken in this work is similar to the quasi-static approach (case C-2) presented in [27]. The use of CFD and the full structural equations in the present work greatly increase the computational cost compared to the models used by Culler and McNamara. The increased cost means that for long simulation times it is desirable to study problems with stable dynamic response so that the flow solution does not need to be updated as often. Therefore, the flight conditions that are used in this test case, and which are outlined in the following section, are chosen because they lead to a dynamically stable panel response as predicted by Culler and McNamara [27]. The computation of a dynamically stable response allows for larger time steps between flow updates, and the quasi-static update procedure is valid. As is discussed in more detail in Section 5.3.1.2, the characteristic time scale of the structural response is roughly two orders of magnitude shorter than the thermal time scale. Therefore, static structural response solutions are computed, and only the thermal response is treated in a time accurate fashion.

5.3.1.1 Geometry and Test Conditions

The geometry, shown schematically in Figure 5.9, consists of a rigid two-dimensional wedge in a hypersonic flow. A compliant panel of length $a = 1.5$ meters is located at a distance of 1.0 meter from the leading edge of the wedge. The wedge angle, θ, is 5°. The deforming panel consists of three layers of material: a radiation shield, a layer of thermal insulation, and a load-bearing plate structure. The arrangement of the layers is shown in Figure 5.9b, and Table 5.2 lists the materials used and the thickness of each layer. The radiation shield and thermal insulation together form the TPS, and the titanium plate carries the aerothermal loads. The thermal and elastic material properties are isotropic and temperature dependent, with the initial values at $T = 300.0$ K given in Table 5.3. Density is constant with respect to temperature for all three materials, and structural data is only listed for titanium since the plate

structure is the only load bearing layer. Data for the temperature dependence of the properties is found in [72] for the radiation shield and thermal insulation, and in [3] for the titanium plate. Note that in the remainder of this chapter the term "panel" refers to the complete layered material, and "plate" refers to only the load bearing structure.

(a) Schematic of geometry   (b) Schematic of plate layers
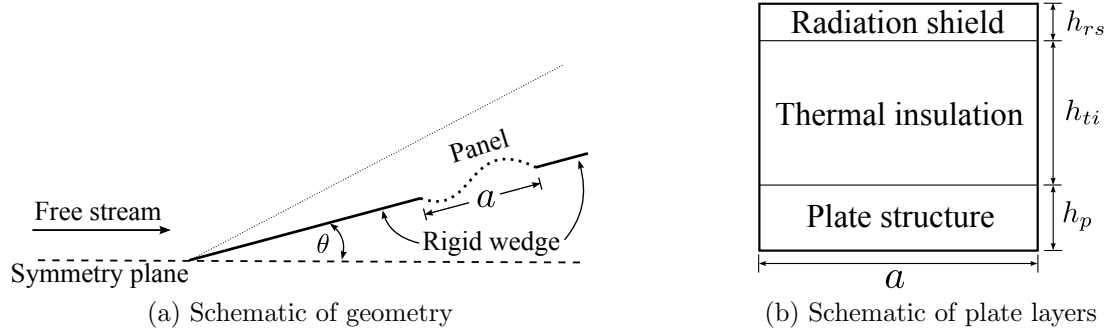
Figure 5.9: Geometry for the aerothermoelastic plate test case.

Layer	Material	Thickness [mm]
Radiation shield	PM-2000	2.0
Thermal insulation	internal multiscreen insulation (IMI)	10.0
Plate structure	titanium (Ti-6Al-2Sn-4Zr-2Mo)	5.0

Table 5.2: Plate materials and thicknesses for each panel layer.

Material	ρ $[kg/m^3]$	c $[J/Kg-K]$	κ $[W/m-K]$	Emissivity	E [GPa]	α $[1/K \times 10^{-6}]$	ν
PM-2000	1010.0	464.7	0.250	0.7	–	–	–
IMI	73.0	729.3	0.0258	–	–	–	–
titanium	4540.0	456.8	6.954	–	113.9	2.159	0.32

Table 5.3: Material properties for each plate layer at $T = 300.0$ K.

In keeping with Culler and McNamara, it is assumed that the radiation shield and thermal insulation do not carry any load, and so only the titanium plate is considered

109

in the structural material response code. The thermal response, however, is computed for all three layers of the panel structure. Flow field pressures are applied directly at the plate surface, which assumes that the normal vectors are the same at the titanium plate as they are at the flow boundary. This is an assumption, but for this particular geometry it should introduce very little error, and it is consistent with the approach in [27]. Since only the static response of the structure is considered, the mass of the thermal protection layers is neglected in the elastic calculations. The thermal protection layers are assumed to deform in exactly the same way as the surface of the titanium plate, and so to move these mesh layers the plate surface deformation is mapped directly to the TPS mesh.

A Mach 8 flow is assumed for most of the test cases that are presented. The free stream properties that are used are shown in Table 5.4, and are based on the standard atmosphere [1] at an altitude of 30km. The computational domains for the fluid and solid problems, along with the boundary conditions used, are shown in Figure 5.10.

Altitude [km]	30
M_∞	8
T_∞ [K]	226.7
ρ_∞ [kg/m^3]	1.801×10^{-2}
p_∞ [Pa]	1171.9

Table 5.4: Free stream flow properties for the aerothermoelastic plate test case.

Clamped boundary conditions are used on the edges of the plate for most of the results that are presented, rather than simply-supported as used by Culler and McNamara. The reason for using clamped boundary conditions is to prevent the edges of the deforming panel from rotating into the flow field or overlapping the fixed-wall boundaries, which could happen with a simply supported boundary condition. The use of clamped boundary conditions is expected to result in smaller panel deformations than the simply supported case. The temperature at the rigid wall boundaries is

set using a radiative equilibrium boundary condition with the boundary emissivity assumed to be 0.7, which is the same as the radiation shield emissivity. The shape of the flow domain is designed to prevent any additional shocks that may form due to panel deformation from impinging on the inlet boundaries. One end of the panel mesh is shown in Figure 5.11 with the different material layers indicated. The full panel mesh has 3168 elements with 594 cells in the radiation shield, 1782 cells in the thermal insulation, and 792 cells in the plate structure.

1. Supersonic inlet
2. Supersonic outlet
3. Symmetry
4. Coupled, no-slip wall
5. Radiative equilibrium, no-slip wall
6. Aerodynamic heating
7. Adiabatic / specified displacement (clamped)
8. Adiabatic / specified traction
9. Specified traction
10. Adiabatic
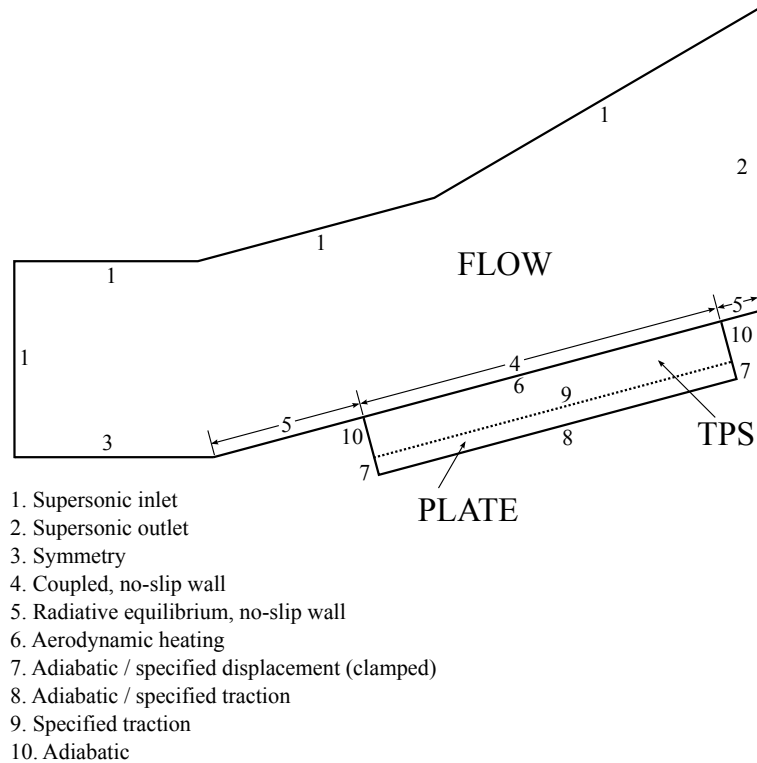
Figure 5.10: Schematic of the computational domain and boundary conditions.

The total simulation time is 900.0 s. This refers to the total length of time over which the thermal problem is allowed to evolve. A constant time step of $\Delta t = 1.0$ s is used for the thermal solutions. The pressure along the back of the panel varies along the panel length as shown in Figure 5.12. This is an approximation of the initial pressure distribution across the top surface of the panel that is computed by running an uncoupled flow simulation. This back pressure is chosen so that the pressure
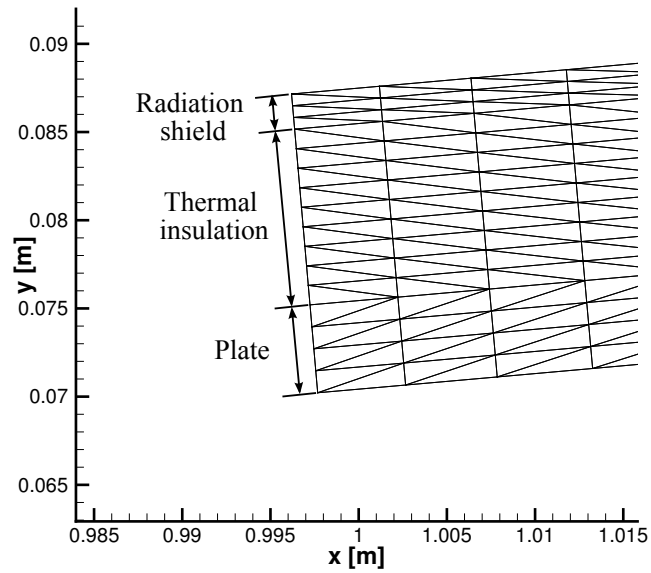
Figure 5.11: Panel mesh showing the different material layers.
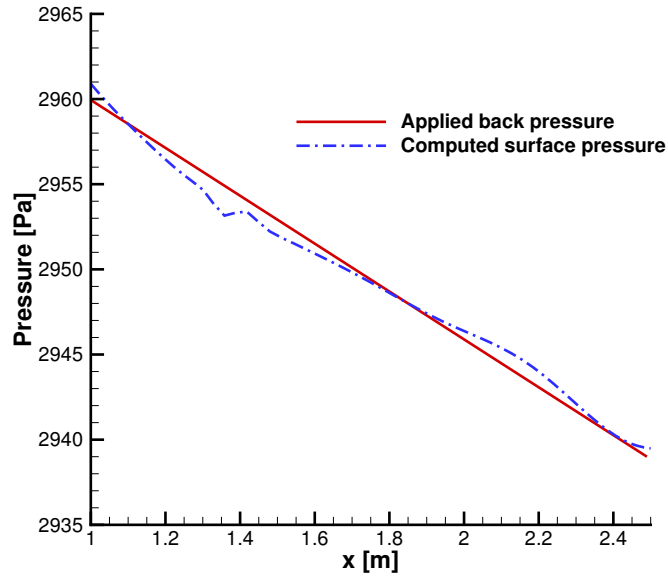
Figure 5.12: Panel back pressure and initial surface pressure.

differential across the panel is approximately zero at the start of the simulation.

The coupling procedure for this series of test cases uses a dynamic coupling pro-

cedure, rather than the fixed iteration coupling used in the IRV-2 case, to determine
when the material response code is called from LeMANS. The material response code
is called when the maximum change in the heat flux over the length of the panel
between flow iterations drops below 5.0×10^{-3} %. The code moves to the next tra-
jectory point when the change in the L_2 norm of the panel heat flux values between
calls to the material response code drops below 1.0 %. More details on the coupling
procedure are found in Section 3.4.

5.3.1.2 Characteristic Time Scales

The quasi-static aerothermoelastic coupling procedure that is described in Chapter
III relies on the separation of time scales between the fluid, structural, and thermal
components of a problem. It is assumed that the characteristic time scale of the
fluid is much shorter than the characteristic time scale of the structural response,
so that the fluid computations can be treated as steady compared to the structural
computations. In the same way, for this test case, the structural time scale is assumed
to be much shorter than the thermal time scale, so the structural response can be
treated as steady compared to the thermal response. The end result is that only
the thermal problem is solved in a time accurate fashion, while both the fluid and
structural problems are treated as steady state.

Equations for the characteristic fluid, structural, and thermal time scales are
shown in Equations 5.1, 5.2, and 5.3 [28].

$$t_f = \frac{L}{U_\infty} \tag{5.1}$$

$$t_s = \frac{1}{f_1} \tag{5.2}$$

$$t_{th} = \frac{\rho c h_p^2}{\kappa} \tag{5.3}$$

For the fluid time scale, L is the length of the panel, and U is the magnitude of the free stream velocity. The structural response time is based on the lowest natural frequency, f_1, of the structure. Given the geometry in Section 5.3.1.1, and assuming clamped boundary conditions on the plate, f_1 can be found from Equation 5.4 [68].

$$f_1 = \frac{(4.73)^2}{2\pi L^2} \sqrt{\frac{EI}{\rho A}} \tag{5.4}$$

The thermal time scale is found by assuming a Fourier number equal to 1.0 [94, 28], which allows the time scale to be determined by the ratio of a characteristic length scale squared to the thermal diffusivity.

Using the above definitions and the material properties for each material at $T = 300.0$ K, the time scales shown in Table 5.5 are computed. Note that the structural time scale is only computed for the titanium plate, since that is the only layer in which the structural equations are solved. The ratio of the thermal and structural time scales in the titanium plate is the relevant parameter in deciding whether or not a quasi-static thermal-structural response is valid. If B, the ratio of time scales given in Equation 5.5, is much greater than 1, then it is generally valid to treat the structural response as static and the thermal response as transient [17, 94].

t_{th} (PM-2000)	7.5 s
t_{th} (IMI)	206.4 s
t_{th} (Ti)	7.5 s
t_s	8.7×10^{-2} s
t_f	6.2×10^{-4} s

Table 5.5: Characteristic time scales for the aerothermoelastic plate test case.

$$B = \frac{t_{th}}{t_s} \tag{5.5}$$

Looking at Table 5.5, the structural time scale of the plate is two orders of mag-

nitude shorter than the thermal time scale, so $B >> 1$, and a quasi-static thermal-structural response is valid. Similarly, the flow time scale is two orders of magnitude shorter than the structural response time, so steady-state flow calculations are valid.

5.3.2 Flow Solution

To ensure that the initial uncoupled surface properties are sufficiently resolved, a grid convergence study is performed for the uncoupled case. Only the flow is modeled, and the flow conditions are the same as for the full coupled case. Three different meshes are used in the grid convergence study with 13640 cells, 27156 cells, and 54250 cells each. In addition to the number of cells, the initial cell length at the wall is varied from 5.0×10^{-7} m in the coarsest mesh to 1.25×10^{-7} m in the most refined mesh.

Figure 5.13 shows the results of the grid convergence study. Only the heat flux and pressure values are considered since these are the flow quantities that are most relevant for the aerothermoelastic coupling. Figure 5.13 indicates that mesh 2 consisting of 27156 cells is sufficient to resolve the initial uncoupled surface properties, and Figure 5.14 shows this mesh.

Figure 5.15 shows solution contours for the flow-only case using the 27156 cell grid. As can be seen, a relatively weak oblique shock forms on the wedge, but the highest flow temperatures are located very near the wall boundary. This is due to the large amount of viscous dissipation that occurs near the wall boundary as a result of the very high flow velocity. It is also the result of using a radiative equilibrium boundary condition at the wall.

Figure 5.16 shows the wall boundary values for the uncoupled case. The position of the panel is superimposed on the plots, and it can be seen that the highest temperature, pressure, and heat flux are located near the leading edge of the panel. The wall temperature profile is determined from the radiative equilibrium boundary

condition, and so it follows the same trend as the heat flux profile.
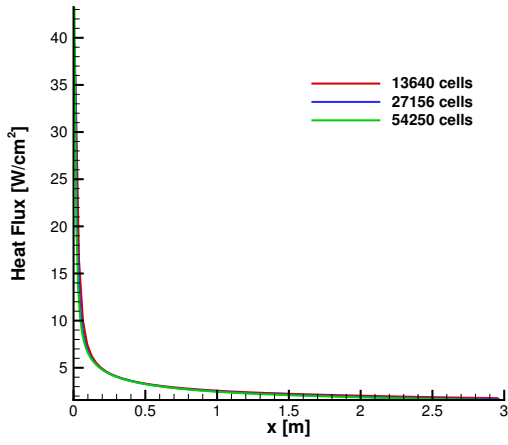
(a) Heat Flux
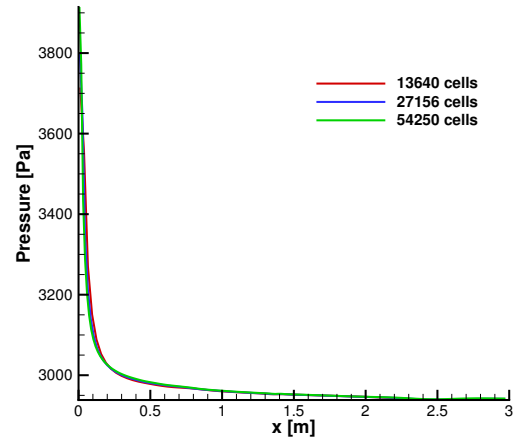
(b) Pressure
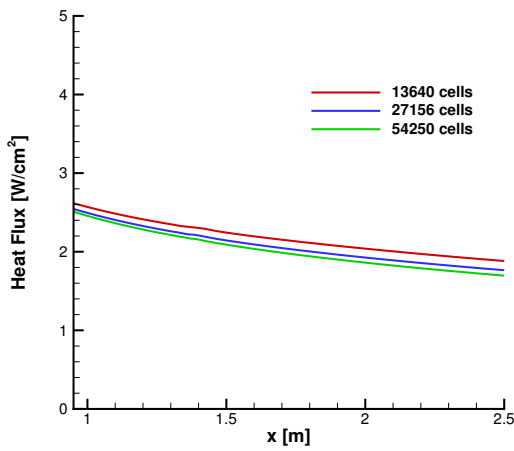
(c) Heat Flux over the panel region
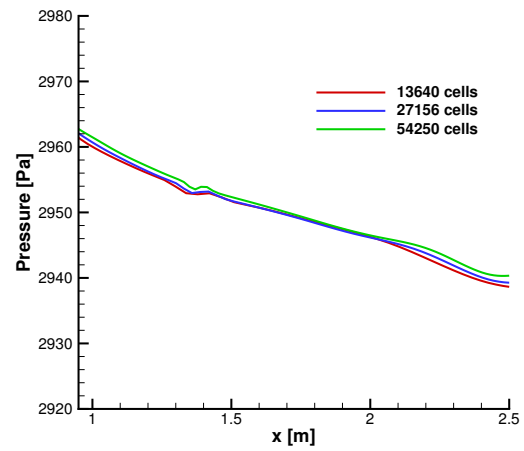
(d) Pressure over the panel region

Figure 5.13: Wall values for three different mesh refinements for the uncoupled flow case.
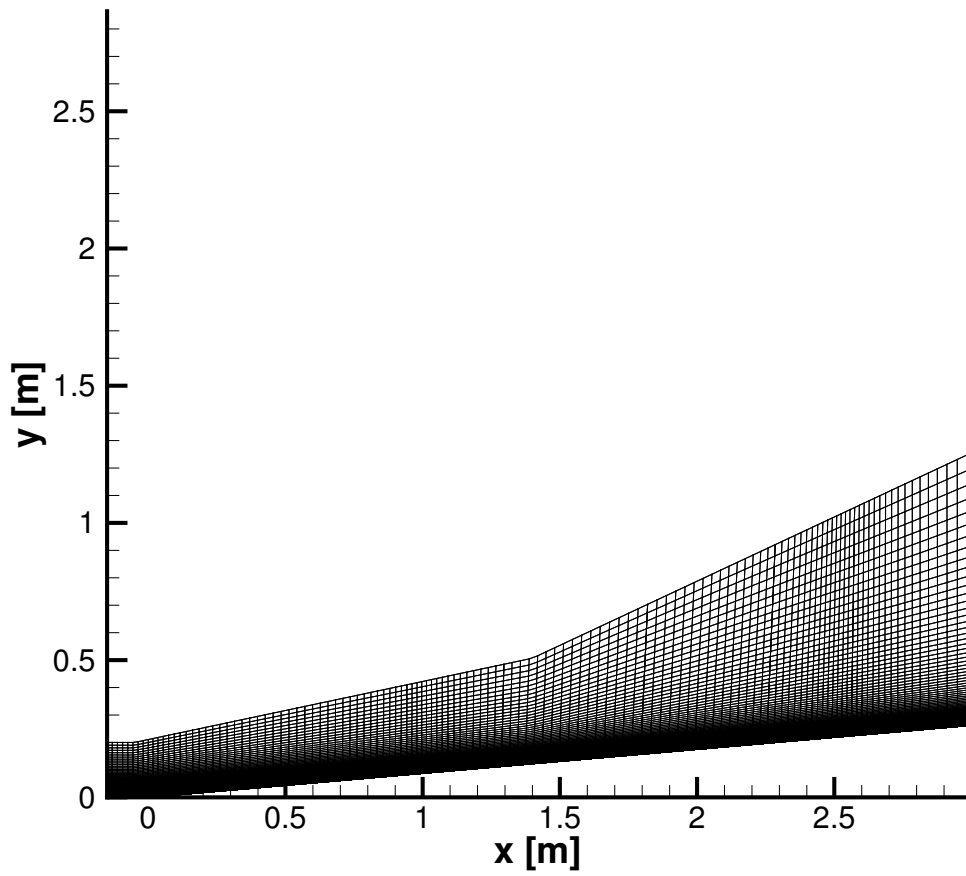
Figure 5.14: Flow mesh used for the aerothermoelastic test case.

(a) ρ_{N_2}
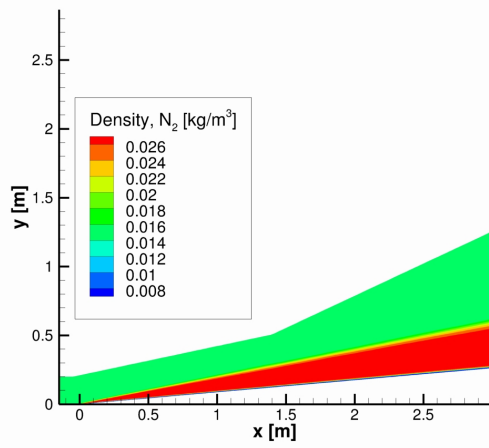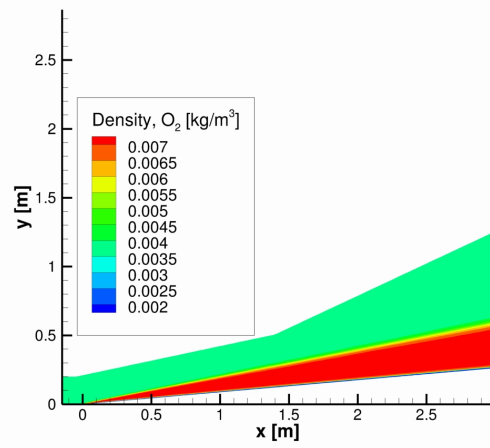
(b) ρ_{O_2}
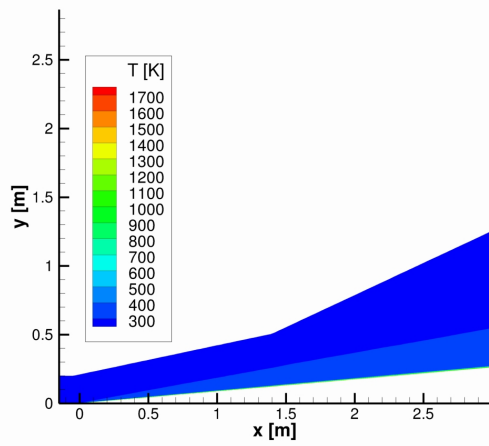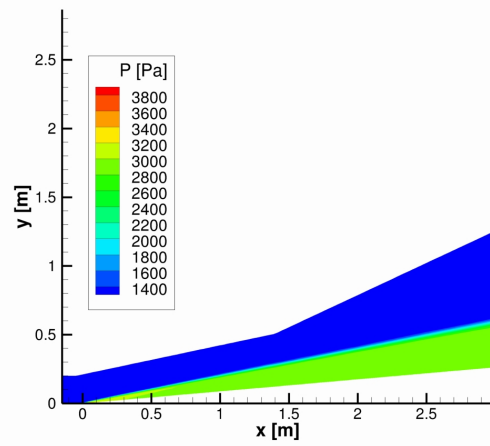
(c) Temperature

(d) Pressure

Figure 5.15: Flow contours for the uncoupled solution.
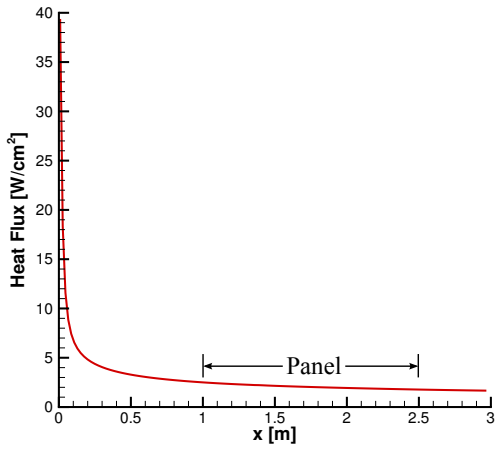
(a) Heat Flux
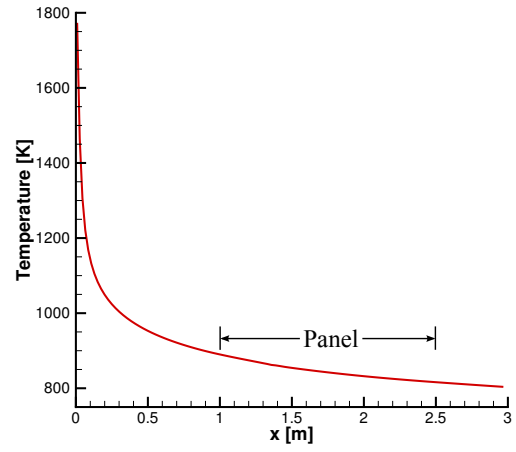
(b) Temperature
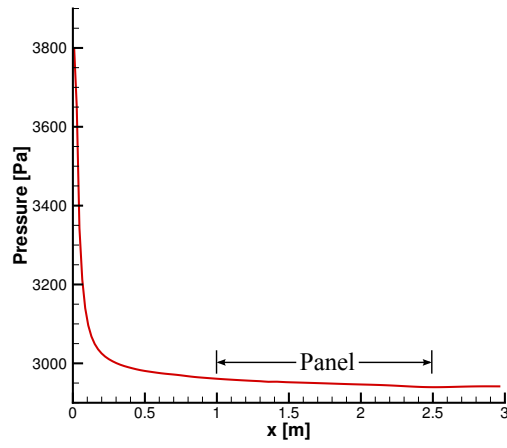
(c) Pressure

Figure 5.16: Wall values for the uncoupled flow case.

5.3.3 Mach 8 Thermochemical Equilibrium, Neglecting Pressure

As a first step towards the full aerothermoelastic test case, a case is run which uses the same geometry, flow properties, and material properties, but neglects the aerodynamic pressure force on the plate. The panel deformation is computed, but all deformation is due only to thermal effects. Using this simplified test case, the effect of the trajectory step length, t_s, on the results is investigated. The trajectory step is the length of time that the thermal material response code is run before computing the updated plate geometry and updating the flow solution. This splits a continuous simulation into a sequence of discrete "snapshots" of the coupled solution. Four different trajectory step lengths are considered: $t_s = 10.0$ s, $t_s = 5.0$ s, $t_s = 2.5$ s, and $t_s = 1.25$ s. Using a shorter step length increases the computational cost of the coupling procedure since it increases the number of discrete trajectory points at which a solution must be obtained.

Figure 5.17 shows contours of flow temperature and solid y-displacement for the longest and shortest trajectory step lengths. In both cases, the panel deformation is quite small, so there is very little impact on the flow field. The shorter trajectory step length, $t_s = 1.25$ s, leads to roughly an order of magnitude more deflection in the panel than the $t_s = 10.0$ s case. This causes a weak expansion to form over the front of the panel, and a weak shock to form over the aft portion of the panel. This is shown more clearly in Figure 5.18 where additional contour levels are included to highlight the small changes in flow properties.

The thermal surface properties at the final simulation time of $t = 900.0$ s are shown in Figure 5.19. The $t_s = 10.0$ s case shows very little change in either the temperature or heat flux profiles from the uncoupled flow-only results since there is essentially no panel deformation. The drop in temperature and corresponding increase in heat flux for the $t_s = 10.0$ s case is due to the additional heat transport into the material that is not accounted for with a radiative equilibrium boundary condition.
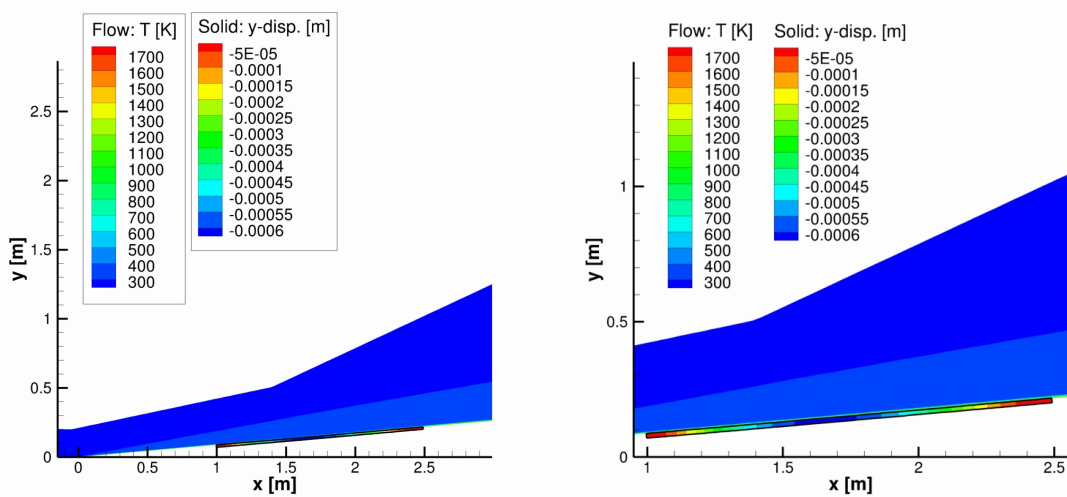
(a) Full computational domain, $t_s = 10.0$ sec.

(b) Close-up of panel region, $t_s = 10.0$ sec.
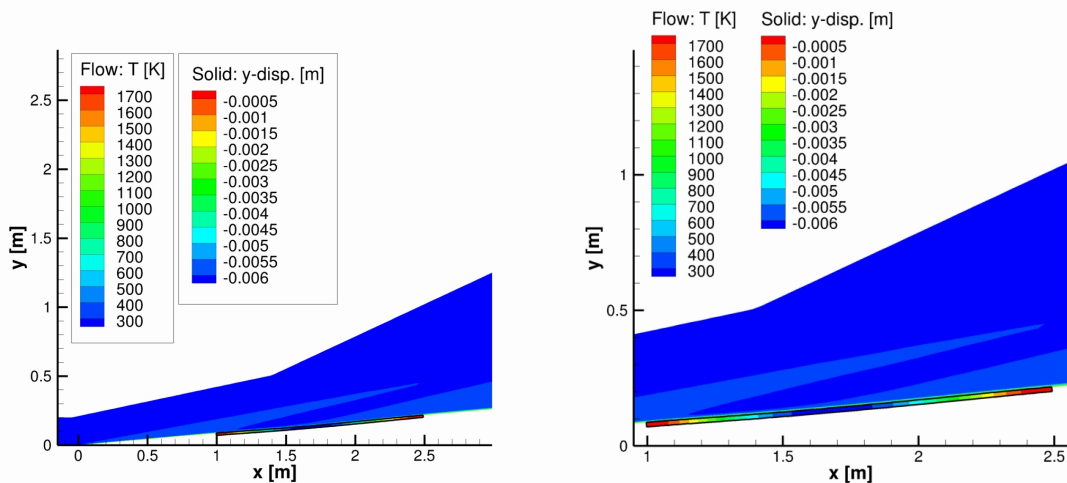
(c) Full computational domain, $t_s = 1.25$ sec.

(d) Close-up of panel region, $t_s = 1.25$ sec.

Figure 5.17: Flow temperature and solid y-displacement contours for the converged solution at $t = 900.0$ seconds neglecting pressure for two different trajectory step lengths.

As the deformation increases with shorter trajectory steps, the slight effects due to the expansion and shock which form can start to be seen. The temperature and heat flux decrease through the expansion over the front of the panel, and increase through the compression that forms over the back half of the panel.

The cause of the larger panel deflections for shorter trajectory step lengths is due
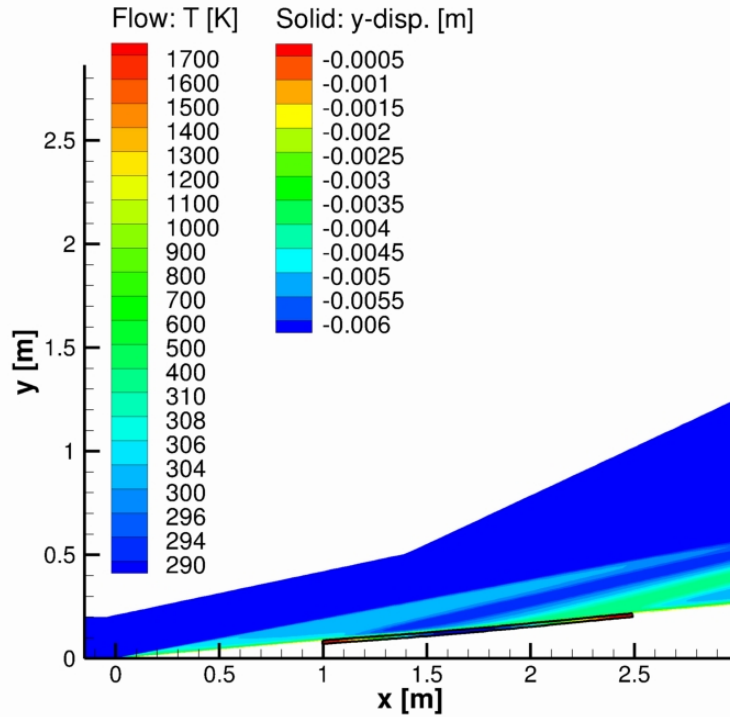
Figure 5.18: Flow temperature and solid y-displacement contours for $t_s = 1.25$ seconds using more refined contour levels.

to the higher temperatures at the titanium plate at a given time when a shorter time is used for t_s. This can be seen in Figure 5.20a. The temperature at the top surface of the plate structure for the shortest and longest trajectory lengths is shown at three different solution times. The plate temperatures are considerably lower than the panel surface temperatures due to the thermal protection layers above the plate. The in-depth profiles show the effect of non-constant heat flux across the length of the panel, and the plate temperatures are slightly higher near the leading edge of the plate. Additionally, the shorter trajectory lengths lead to slightly higher plate temperatures, and the temperature difference between the $t_s = 10.0$ s case and the $t_s = 1.25$ s case appears to be growing over time.

Figure 5.20b shows the temperatures at the top and bottom of the plate at two different simulation times and for two different trajectory step lengths. There is

(a) Temperature   (b) Heat Flux
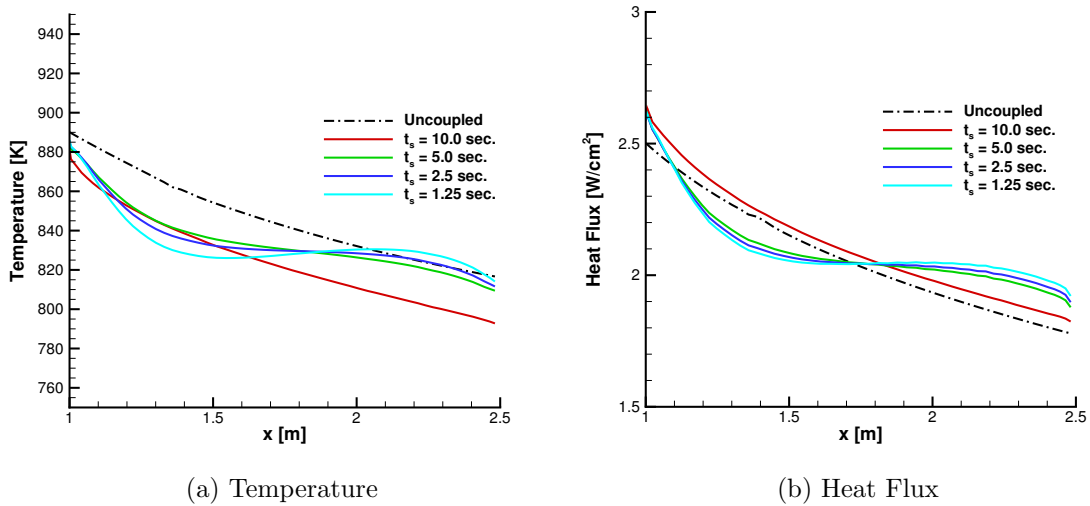
Figure 5.19: Surface properties at $t = 900.0$ seconds for various trajectory step lengths.

almost no temperature gradient through the plate thickness for the cases shown. As the simulation progresses in time, however, the temperature gradient along the length of the plate grows.

(a) Surface temperature   (b) Front and back plate temperatures
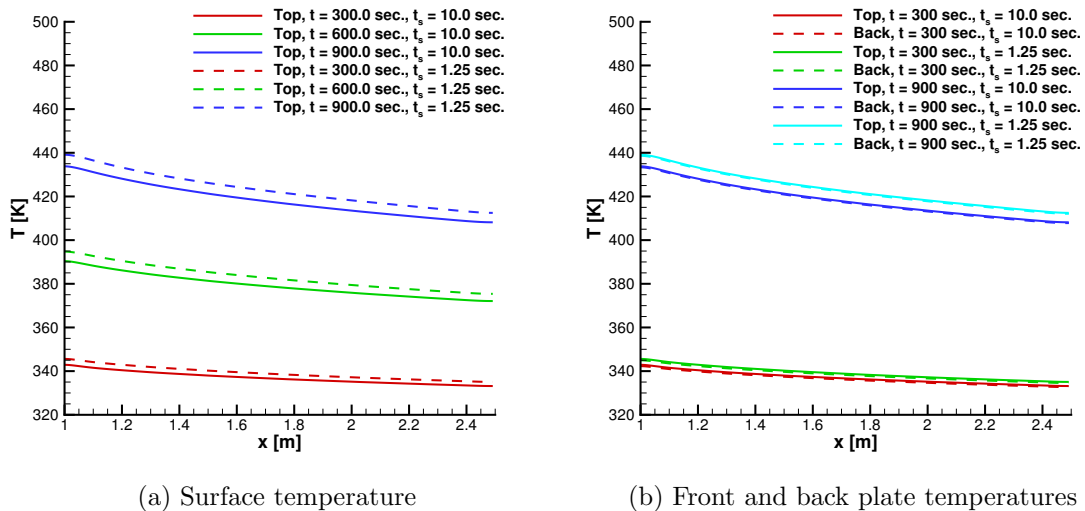
Figure 5.20: Plate temperatures for various trajectory step lengths.

(a) Young's modulus ratio   (b) Thermal expansion coefficient ratio
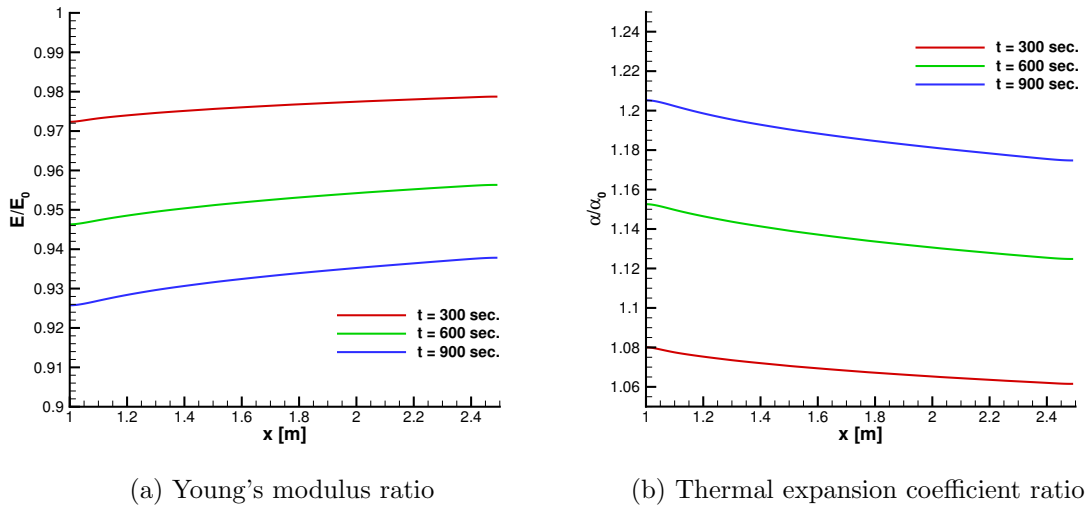
Figure 5.21: Plate material properties at three different simulation times with a trajectory step length of $t_s = 1.25$ seconds.

The effect of temperature on the material properties is shown in Figure 5.21. As expected based on the material properties, the structural stiffness decreases and the thermal expansion increases over the course of the 900.0 second test as the plate temperature increases. Additionally, there is a property gradient along the length of the plate due to the nonuniform temperature distribution. The values shown in Figure 5.21 are only for the middle of the plate, and there will also be a minor change in properties through the thickness of the plate. As shown in Figure 5.20, however, the temperature change through the plate is very small, and so the change in properties is also very small relative to the change along the plate length.

Figure 5.22 shows a comparison of the transverse panel deflection at $t = 900.0$ s normalized by the plate thickness, $h_p = 0.005$ m, for each of the trajectory step lengths as a function of the normalized distance along the plate. Due to the lack of aerodynamic forces acting on the plate, the deflection, which is caused only by thermal forces, is nearly symmetric, but biased slightly towards the front of the plate due to the higher temperatures in that region [28]. The amount of deflection increases

with decreasing trajectory step length, but the largest jump occurs from the $t_s = 10.0$ s case to the $t_s = 5.0$ s case.
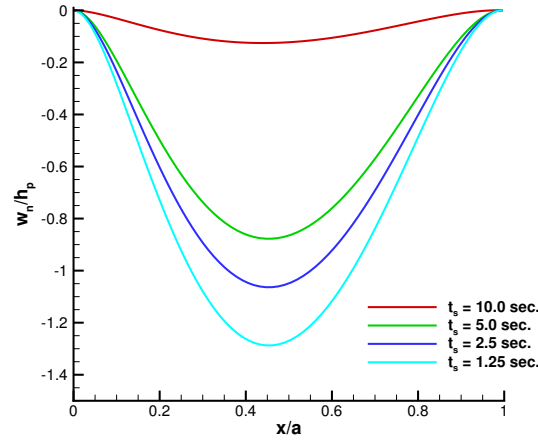
Figure 5.22: Normalized transverse panel deflection at $t = 900.0$ seconds for different trajectory step lengths.

The reason for larger panel deflections with shorter trajectory steps is that shorter trajectory steps lead to a slightly faster heating rate of the plate. This causes the plate to buckle sooner, and so the deflection at a given time after buckling is larger for the shorter trajectory steps. Figure 5.23a shows the temperature at a panel location of $x/a = 0.5$ for the wall boundary and the surface of the plate. The wall boundary heats up very rapidly, and reaches an equilibrium temperature. Due to the TPS panel layers, however, the plate heats up much more slowly, and does not reach an equilibrium temperature within the 900.0 second simulation length. So the deflections shown in Figure 5.22 do not represent final converged panel shapes, but rather they are "snapshots" of the panel shape after 900.0 seconds of heating.

Figures 5.23a and 5.23b show that the shorter trajectory step lengths maintain higher panel surface temperature and heat flux over the course of the 900.0 second simulation. The longer trajectory step lengths lead to surface temperature and heat flux values that drift slightly after the surface reaches thermal equilibrium. As the

deflection of the panel becomes large enough to impact the flow solution, the temperature and heat flux increase at the $x/a = 0.5$ panel location. As can be seen from Figure 5.22, this is due to the fact that $x/a = 0.5$ is slightly towards the compression side of the panel.

A time history of the panel deflection at $x/a = 0.5$ is shown in Figure 5.23c. This plot clearly shows the earlier start to the deformation for the shorter trajectory step lengths, and the subsequent larger deflections at $t = 900.0$s.

The shorter trajectory steps more accurately capture the transition regions in the time history of the problem. For instance, Figure 5.24 shows the heat flux at $x/a = 0.5$ for the early portion of the simulation before the equilibrium region is reached. The $t_s = 10.0$ s case does not capture the earliest portion of the heat flux curve, and it forms a fairly jagged curve as the surface properties relax to equilibrium. The $t_s = 1.25$ s case, on the other hand, captures more of the initial heat flux, and follows a much smoother curve as the solution relaxes to equilibrium. This causes the integrated heat load on the vehicle to be slightly different between the different values of t_s. These relatively small differences between the trajectory steps can grow over the course of a 900.0 s simulation since each trajectory point builds off of the solution from the previous trajectory point.

Figure 5.25 shows the panel deflections and plate temperatures for the first 200.0 s of the simulation. Results are shown for the $t_s = 10.0$ s and $t_s = 1.25$ s trajectory step lengths at three different locations along the panel. In both cases, the panel begins to deform almost as soon as the temperature rises above 300.0 K, but the deformations are very small. An interesting trend is that the front portion of the panel ($x/a = 0.25$) and the midpoint of the panel ($x/a = 0.5$) both deform away from the flow field, with the front portion deforming more rapidly. The back portion of the panel ($x/a = 0.75$), on the other hand, begins to deform into the flow field. This pattern of deformation remains fairly constant for all trajectory step lengths out to roughly 800.0 seconds,
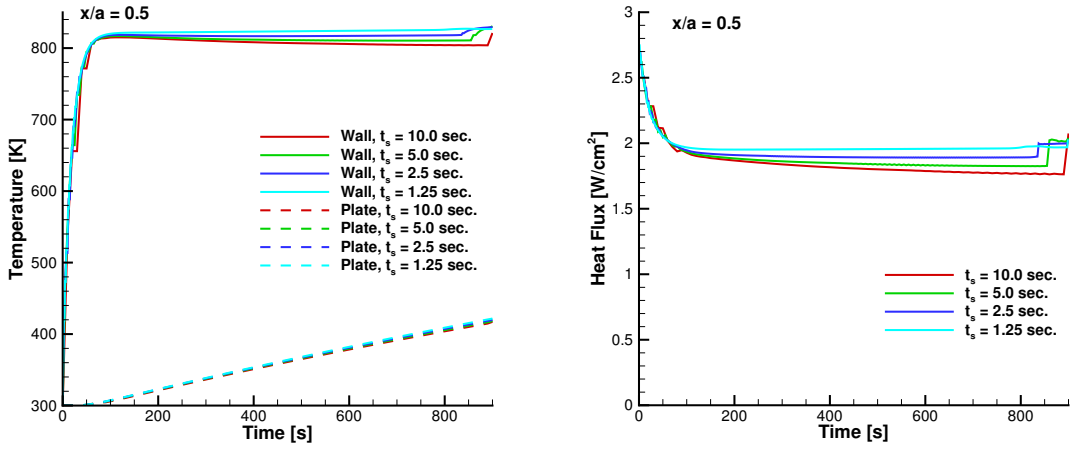
(a) Temperature

(b) Heat Flux
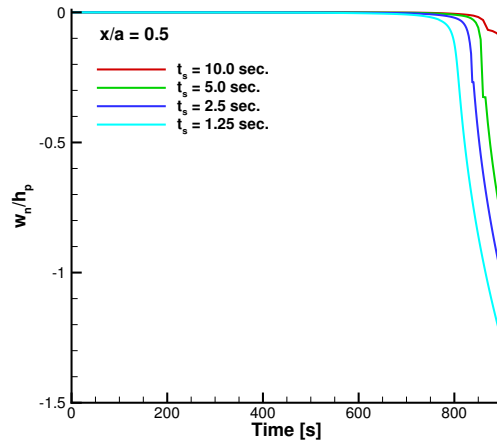
(c) Deflection

Figure 5.23: Time history of solution for various trajectory step lengths at a plate location of $x/a = 0.5$.

at which time the midpoint of the panel begins to deflect more rapidly than the front portion, and the entire panel begins to deform very rapidly away from the flow field. This behavior can be seen in Figure 5.26 which shows the time history of deflections and temperatures at three panel locations for all four trajectory step lengths.

The period of rapid panel deflection occurs at roughly the same plate temperature for all four trajectory step lengths. So, even though the deflections occur at

(a) Heat flux
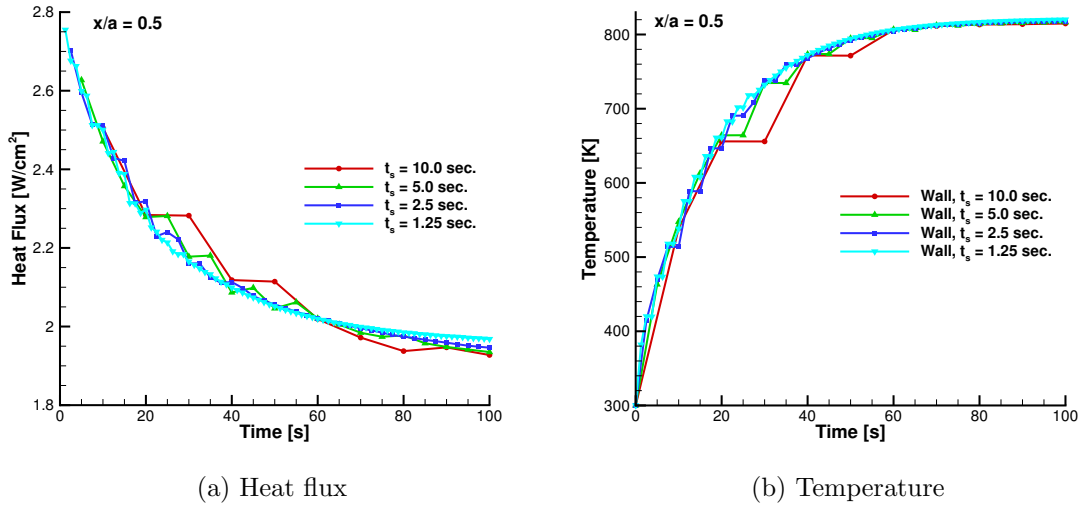
(b) Temperature

Figure 5.24: Heat flux and surface temperature at early simulation times.

(a) $t_s = 10.0$ s

(b) $t_s = 1.25$ s
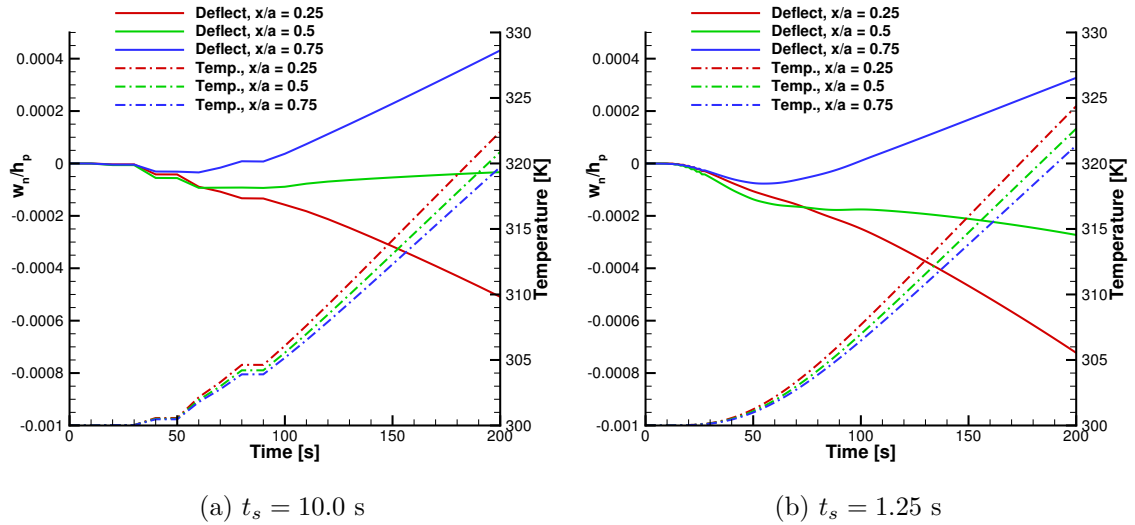
Figure 5.25: Panel deflection (solid lines) and plate temperature (dash-dot lines) at three different plate locations for two different values of t_s and for early simulation times.

different points in time for the different trajectory step lengths, they are starting at approximately the same temperature values. This seems to indicate that the sudden increase in deflection is due to a critical temperature being reached, at which point

the plate buckles due to the thermal expansion. This is consistent with the idea that the thermal loads are driving the panel response in this test case, and the deflections are qualitatively similar to those shown by Usmani [98] for beams subjected to purely thermal loading.

(a) $t_s = 10.0$ s

(b) $t_s = 5.0$ s

(c) $t_s = 2.5$ s

(d) $t_s = 1.25$ s
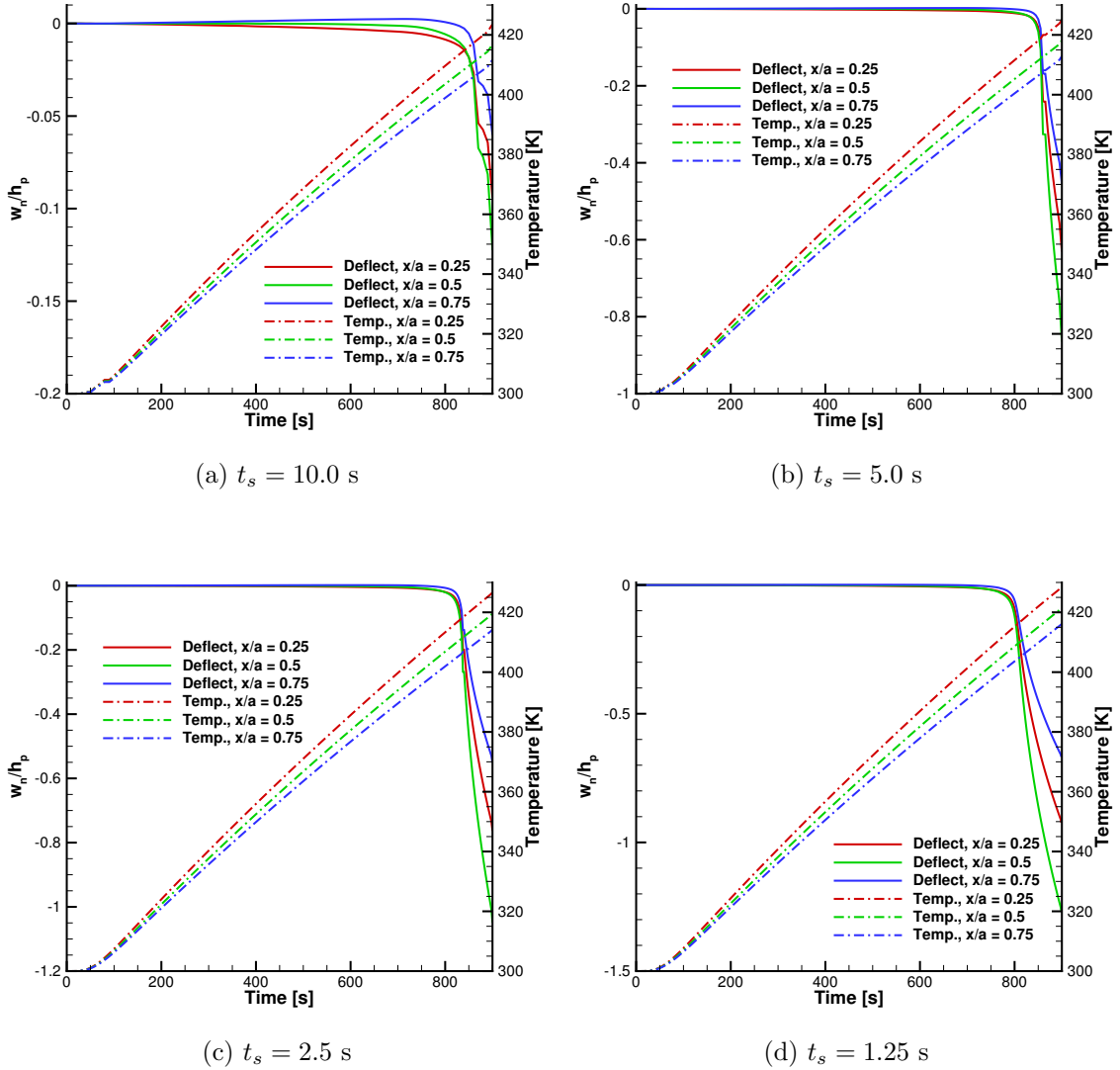
Figure 5.26: Panel deflection (solid lines) and plate temperature (dash-dot lines) at three different plate locations for two different values of t_s.

It is clear that t_s plays a significant role in the time evolution of the solution. Ideally, t_s should be refined until a converged time history is obtained. The length

of t_s that is needed to produce convergence is not clear, however, it is below the values tested in this work. As the panel temperature reaches an equilibrium value, it is expected that the impact of t_s on the final equilibrium panel shape will be much smaller than the impact on the time evolution.

5.3.4 Mach 8 Thermochemical Equilibrium, Including Pressure

Building off of the previous test case, the same geometry and flow conditions are used, but the aerodynamic loads are considered in addition to the thermal loads. Once again, a series of trajectory step lengths is investigated, but the step length now ranges from 20.0 seconds to 2.5 seconds. Based on results presented by Culler and McNamara [27], the free stream conditions listed in Table 5.4 should lead to a dynamically stable panel response, and therefore the quasi-static coupling framework presented in this work is valid.

Figure 5.27 shows the time history of panel deflection, surface pressure, temperature, and aerodynamic heat flux at a location of $x/a = 0.5$ along the length of the panel for the four different trajectory steps considered. There are several things to note about these plots. Looking first at Figure 5.27a, the different trajectory step lengths again lead to very different deflection histories. In addition, as the trajectory step length becomes shorter, the panel begins to oscillate, and the deflection magnitudes become quite large relative to the plate thickness. The peak deflection amplitudes for the $t_s = 2.5$ s case are roughly six times larger than for the corresponding case in Section 5.3.3. This oscillatory behavior indicates that the problem is no longer dynamically stable, and the quasi-static solution procedure breaks down. In fact, for the $t_s = 2.5$ s and $t_s = 5.0$ s cases it is not possible to run for the full 900 seconds because the surface properties are not able to converge sufficiently between material response calls in order to move on to the next trajectory point. The code does not complete the full 900.0 s simulation because of an oscillating convergence

130

history, not because it crashes. This is discussed in more detail in Section 5.3.4.1.

The lack of a dynamically stable response for this test case, even though it is predicted to be stable in [27], may be due to the difference in computational tools that are used. The flow model in particular differs significantly from that used by Culler and McNamara, and may be responsible for the difference in behavior. Alternatively, a different trajectory step length may lead to a stable response. However, this does not seem likely since the longer trajectory steps show signs of clipping the solution, and the shorter trajectory steps seem to enhance the instability. While the dynamic nature of the panel response cannot really be studied with the coupling procedure used in this work, it is still possible to look at the steady state snapshots of the flow that did converge, and investigate the impact of the panel deformation on the flow solution.

Looking again at Figure 5.27, it can be seen that the panel deflections are accompanied by oscillations in the surface properties as well. The pressure oscillations, shown in Figure 5.27b, are in phase with the deflections and show the effects of a large snap-through from a concave to a convex deformation, followed by a more steady increase or decrease to the maximum or minimum pressure value. The maximum and minimum pressure values lag slightly behind the maximum convex and concave deflections. The temperature at the panel surface also oscillates, however, the oscillations only roughly match up with the displacement oscillations. This is due to the longer time scales of the thermal problem. The temperature oscillations at the panel surface are not present at the plate surface within the simulation time that is considered, due to the time lag created by the TPS. The heat flux is similar to the temperature in that it does not show the same level of oscillation as the pressure and displacement profiles.

Figure 5.28 shows the flow pressure, flow temperature, and solid y-displacement contours for the $t_s = 2.5$s trajectory step length at two different times corresponding

(a) Deflection

(b) Pressure

(c) Temperature

(d) Heat Flux
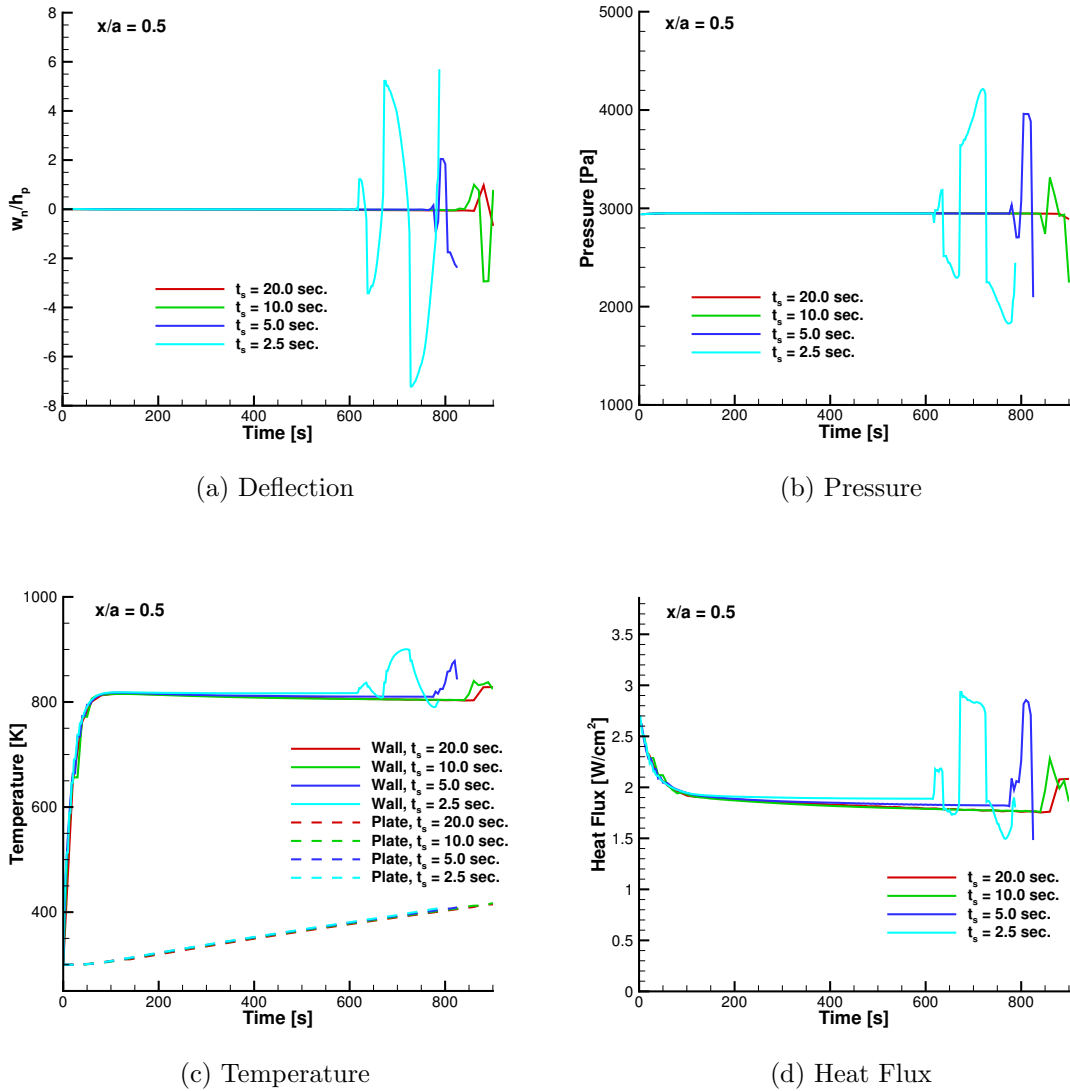
Figure 5.27: Time history of solution for 4 different trajectory step sizes.

to the largest positive and negative panel deflections determined from Figure 5.27a. The use of time values does not indicate a physically accurate period of oscillation, but only refers to the time within the thermal solver at which these deformations occurred. Contrary to the flow contours shown for the no pressure test case, these figures show considerable changes in the flow properties due to the much larger panel deformations. In the case of convex panel deflection, a shock forms over the front

portion of the panel accompanied by an expansion and another compression near the aft of the panel. The concave panel deflection leads to the inverse situation where there is an expansion over the front of the panel and a shock over the rear. For both the convex and concave deformations, the shocks that form are stronger than the initial shock generated at the leading edge of the wedge, and lead to high localized pressures and temperatures.

(a) $t = 672.5$ seconds, pressure

(b) $t = 727.5$ seconds, pressure

(c) $t = 672.5$ seconds, temperature

(d) $t = 727.5$ seconds, temperature
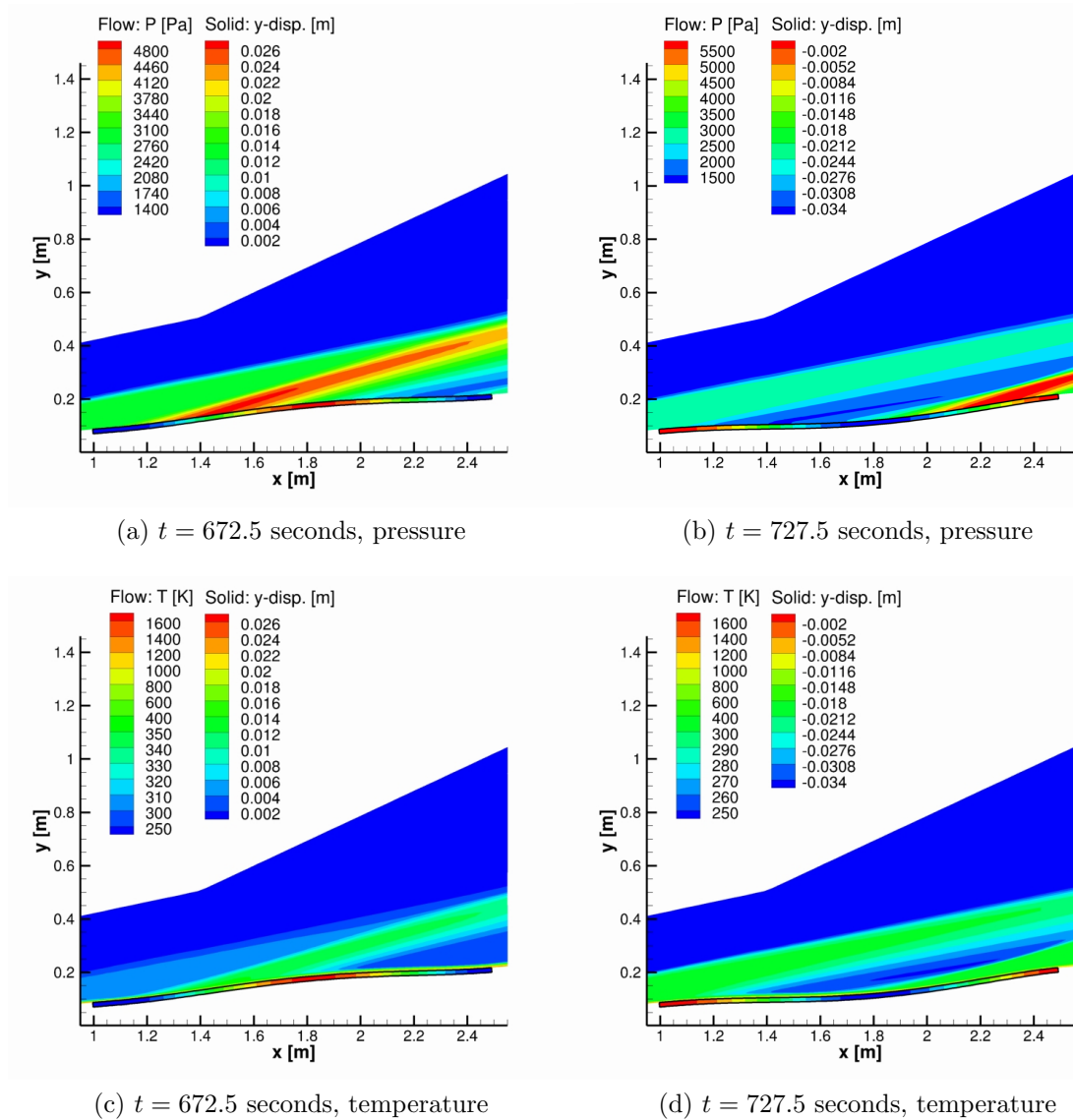
Figure 5.28: Flow and solid contours for for two times using a trajectory step size of $t_s = 2.5$ seconds.

Figure 5.29 shows the wall pressure and temperature profiles for $t = 672.5$ s and $t = 727.5$ s. The concave deflection ($t = 727.5$ s) shows higher maximum pressures and temperatures than the convex deflection case. This is due to the stronger compression that forms over the aft portion of the panel, which is a result of a larger magnitude panel deflection.

(a) Pressure   (b) Temperature
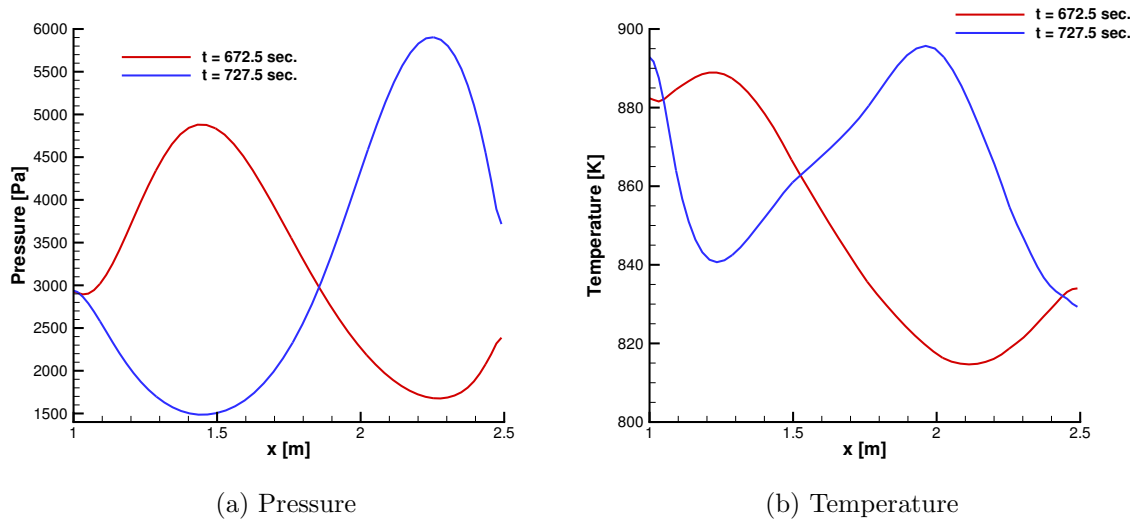
Figure 5.29: Surface properties at two different solution times for the $t_s = 2.5$ s test case.

Figure 5.30 shows the normalized deflection of the panel for some of the different trajectory step lengths. Figure 5.32a shows the deflection for the $t_s = 20.0$ s and $t_s = 10.0$ s cases at $t = 900.0$ seconds. These are the only two cases that ran for the full 900.0 seconds, however, they still show signs of dynamic instability. Similar to the no pressure case, different trajectory step lengths lead to the panel starting to deform at different times. In the case of a dynamic response, this leads to deflection profiles for the same simulation time that look very different since they represent different instants in time during a dynamic response.

The second figure shows the maximum positive and negative panel deflections for the $t_s = 2.5$ s second case. These correspond to the flow contours shown in Figure

(a) $t = 900.0$ seconds

(b) $t_s = 2.5$ seconds
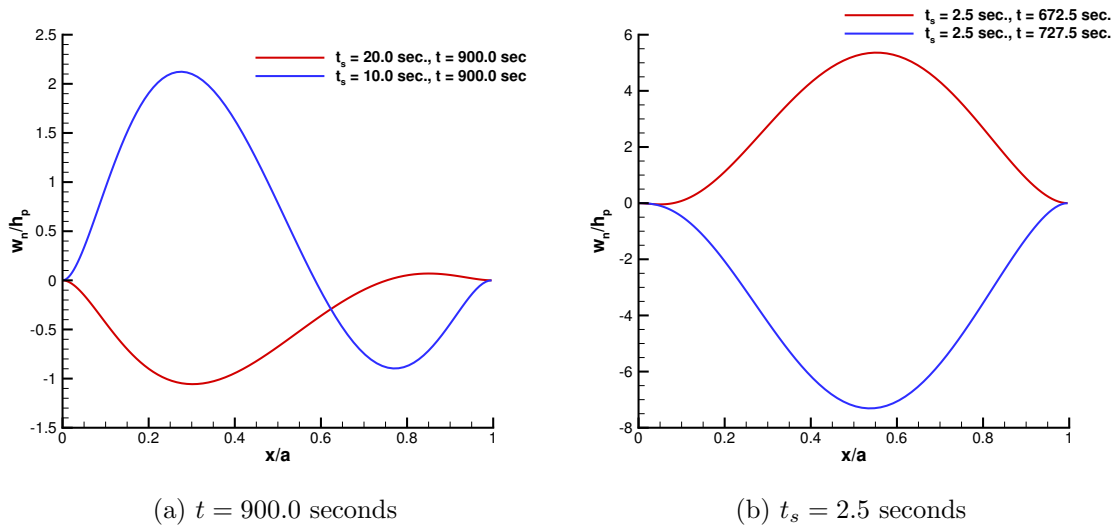
Figure 5.30: Normalized deflections for two different times using multiple trajectory step lengths.

5.28. The deflections are not symmetric about the zero displacement line, and the magnitude of the displacement is greater in the negative direction. The location of maximum displacement is slightly aft of the midpoint of the panel. This is in contrast to the no pressure case where the maximum displacement was slightly forward of the midpoint. The change is due to the addition of the pressure forces on the plate which act to push the plate aft.

The temperature distribution at the middle of the plate at several different times is shown in Figure 5.31. Similar to the no pressure case, shorter trajectory lengths lead to slightly higher in-depth temperatures at a given time. Also, although it is not shown, the temperature gradient through the panel is once again very small, so the plate surface temperatures are approximately the same as the mid-plate temperatures. The temperature effects due to panel oscillation have not penetrated to the depth of the plate, so the temperature profiles look very similar to the profiles from the no pressure case shown in Figure 5.20.

Figure 5.32 shows a comparison between the pressure and viscous stresses for the

135
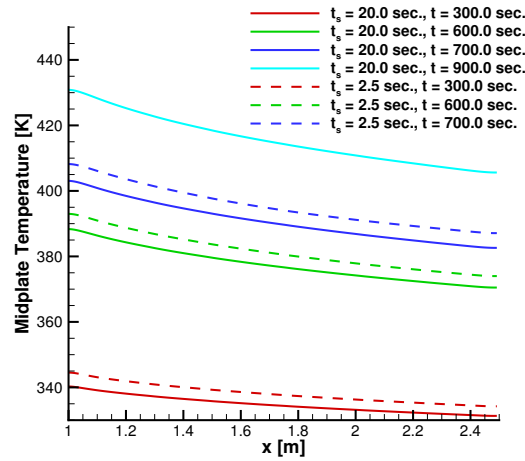
Figure 5.31: Temperature at the middle of the plate for several different solution times and two different trajectory step lengths.
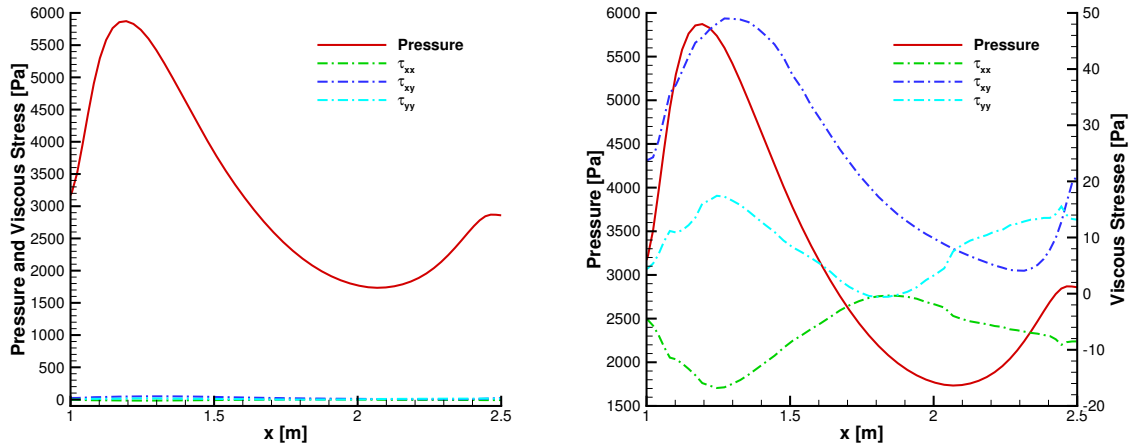
(a) Pressure and viscous stress, same axes

(b) Pressure and viscous stress, different axes

Figure 5.32: Comparison of the pressure and viscous stresses for the $t_s = 2.5$ s case at $t = 785.0$ s.

convex deformation case with $t_s = 2.5$ s. The pressure force on the plate, even when there is a strong expansion, is considerably larger than the viscous forces. In fact, for this test case, the viscous forces are completely negligible.

5.3.4.1 Effect of Plate Boundary Conditions

The previous results all utilize a clamped boundary condition on the edges of the plate. Use of a simply-supported boundary condition should result in larger total deformations since the ends of the plate are free to rotate. In order to test the impact of the plate boundary conditions on the panel response, the previous test case is modified to use a simply-supported boundary condition at the midpoints of the plate ends. For this test, only a trajectory step of 10.0 seconds is used since this trajectory step length ran for the full 900.0 s in the clamped boundary condition case.

Figure 5.33a shows the time history of deflection for the simply-supported case. For the clamped case it was found that a trajectory step of 10.0 s became unstable at approximately $t = 800.0$ s, but would converge for the full 900.0 s simulation. In the simply-supported case, however, after the first large deflection at roughly $t = 400.0$ s, the surface properties would not converge between calls to the material response code. The deformations obtained are much larger than the clamped $t_s = 10.0$ s case, but still smaller than those observed in the clamped $t_s = 2.5$ s case. The instability also starts much sooner in time for the simply-supported case than for any of the clamped cases.

Figure 5.33b shows the normalized deflection as a function of the coupling iteration number rather than time. The coupling iteration number refers to the number of calls made to the material response code from LeMANS. Multiple coupling iterations typically occur within a single trajectory step in order to converge the surface properties during that step. The iterations that occur during a trajectory step at a simulation time of $t = 450.0$s are highlighted in the figure. It is interesting to note that the deflections at the iterations within a single time step show bounded, nearly symmetric, non-periodic oscillations. Similar behavior is also present in the clamped boundary condition case, but it is much more pronounced with the simply-supported boundary conditions.

(a) Deflection vs. time   (b) Deflection vs. coupling iteration
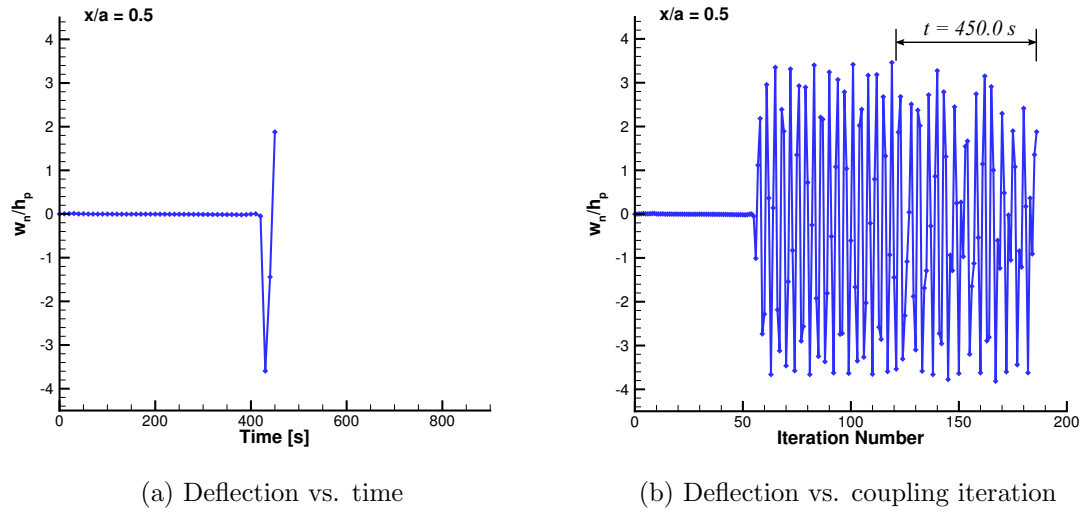
Figure 5.33: Deflection as a function of time and coupling iteration number for a simply supported boundary condition.

Figure 5.34 shows the panel shapes obtained at 10 consecutive iterations within the $t = 450.0$s time step. Similar panel shapes are observed by Dowell [33] and Dugundji [35] using plate theory and a first-order aerodynamic approximation. The results shown in [33] and [35] are time resolved, however, and so the sequence of panel shapes does not match what is shown in Figure 5.34. Due to the lack of time accuracy in the computed structural response, and since the trajectory steps are considerably longer than the time scale of the dynamic response, the panel shapes that are computed show different instantaneous panel deflections within the entire range of panel oscillation, and not a continuous series of deflections in time. If the trajectory step length is shortened to capture the dynamic response time scale, and the panel acceleration is accounted for in the structural response, then it should be possible to resolve the oscillations accurately using the present approach, however, it is computationally prohibitive for this test case.
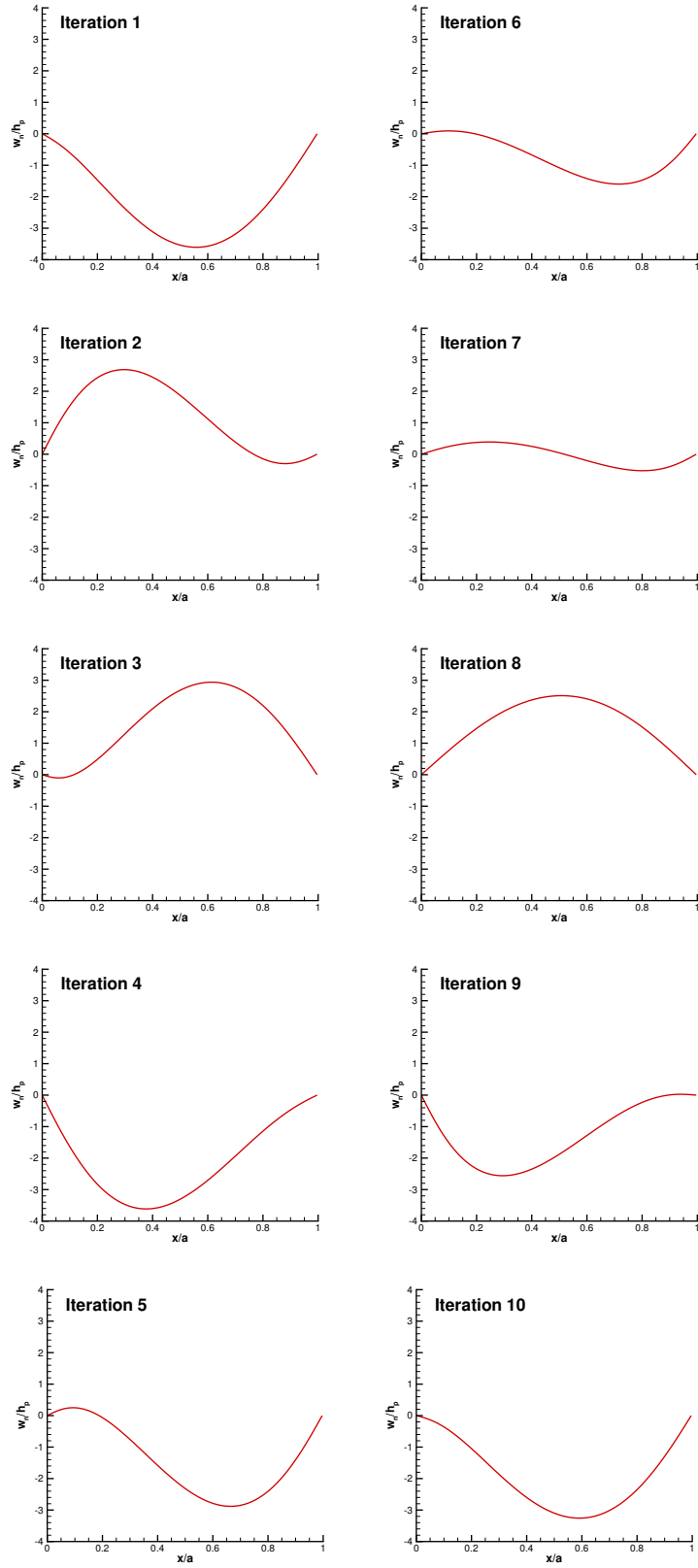
Figure 5.34: Series of panel deflections at $t = 450.0$ seconds.

5.3.5 Mach 7 Thermochemical Equilibrium, Effect of Plate Back Pressure

Based on analysis shown in [27], the time-to-flutter for this panel test case increases as the flight Mach number decreases. This is due to lower levels of thermal and aerodynamic loading. Therefore, a test case using the same panel geometry and free stream flow conditions is run at a Mach number of 7, to investigate whether the lower heating and aerodynamic force levels will lead to a dynamically stable response at $t = 900.0$ s.

An additional comparison is made using two different back-plate pressure distributions. The initial back pressure is set using the same approach as in the Mach 8 case, where a linear pressure variation is assumed with the values taken from the Mach 7 flow-only solution. This pressure distribution is then increased uniformly by 10% to produce a higher back pressure along the panel. As discussed in Section 5.3.1.1, the actual magnitude of the back pressure for this test geometry, as well as the pressure distribution along the back of the panel is not known. Two different trajectory step lengths are used: $t_s = 10.0$ s and $t_s = 5.0$ s.

Figure 5.35 shows the time histories of deflection and temperature at $x/a = 0.5$ for all of the cases considered. The original pressure cases do not show any panel deflection until almost 900.0 seconds, and the panel surface temperature quickly reaches an equilibrium temperature that is nearly constant for most of the simulation. The equilibrium temperature for this case is slightly lower than for the Mach 8 case since the aerothermal load is lower. All of this is consistent with lower Mach numbers delaying the panel dynamic response.

When the back pressure is increased by 10%, the results change considerably. For both trajectory step lengths, the panel begins to deform immediately due to the pressure difference across the panel. The $t_s = 5.0$ s case has a more rapid rate of deformation, but both cases reach approximately the same maximum stable deformation. At roughly $t = 800.0$ s, the $t_s = 5.0$ s case becomes unstable, and the panel

140

(a) Deflection   (b) Temperature
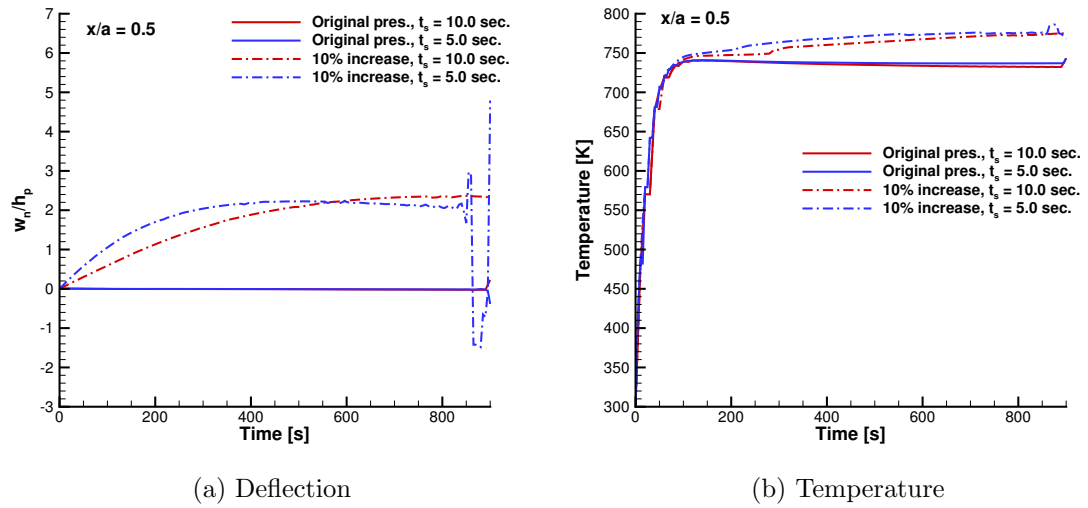
Figure 5.35: Time history of deflection and temperature for the Mach 7 test case with different plate back pressures.

begins to oscillate around the stable deformed position [33]. The $t_s = 10.0$ s case, on the other hand, remains stable over the entire 900.0 second simulation, with the panel deformation nearly constant after 600.0 seconds.
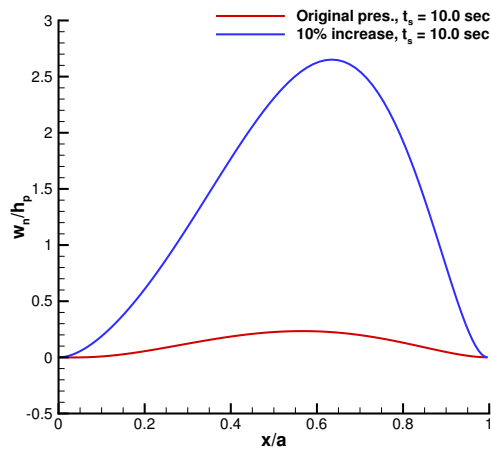
Figure 5.36: Normalized deflections at $t = 900.0$s for the Mach 7 test case with different plate back pressures.

Figure 5.36 shows the normalized deflections for the two different plate back pres-

sure cases at the $t_s = 10.0$ s trajectory step length. The increased back pressure case shows much larger deflections, which is mainly due to the pressure imbalance across the plate at the initial time. Both of the plotted deflections show the peak deflection occurring aft of the plate midpoint.

These results indicate that decreasing the aerodynamic and aerothermal loads on the panel by using a lower free stream velocity does delay the onset of unstable behavior. This is consistent with Culler and McNamara's results. Additionally, a moderate increase of the plate back pressure causes significant changes in the deformation of the plate. At higher back pressures the plate shows either a stable response over the 900.0 second simulation, or an unstable response when a shorter trajectory step length is used. The considerable changes that occur for different back pressures indicate that appropriate determination of the back pressure is very important in determining the panel response.

5.3.6   Effect of Nonequilibrium

The cases presented up to this point do not allow for the possibility of thermal or chemical nonequilibrium. To assess the potential impact of nonequilibrium on the panel response, the Mach 8 case from Section 5.3.4 with clamped boundary conditions is rerun, but with the vibrational energy conservation equation included, and allowing for chemical reactions to take place. The only reactions that could potentially take place within the 5 species air model that is used are the dissociation of N_2 and O_2, and the subsequent formation of NO. Two different trajectory step lengths, $t_s = 10.0$ s and $t_s = 2.5$ s, are used.

Figure 5.37 shows the time histories of the panel deflection, surface pressure, surface temperature, and heat flux for the cases allowing for nonequilibrium and for the equilibrium results from Section 5.3.4. For both of the trajectory step lengths that are considered the results from the equilibrium and nonequilibrium solutions are

very similar. The only difference is that the nonequilibrium results become unstable at a slightly earlier time, so the dynamic response is shifted slightly earlier in time. This offset is less evident for the $t_s = 10.0$ s case than it is for the $t_s = 2.5$ s case.

Looking at the translational and vibrational heat fluxes in Figure 5.38 for the $t_s = 10.0$ s case, it can be seen that the heat flux due to vibrational energy is over 2 orders of magnitude lower than the translational energy heat flux. This indicates that thermal nonequilibrium is a very minor factor for this test condition.

Figure 5.39 shows vibrational temperature contours for the maximum positive and negative panel deflections. In the convex deformation case, there is no discernible change in vibrational temperature even through the shock. In the concave case there is a very slight increase in temperature after the shock, but it's only about a degree above the free stream temperature. As was shown in Section 5.3.4, the shock in the concave case is stronger than the shock caused by convex deformation, and while this leads to only a very small change in Figure 5.39, it is conceivable that for large enough deformations, or for similar deformations at higher speeds, the effects of nonequilibrium could become more important.

Similar results are obtained for the flow chemistry. Flow temperatures are too low for the dissociation of N_2 and O_2, and so the flow remains in chemical equilibrium.
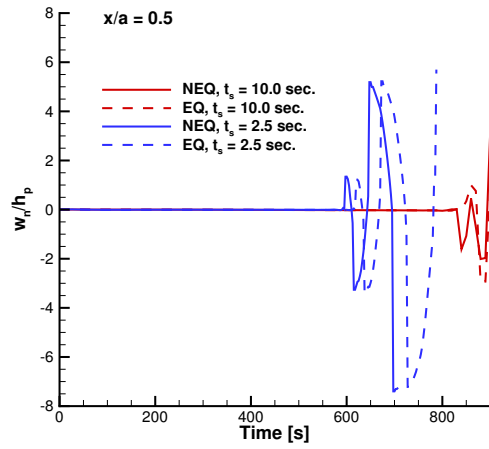
(a) Deflection
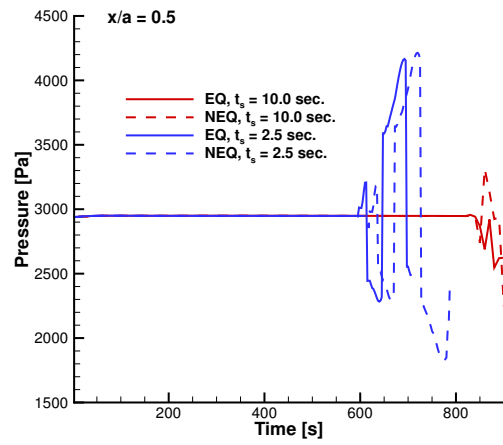
(b) Pressure
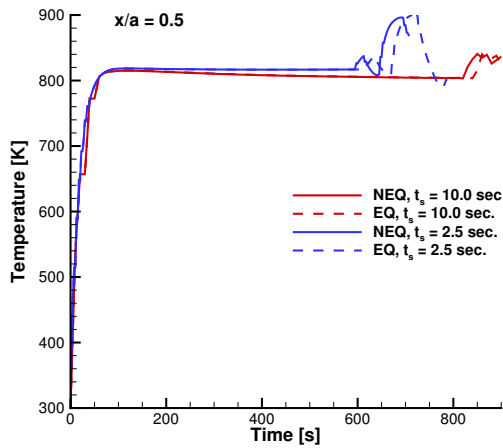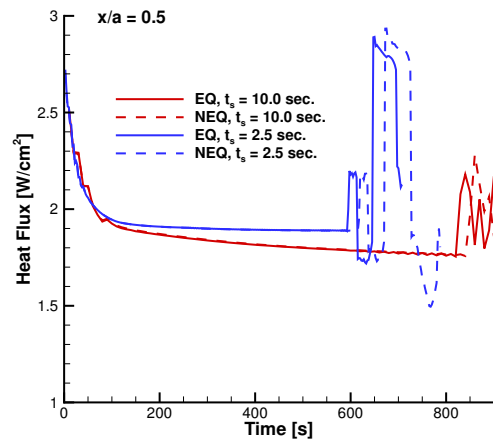
(c) Temperature

(d) Heat Flux

Figure 5.37: Comparison of time histories for equilibrium (EQ) and nonequilibrium (NEQ) flow models.
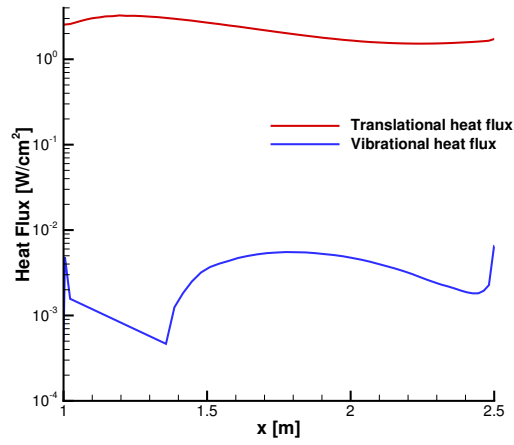
Figure 5.38: Heat flux at $t = 900.0$ seconds for the $t_s = 10.0$ s case.
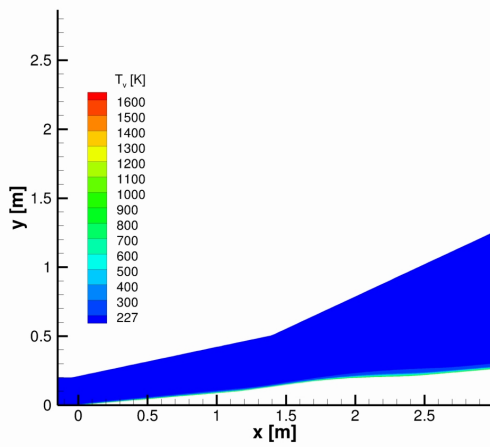
(a) $t = 650.0$ seconds
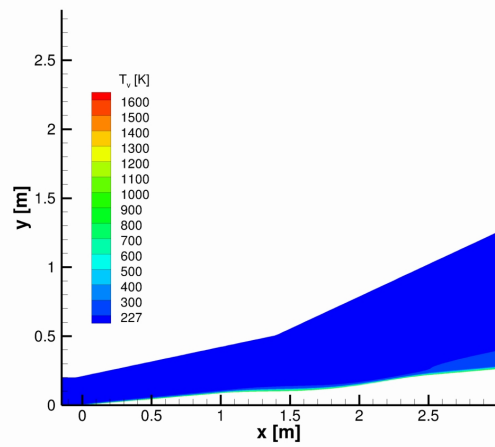
(b) $t = 700.0$ seconds

Figure 5.39: Vibrational temperature contours at maximum deflection points of the $t_s = 2.5$ s case.

145

5.3.7 Computational Cost

An important consideration with any form of computational modeling is the time that it takes to produce a solution. Table 5.6 shows the total time required for the Mach 8 equilibrium and nonequilibrium cases from sections 5.3.3, 5.3.4 and 5.3.6, as well as the percentage of time spent on flow calculations versus material response calculations. The total time listed is the CPU time, not the wall time. Most of the results presented make use of 16 processors for the flow computations, but only a single processor for the material response calculations. In order to present a fair comparison, times are shown only for the cases that completed the full 900.0 second simulation to a similar level of convergence.

Case	t_s [s]	Material Response Calls	CPU Time [hr]	% Flow	% Material Response
No Pres.	10.0	106	342.1	98.73	1.27
	5.0	198	700.2	97.73	2.27
	2.5	377	806.7	97.38	2.62
	1.25	738	1128.4	96.63	3.37
EQ Pres.	20.0	58	509.4	97.72	2.28
	10.0	122	563.1	96.97	3.03
NEQ Pres.	10.0	115	1242.9	98.50	1.50

Table 5.6: Computational times for aerothermoelastic test cases.

The CPU time increases as t_s decreases, but the percentage of time spent in the flow and material response codes changes very little between the different trajectory step lengths. In general, there is a slight increase in the percentage of time spent in the material response code as shorter trajectory steps are used. As expected, the computational time increases as the level of coupling is increased, and as the flow model becomes more complicated. Including the effects of nonequilibrium in the flow solution is the largest source of increased computational time between the three different cases listed. In all cases, the vast majority of computational time is spent

computing the flow solution.

5.4   Summary

This chapter presented the results from two different test cases that involved fluid-thermal-structural coupling. The first test case investigated the aerothermal response of the IRV-2 reentry vehicle, including surface ablation of a carbon-carbon material. The results generated using the methods presented in this thesis were compared with results from a one-dimensional material response code, and also from the ASCC code. The multi-dimensional material response results were found to produce temperature contours that are more physically realistic than the one-dimensional results, but the predicted surface temperatures were higher for the multi-dimensional case, and the ablation rate was lower in the stagnation region. Additionally, it was found that the multi-dimensional material response code produced stagnation point temperature predictions that were closer to the ASCC results than the one-dimensional code, but the predicted stagnation point recession rate was considerably lower than the ASCC results.

The second test case that was studied consisted of an insulated metallic plate exposed to an hypersonic flow. As the temperature of the plate increased due to aerodynamic heating, thermal stresses within the material caused the plate to deform. The deformation then impacted the flow solution. It was found that the amount of simulation time between updates to the flow solution has a strong impact on the time evolution of the coupled solution. Shorter update times lead to a faster increase in plate temperature, and therefore an earlier start to the plate deformation. When the effects of flow pressure were included in the analysis, the aerothermoelastic response of the plate became unstable, and the quasi-static coupling procedure presented in this work began to break down. The time between flow updates was again found to have a significant impact on the start time and the severity of the instability.

Varying the pressure on the back side of the plate was found to strongly influence the time history of the plate deflection, and in some cases suppressed the appearance of unstable behavior. Additionally, it was found that for the flight conditions considered in this test case, the effects of thermal and chemical nonequilibrium in the flow field are negligible.

These results indicate that the aerothermoelastic framework implemented in the present work is robust for cases where the quasi-static coupling approximation is valid. When material deformation is accounted for in the framework, frequent updates of the flow solution are required for accurate time evolution of the coupled problem. This increases the computational cost of the approach presented.

CHAPTER VI

Conclusions

6.1 Summary

Chapter I introduced the concept of aerothermoelastic coupling, and discussed some of the difficulties that exist in modeling fluid-thermal-structural interactions for hypersonic vehicles. The high-speed flight environment leads to significant thermal loads on the structure of a vehicle, which can in turn lead to structural deformation and material property degradation. This sets up a coupled interaction between the fluid and structure that is driven by thermal effects. Depending on the type of hypersonic vehicle and its time of flight and trajectory, an ablative thermal protection system may be needed to ensure that the vehicle survives. Use of this type of TPS can minimize the impact of thermal effects on the load bearing structure of the vehicle, but it becomes necessary to model the chemical ablation and surface recession of the ablative material. The need to account for all of these different physical phenomena and the coupling between them, leads to a complicated computational modeling problem.

Chapter II presented the models used to study the fluid, thermal, and structural aspects of the coupled aerothermoelastic problem. The governing equations and relevant constitutive relations for each model were discussed. The chapter first described LeMANS, a Navier-Stokes code developed previously at the University of Michigan,

that is used for the flow field modeling in this study. Next, details on the equations, modeling assumptions, and boundary conditions were given for the thermal and structural material response code that is developed in this work. The total Lagrangian structural formulation was discussed, along with the use of the 2$^{\text{nd}}$ Piola-Kirchhoff stress tensor and the Green-Lagrange strain tensor.

Chapter III discussed the numerical approaches used to solve the governing equations for the models presented in Chapter II. The use of the finite volume method in LeMANS was described first, along with an overview of the ghost-cell method for implementing boundary conditions. Next, the control volume finite element method was introduced, and its application to the thermal and structural governing equations was discussed. Details were given on the systems of nonlinear equations that result from the CVFEM. A discussion of the time integration methods that are used, along with the Newton-Raphson method was also presented. Finally, the chapter described the partitioned framework that is used to couple the fluid, thermal, and structural aspects of an aerothermoelastic problem. The quasi-static coupling assumption and the types of problems where it is valid were discussed. Also discussed was the interpolation scheme between the fluid and solid meshes, and the boundary conditions that must be applied at the interface between the fluid and solid.

Chapter IV discussed the idea of code verification, and introduced the Method of Manufactured Solutions (MMS). The application of the MMS to LeMANS and the thermal-structural material response code was presented, including the manufactured solutions used in each case, and the handling of specialized boundary conditions. The chapter next discussed code order of accuracy, and how it is computed for the purpose of code verification. Results from using the Method of Manufactured Solutions were then presented for LeMANS and the material response code. It was found that the original implementation of the supersonic inlet, supersonic outlet, and inviscid wall boundary conditions in LeMANS resulted in first-order accuracy, rather than the ex-

pected second-order accuracy. The implementation of these boundary conditions was modified, and second-order accuracy was demonstrated. The viscous wall boundary condition was also tested, and it was found that a lower than expected order of accuracy was observed based on the L_∞ error norms. The root cause of this decreased order of accuracy was not fully identified, however, it is thought to be related to the way that LeMANS computes viscous fluxes at the wall boundary, and is not the result of an error in the code.

The thermal material response code was tested for anisotropic, temperature dependent materials using a time dependent manufactured solution. For all boundary conditions and computational domains that were tested, the expected second-order accuracy was observed. The structural material response code was also tested for materials with temperature dependent properties, but using a steady manufactured solution. The specified displacement boundary condition showed the expected second-order accuracy in displacements and first-order accuracy in stresses. Next, the specified traction boundary condition was applied on three sides of the domain and the displacements were specified along the fourth side. Results from this case showed the expected orders of accuracy using the L_2 error norm, but the L_∞ norm showed a slightly lower than expected order of accuracy for the displacement components. This is thought to be the result of how the control volumes are constructed at mesh locations where the different boundary conditions meet, and it is not indicative of an error in the traction boundary condition.

A manufactured solution was also developed to allow for the testing of the inter-code interpolation between the fluid and solid meshes. This solution consisted of temperature distributions in the flow and solid domains that had the same temperature and temperature gradient at the fluid-solid interface, but different values away from the interface. This allowed for the wall heat flux values interpolated from the flow mesh to be used as the boundary conditions for the manufactured solution in

151

the solid domain. Using this approach, it was found that second order accuracy was maintained in the thermal code after the coupling process based on the L_2 norms, but the L_∞ norms showed first-order accuracy. This was again a result of the control volume construction at points where multiple boundary conditions meet.

The Method of Manufactured Solutions was found to be a valuable tool as part of the code development process. The ability to test realistic material properties and boundary conditions helped to build confidence in the tested codes. There are, however, some drawbacks to using the MMS as a verification tool. The manufactured source terms that are needed to balance the governing equations can become extremely large and complicated to work with. The use of a symbolic math program such as Mathematica helps to alleviate this difficulty, but does not eliminate it. Additionally, as the source terms become more complicated, the run time for a program can increase considerably. Finally, the sensitivity of the method to detecting differences from the expected order of accuracy even in very small regions of the domain can be both a blessing and a curse. These highly localized regions of error, sometimes only individual control volumes, are often not due to errors in the code, and they typically do not impact the rest of the solution, but they show up in the order of accuracy analysis in the same way that an actual error does. This can complicate the process of debugging a code.

Chapter V presented results for two different test cases that were studied using the aerothermoelastic framework outlined in Chapter III. LeMANS was used to obtain flow field solutions, and the material thermal and structural response code was used to find solutions in the structural domain. The first test case was an aerothermal analysis of the nosetip of the IRV-2 reentry vehicle, including a model for the surface ablation of carbon-carbon. A series of points along the IRV-2 reentry trajectory were modeled assuming an axisymmetric steady-state flow solution and an axisymmetric transient thermal solution. Code-to-code comparisons were made with

152

a one-dimensional material response code, and with the ASCC code. Significant differences in the temperature distributions were observed between the axisymmetric and one-dimensional results. The axisymmetric surface temperature results were found to compare slightly better with the ASCC code than the one-dimensional results, but the stagnation temperature and recession rate was still under-predicted compared with the ASCC results.

The second test case involved the aerothermoelastic analysis of a thermally insulated metal plate inclined at 5° to a hypersonic flow for a simulated time of 900.0 s. The flow field was assumed to be steady relative to the structural response, and the elastic response of the plate was assumed to be steady relative to the transient temperature solution. A series of numerical trajectory step lengths were investigated in order to determine the impact of updating the structural response and fluid solutions at longer time intervals. The update intervals that were used were considerably longer than those used for this test case with lower order models [27]. This was done to decrease the computational cost associated with using the high-fidelity models presented in the earlier chapters on a long thermal soak problem. It was found that in all cases, the time response of the panel was a strong function of the trajectory step length that was used, with shorter steps resulting in earlier, larger panel deformations. This resulted in different behavior at the 900.0 s simulation time. The different deformations were the result of faster panel heating for the shorter trajectory steps, and the fact that the plate did not reach an equilibrium temperature within the simulation time due to the thermal insulation. It was also found that for a test case where lower order models with increased time resolution predicted a dynamically stable response, the methods presented in this work resulted in a dynamically unstable response that caused the quasi-static coupling framework to break down. Before break down occurred, an oscillatory plate behavior was observed, with alternating large convex and concave panel deformations. These deformations had a significant impact on the flow

field, and led to relatively strong shocks and expansions forming over the plate. The pressure distribution across the back of the plate was varied, and this was found to have a strong impact on the response and stability of the panel. The impact of considering thermochemical nonequilibrium in the flow was also investigated, and it was found that for this case, the effects of nonequilibrium were negligible.

Based on the results presented, it is likely that much shorter trajectory step lengths must be used in order to obtain convergence in the time history of the panel response. This will greatly increase the computational cost associated with solving long simulation time problems using high-fidelity codes, even when a steady state flow model is used. Additionally, the stability of a solution response is not the same between this work and that performed using simplified aerodynamic models. This is potentially the result of using full Navier-Stokes CFD solutions that more accurately capture the nonlinear fluid response to panel deformations. It is also possible that the instabilities that were observed are a result of assumptions made in the coupling framework, and are not physically accurate.

6.2 Contributions

The work presented in this dissertation advances the simulation of coupled fluid-thermal-structural problems in hypersonic flows in several ways. Contributions are made to both the flow and material response models, as well as the aerothermoelastic coupling framework. The specific contributions are outlined below.

1. The main goal of this work, and the single largest contribution is the development of a new material response code that can solve for the multi-dimensional thermal and elastic structural behavior of arbitrary geometries undergoing large deformations. While the modeling approaches used are not new, the code is developed independently from other research groups and is designed with

the capability to be directly coupled with the LeMANS CFD code for use in aerothermal and aerothermoelastic simulations. Additionally, the use of separate thermal and structural portions of the code is beneficial for inclusion within a partitioned coupling approach, since it makes it easy to use widely varying time steps between the different physical processes. It also allows for easy implementation of simplifying assumptions such as quasi-static thermo-structural response. Finally, the capability to model surface ablation in the same code makes it a useful tool for a wide range of fluid-thermal-structural problems that are important when studying hypersonic vehicles. This flexibility allows it to be used as a base for the investigation of different aerothermal and aerothermoelastic problems by other researchers.

2. An aerothermoelastic framework is developed that allows for a steady state flow code to be coupled with transient thermal and structural response codes. This takes advantage of the large time scale separation between different physical phenomena that is often present in hypersonic cases. This is done in order to decrease overall computational time by not requiring a time accurate flow response. A method of dynamically determining how often to update the flow and structural solutions is proposed that is based on the local convergence of the fluid-solid interface values. Included in this framework is the coupling of a nonequilibrium Navier-Stokes solver with a full nonlinear structural response code. The coupling of these two components allows for much greater physical accuracy in the aerothermoelastic response of a structure at hypersonic speeds than is possible with either perfect gas CFD codes or lower order aerodynamic models.

3. The Method of Manufactured Solutions is used to verify several different codes, including a multi-species CFD code. In the process, commonly used methods

of enforcing boundary conditions in a hypersonic CFD code are shown to lead to lower order of accuracy than is expected. This illustrates the benefits of rigorously verifying even an established code. Additionally, a method of verifying a partitioned code framework using the Method of Manufactured Solutions is described and demonstrated for a flow-thermal coupling. This method can be extended to more complicated coupling scenarios, while still allowing for formal order of accuracy verification.

4. Results from a quasi-static, coupled, fluid-thermal-structural problem computed using the high-fidelity models presented in this work, show a strong dependence on the length of time between flow updates. This indicates that even in cases where the structural response is dynamically stable, the time between flow updates needs to be quite small in order to obtain convergence in the time evolution of the problem.

6.3 Future Work

The work presented in this thesis represents a first step towards a high-fidelity aerothermoelastic framework. There are still significant improvements that can be made to the individual codes, as well as to the coupling framework. Some of the most useful next steps are outlined here, along with potential directions in which the research could be continued.

It is important to continue testing of the aerothermoelastic framework via code-to-code comparisons and comparisons with lower fidelity models. These types of analyses help to build confidence in the code's implementation, and also serve to identify when the quasi-static coupling assumptions made in the current framework begin to break down. It would be interesting to couple a lower order structural model, such as von Kármán plate theory, with LeMANS using the same coupling approach, in order to

more clearly understand the impact of a high-fidelity flow model on the structural response of a panel. This would allow for a more exact comparison with published lower order results, and would also allow for the differences in results that are caused by the flow model and those that are caused by the structural model to be more easily separated.

The effect of the various coupling parameters in the present aerothermoelastic framework should be further investigated. In particular, the effect of trajectory step length, the parameters used to determine when the material response code is called from LeMANS, and the parameters that determine when LeMANS moves from one trajectory point to the next should be more fully quantified. Additional cases for the insulated metallic plate geometry should be run with shorter trajectory steps in order to determine when convergence in the time history of the panel response is obtained.

One of the most important modifications that should be made to the presented material response code is to implement the capability for parallel processing through domain decomposition. For the smaller two-dimensional problems considered in this work, it was feasible to solve for the material response on a single processor, however, as problems become more complicated, and as three-dimensional problems are investigated, it will become infeasible to use a single processor.

The linear algebra methods that are used in the material response code can be improved by switching to those that are available in LAPACK [2]. This would give access to highly optimized linear algebra routines for solving systems of equations, as well as access to a number of improved preconditioning routines compared to those currently used. The use of LAPACK could potentially decrease simulation times, improve code robustness, and cut down on future code development time.

In LeMANS, the introduction of full time-accuracy is important for the ability to model aeroelastic problems. Extending the coupling framework to include a time-accurate CFD code would allow for a much wider range of aerothermoelastic problems

to be investigated. In particular, the impact of thermochemical nonequilibrium at high Mach numbers on the dynamic response of structural components represents an important extension of this work.

Additionally, the effects of turbulence should be included in the aerothermoelastic coupling framework. Accounting for turbulence will lead to much higher heat fluxes, which could have a large impact on the deformation of a structure. The Baldwin-Lomax turbulence model is currently implemented in LeMANS, however, the inclusion of a more physics based model would be useful.

It would also be interesting to include the capability of modeling charring ablative materials, which include in-depth material decomposition. This would greatly extend the applicability of the material response code. In-depth decomposition, along with the ability to model the structural response, would allow for predictions of the stresses generated in a charring TPS material due to both temperature change and the internal pore pressure caused by pyrolysis gases. This could be further extended to the study of mechanical ablation.

APPENDICES

APPENDIX A

Constitutive Relations

This appendix gives the specific form of the constitutive matrices used in the structural mechanics portion of this work. These are the standard relations for elastic materials based on Hooke's law. Two different constitutive matrices are presented, the first for orthotropic materials, and the second for isotropic materials. In both cases the stress-strain relationship is given by Equation A.1, where $\mathbf{D}$ is the constitutive matrix.

$$
\begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{xy} \\ \sigma_{xz} \\ \sigma_{yz} \end{pmatrix} = \mathbf{D} \begin{pmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{zz} \\ \epsilon_{xy} \\ \epsilon_{xz} \\ \epsilon_{yz} \end{pmatrix} \tag{A.1}
$$

In the following matrices, ν_{ij} is Poisson's ratio, E_i is Young's modulus, and G_{ij} is the shear modulus. In the case of isotropic materials, the subscripts are dropped since the material properties do not depend on the coordinate direction. Note that the constitutive relations are written in terms of the true tensorial shear strains, and not the engineering shear strains. The engineering shear strains can be found by

multiplying the tensorial shear strains by 2.

Orthotropic Materials

$$\mathbf{D} = \begin{bmatrix} \frac{1-\nu_{yz}\nu_{zy}}{E_y E_z \Delta} & \frac{\nu_{yx}+\nu_{zx}\nu_{yz}}{E_y E_z \Delta} & \frac{\nu_{zx}+\nu_{yx}\nu_{zy}}{E_y E_z \Delta} & 0 & 0 & 0 \\[2mm] \frac{\nu_{xy}+\nu_{xz}\nu_{zy}}{E_z E_x \Delta} & \frac{1-\nu_{zx}\nu_{xz}}{E_z E_x \Delta} & \frac{\nu_{zy}+\nu_{zx}\nu_{xy}}{E_z E_x \Delta} & 0 & 0 & 0 \\[2mm] \frac{\nu_{xz}+\nu_{xy}\nu_{yz}}{E_x E_y \Delta} & \frac{\nu_{yz}+\nu_{xz}\nu_{yx}}{E_x E_y \Delta} & \frac{1-\nu_{xy}\nu_{yx}}{E_x E_y \Delta} & 0 & 0 & 0 \\[2mm] 0 & 0 & 0 & 2G_{xy} & 0 & 0 \\[2mm] 0 & 0 & 0 & 0 & 2G_{xz} & 0 \\[2mm] 0 & 0 & 0 & 0 & 0 & 2G_{yz} \end{bmatrix} \tag{A.2}$$

where,

$$\Delta = \frac{1 - \nu_{xy}\nu_{yx} - \nu_{yz}\nu_{zy} - \nu_{zx}\nu_{xz} - 2\nu_{xy}\nu_{yz}\nu_{zx}}{E_x E_y E_z}$$

Isotropic Materials

In the case of isotropic materials, the shear modulus can be written in terms of Young's modulus and Poisson's ratio as shown in Equation A.3. This leads to the constitutive matrix shown in Equation A.4.

$$G = \frac{E}{2\,(1+\nu)} \tag{A.3}$$

$$\mathbf{D} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & 1-2\nu & 0 & 0 \\ 0 & 0 & 0 & 0 & 1-2\nu & 0 \\ 0 & 0 & 0 & 0 & 0 & 1-2\nu \end{bmatrix} \tag{A.4}$$

APPENDIX B

Analytical Integrals

This appendix outlines the procedure for analytically computing the control volume integrals in a reference element. Examples are given for surface and volume integrals in triangular elements, and the extension to tetrahedral elements is discussed. The conduction and storage terms of the heat equation are used to illustrate the procedure. The process shown here is applied to the portion of each primary mesh element that contributes to the ith control volume in the dual mesh.

As discussed in Chapter III, integration is performed in a reference element, and the result is mapped back to physical space in order to assemble the discrete form of the governing equations. Figure B.1 shows the control volume contribution, denoted by the nodes (i, k_1, k_2, k_3), from a triangular element to the control volume around node i. The elemental control volume contribution is split into two triangular sub-control volumes, v_1 and v_2, defined by the indices (i, k_1, k_2) and (i, k_2, k_3), respectively. Each sub-volume has a surface denoted by f_1 or f_2. It is not necessary to integrate over the (i, k_1) or (i, k_3) surfaces since these are part of the interior of the control volume around node i. Surface integrals are computed over f_1 and f_2 separately, and the results are added together to compute the total total surface integral for the elemental control volume. The volume integrals over v_1 and v_2 are computed individually as

well. The reason for computing these integrals separately is to allow each sub-control volume to be mapped to a triangular reference element for integration, rather than integrating over the full diamond-shaped elemental control volume contribution. This greatly simplifies the integration.
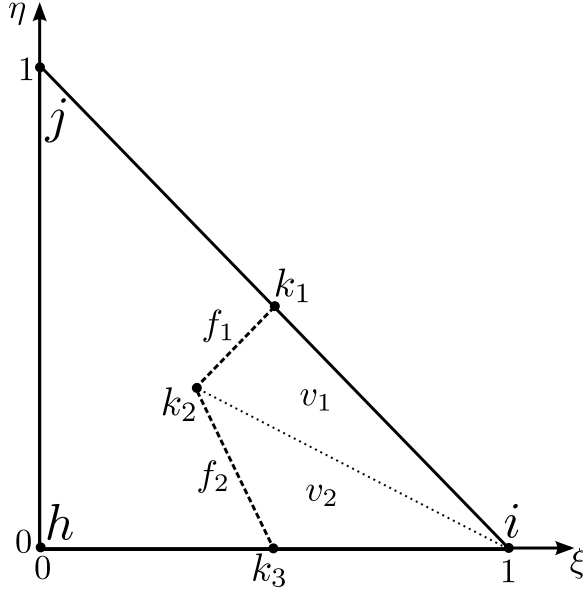
Figure B.1: Triangular reference element with the sub-control volume for node i indicated.

The procedure for integrating over the sub-control volume associated with node i is as follows. First, the necessary values from the h, i, and j nodes, which are nodes within the primary mesh, are interpolated to the k_1, k_2, and k_3 sub-element nodes using the linear shape functions given in Equation 3.5, and repeated here for convenience.

$$\text{triangles}\begin{cases} \phi_1 = 1 - \xi - \eta \\ \phi_2 = \xi \\ \phi_3 = \eta \end{cases} \qquad \begin{rcases} \phi_1 = 1 - \xi - \eta - \zeta \\ \phi_2 = \xi \\ \phi_3 = \zeta \\ \phi_4 = \eta \end{rcases}\text{tetrahedra}$$

164

Next, Jacobians are computed for the faces f_1 and f_2, or for the volumes v_1 and v_2, depending on whether surface or volume integration is being performed. Then, the integration is performed over a reference sub-control volume, assuming a linear variation of the properties at the sub-control volume nodes. The result from each sub-control volume is then mapped back to physical space using the Jacobians, and the sub-control volume results are added together to give the full elemental contribution to node i.

A specific example of this procedure is given below, using the conduction and storage terms from Equation 2.11, which is repeated here. The conduction term leads to a surface integral over the f_1 and f_2 faces, and the storage term leads to a volume integral over v_1 and v_2.

$$\underbrace{\int_{\partial\Omega} \tilde{q}_i \hat{n}_i \, \mathrm{d}\partial\Omega}_{\text{conduction}} - \int_{\partial\Omega} \rho h v_{cs_i} \hat{n}_i \, \mathrm{d}\partial\Omega + \frac{d}{dt}\underbrace{\int_{\Omega} \rho e \, \mathrm{d}\Omega}_{\text{storage}} = \int_{\Omega} Q \, \mathrm{d}\Omega$$

Since Fourier's law is used to compute heat flux, the quantities that need to be interpolated to the sub-control volume nodes are the thermal conductivity, the density, and the internal energy. The temperature derivatives used in the heat flux calculations are assumed to be constant over an element, so they do not need to be interpolated. Equation B.1 shows the interpolated values. The superscripts, k_m, where m ranges from 1 to 3, indicate to which sub-control volume node the element nodal values are being interpolated. They also indicate the location at which the shape functions are evaluated. The subscripts indicate which shape function is being used, and which element node value is being interpolated. The values of 1, 2, and 3 correspond to the i, j, and h nodes in Figure B.1. The thermal conductivity values are shown only for the κ_{xx} component of the conductivity tensor, but the other components are treated

165

in an identical fashion.

$$\hat{\kappa}_{xx}^{km} = \phi_1^{km} \kappa_{xx_1} + \phi_2^{km} \kappa_{xx_2} + \phi_3^{km} \kappa_{xx_3}$$

$$\hat{\rho}^{km} = \phi_1^{km} \rho_1 + \phi_2^{km} \rho_2 + \phi_3^{km} \rho_3 \tag{B.1}$$

$$\hat{e}^{km} = \phi_1^{km} e_1 + \phi_2^{km} e_2 + \phi_3^{km} e_3$$

The x-component of the conduction term is then computed for the f_1 and f_2 faces as shown in Equations B.2 and B.3, where J_{f_1} and J_{f_2} are the Jacobians for the two faces, and ξ is a reference element coordinate. The total control volume contribution is then the sum of the f_1 and f_2 integrals. The approach is identical for the y-component of heat flux, substituting the correct normal vectors and thermal conductivity terms.

$$
\begin{aligned}
\int_{f_1} q_x \hat{n}_x \, \mathrm{d}S &= -J_{f_1} \hat{n}_x \left[\frac{\partial T}{\partial x} \int_{-1}^{1} \frac{1}{2} \hat{\kappa}_{xx}^{k_1} (\xi + 1) + \frac{1}{2} \hat{\kappa}_{xx}^{k_2} (1 - \xi) \, \mathrm{d}\xi \right. \\
&\qquad \left. + \frac{\partial T}{\partial y} \int_{-1}^{1} \frac{1}{2} \hat{\kappa}_{xy}^{k_1} (\xi + 1) + \frac{1}{2} \hat{\kappa}_{xy}^{k_2} (1 - \xi) \, \mathrm{d}\xi \right] \\
&= -J_{f_1} \hat{n}_x \left[\frac{\partial T}{\partial x} \left(\hat{\kappa}_{xx}^{k_1} + \hat{\kappa}_{xx}^{k_2} \right) + \frac{\partial T}{\partial y} \left(\hat{\kappa}_{xy}^{k_1} + \hat{\kappa}_{xy}^{k_2} \right) \right]
\end{aligned} \tag{B.2}
$$

$$
\begin{aligned}
\int_{f_2} q_x \hat{n}_x \, \mathrm{d}S &= -J_{f_2} \hat{n}_x \left[\frac{\partial T}{\partial x} \int_{-1}^{1} \frac{1}{2} \hat{\kappa}_{xx}^{k_2} (\xi + 1) + \frac{1}{2} \hat{\kappa}_{xx}^{k_3} (1 - \xi) \, \mathrm{d}\xi \right. \\
&\qquad \left. + \frac{\partial T}{\partial y} \int_{-1}^{1} \frac{1}{2} \hat{\kappa}_{xy}^{k_2} (\xi + 1) + \frac{1}{2} \hat{\kappa}_{xy}^{k_3} (1 - \xi) \, \mathrm{d}\xi \right] \\
&= -J_{f_2} \hat{n}_x \left[\frac{\partial T}{\partial x} \left(\hat{\kappa}_{xx}^{k_2} + \hat{\kappa}_{xx}^{k_3} \right) + \frac{\partial T}{\partial y} \left(\hat{\kappa}_{xy}^{k_2} + \hat{\kappa}_{xy}^{k_3} \right) \right]
\end{aligned} \tag{B.3}
$$

The integrals of the storage term over the v_1 and v_2 sub-volumes proceed in a similar fashion. Equations B.4 and B.5 show the analytical integrals for this case. The total control volume contribution to node i is then the sum of the v_1 and v_2 integrals. Note that the density and internal energy are allowed to vary separately,

166

rather than just computing the linear variation of the ρe product.

$$
\begin{aligned}
\int_{v_1} \rho e \, \mathrm{d}V = J_{v_1} &\left[\int_0^1 \int_0^{1-\xi} \left(\hat{\rho}^i \left(1 - \xi - \eta \right) + \hat{\rho}^{k_1} \xi + \hat{\rho}^{k_2} \eta \right) \right. \\
&\left. \left(\hat{e}^i \left(1 - \xi - \eta \right) + \hat{e}^{k_1} \xi + \hat{e}^{k_2} \eta \right) \, \mathrm{d}\eta \, \mathrm{d}\xi \right] \\
= \frac{J_{v_1}}{24} &\left[\hat{\rho}^i \left(2\hat{e}^i + \hat{e}^{k_1} + \hat{e}^{k_2} \right) + \hat{\rho}^{k_1} \left(\hat{e}^i + 2\hat{e}^{k_1} + \hat{e}^{k_2} \right) \right. \\
&\left. + \hat{\rho}^{k_2} \left(\hat{e}^i + \hat{e}^{k_1} + 2\hat{e}^{k_2} \right) \right]
\end{aligned}
\tag{B.4}
$$

$$
\begin{aligned}
\int_{v_2} \rho e \, \mathrm{d}V = J_{v_2} &\left[\int_0^1 \int_0^{1-\xi} \left(\hat{\rho}^i \left(1 - \xi - \eta \right) + \hat{\rho}^{k_2} \xi + \hat{\rho}^{k_3} \eta \right) \right. \\
&\left. \left(\hat{e}^i \left(1 - \xi - \eta \right) + \hat{e}^{k_2} \xi + \hat{e}^{k_3} \eta \right) \, \mathrm{d}\eta \, \mathrm{d}\xi \right] \\
= \frac{J_{v_2}}{24} &\left[\hat{\rho}^i \left(2\hat{e}^i + \hat{e}^{k_2} + \hat{e}^{k_3} \right) + \hat{\rho}^{k_2} \left(\hat{e}^i + 2\hat{e}^{k_2} + \hat{e}^{k_3} \right) \right. \\
&\left. + \hat{\rho}^{k_3} \left(\hat{e}^i + \hat{e}^{k_2} + 2\hat{e}^{k_3} \right) \right]
\end{aligned}
\tag{B.5}
$$

The approach for tetrahedral elements is nearly identical, except the sub-control volume integrals are performed over triangular and tetrahedral reference elements, rather than lines and triangular elements. Figure B.2 shows the elemental control volume contribution from a tetrahedral element to node i. There are three different faces, f_1, f_2, and f_3, that make up the sub-control volume. In order to compute the analytical surface integrals, each of these faces is split into two triangles, and the integration is carried out in a reference triangular element. The volume integrals are computed by splitting the sub-control volume into six tetrahedra, two associated with each face. For example, face 3 is made up of the nodes (k_2, k_7, k_4, k_3), and the two tetrahedra associated with this face are given by (i, k_7, k_3, k_2) and (i, k_7, k_3, k_4). The total sub-control volume integral is found by summing over all of the faces or all of the volumes, depending on whether a surface or volume integral is being computed. The full sub-control volume integrals are not listed here, since they follow the same approach as the triangular elements, but using the tetrahedral shape functions from

167

Equation 3.5, and integrating over the additional ζ dimension.
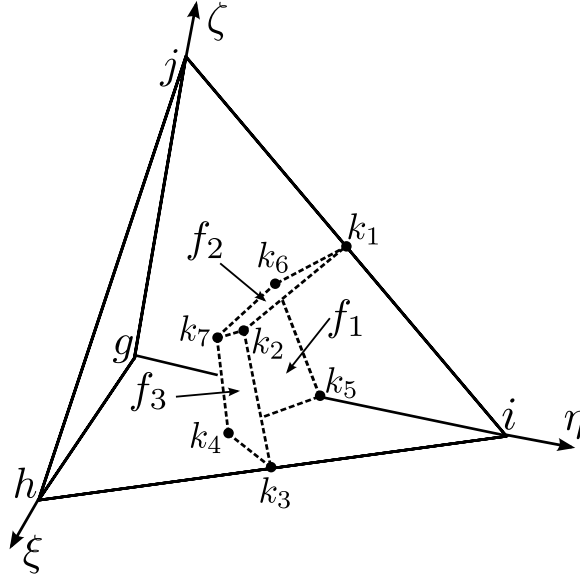
Figure B.2: Tetrahedral reference element with the sub-control volume for node i indicated.

The integrals for the structural equations are performed in exactly the same manner. The only differences are the quantities that are being integrated, and the fact that integration is performed over the initial undeformed geometry since the total Lagrangian formulation is used. This does not impact the procedure outlined above, but it means that the Jacobians used to map the solution from a reference element to a physical element are constant throughout a simulation, even if the material is deforming.

APPENDIX C

Manufactured Solution Constants

This appendix lists the values for the various constants that are used in the manufactured solutions from Chapter IV.

Equation	θ_0	θ_x	θ_y	θ_{xy}	a_x	a_y	a_{xy}	L
$\rho_{N_2}(x,y)$	0.5	0.15	-0.1	0.0	1.2	0.3	0.0	1.0
$\rho_N(\text{x,y})$	0.9	0.15	-0.1	0.0	0.7	1.3	0.0	1.0
$u(x,y)$	800.0	50.0	-30.0	0.0	2.3	0.6	0.0	1.0
$v(x,y)$	800.0	-75.0	40.0	0.0	0.7	0.667	0.0	1.0
$T_{tr}(x,y)$	330.0	67.5	169.0	0.0	2.2	1.4	0.0	1.0

Table C.1: Constants used in the manufactured solution for the LeMANS free stream boundary condition test case.

Equation	θ_0	θ_x	θ_y	θ_{xy}	a_x	a_y	a_{xy}	L
$\rho_{N_2}(x,y)$	0.5	0.15	-0.1	0.0	1.2	0.3	0.0	1.0
$\rho_N(\text{x,y})$	0.5	0.15	-0.1	0.0	1.2	0.3	0.0	1.0
$u(x,y)$	2000.0	50.0	-30.0	0.0	2.3	0.6	0.0	1.0
$v(x,y)$	-1500.0	-75.0	40.0	0.0	0.7	0.667	0.0	1.0
$T_{tr}(x,y)$	2000.0	67.5	169.0	0.0	2.2	1.4	0.0	1.0

Table C.2: Constants used in the manufactured solution for the LeMANS inviscid wall boundary condition test case.

Equation	θ_0	θ_x	θ_y	θ_{xy}	a_x	a_y	a_{xy}	L
$\rho_{N_2}(x,y)$	0.5	0.15	-0.1	0.0	1.2	0.3	0.0	1.0
$\rho_N(\mathrm{x,y})$	0.5	0.15	-0.1	0.0	1.2	0.3	0.0	1.0
$u(x,y)$	2000.0	50.0	-30.0	0.0	2.3	0.6	0.0	1.0
$v(x,y)$	-1500.0	-75.0	40.0	0.0	0.7	0.667	0.0	1.0
$T_{tr}(x,y)$	2000.0	67.5	169.0	0.0	2.2	1.4	0.0	1.0

Table C.3: Constants used in the manufactured solution for the LeMANS viscous wall boundary condition test case.

Equation	T_0	B_x	B_y	ω_T	ϵ_T
$T(x,y,t)$	300.0	100.0	66.667	0.5	400.0

Table C.4: Constants used in the manufactured solution for the thermal material response code with specified temperature and specified flux boundary conditions.

Equation	$\kappa_{0_{ij}}$	$\kappa_{1_{ij}}$	$\kappa_{2_{ij}}$	C_{p0}	C_{p1}
$\kappa_{xx}(T)$	15.0	2.0×10^{-3}	1.0×10^{-5}	–	–
$\kappa_{yy}(T)$	7.5	3.0×10^{-2}	2.0×10^{-4}	–	–
$\kappa_{xy}(T)$	3.0	4.0×10^{-3}	1.0×10^{-4}	–	–
$C_p(T)$	–	–	–	500.0	0.5

Table C.5: Constants used in the manufactured properties for the thermal material response code with specified temperature and specified flux boundary conditions.

Equation	θ_0	θ_x	θ_y	θ_{xy}	θ_t	a_x	a_y	a_{xy}	a_t	T_0	L
$u(x,y,t)$	0.0	0.2	0.0	0.05	1.0	0.5	0	0.3	0.0	300.0	1.0
$v(x,y,t)$	0.0	0.1	0.1	0.0	1.0	0.7	0.5	0.0	0.0	300.0	1.0
$T(x,y,t)$	–	77.0	42.0	0.0	1.0	0.6	0.2	0.0	150.0	300.0	1.0

Table C.6: Constants used in the manufactured solution for the structural material response code with specified displacement and specified traction boundary conditions.

Equation	E_0	E_1	E_2
$E(T)$	6.0×10^{10}	-1.0×10^{8}	-1.0×10^{5}

Table C.7: Constants used in the manufactured Young's modulus for the structural material response code with specified displacement and specified traction boundary conditions.

Equation	α_0	α_1	α_2
$\alpha(T)$	4.0×10^{-6}	1.0×10^{-8}	1.0×10^{-9}

Table C.8: Constants used in the manufactured thermal expansion coefficient for the structural material response code with specified displacement and specified traction boundary conditions.

Equation	ν_0	ν_1	ν_2
$\nu(T)$	1.2×10^{-1}	2.0×10^{-4}	1.0×10^{-6}

Table C.9: Constants used in the manufactured Poisson's ratio for the structural material response code with specified displacement and specified traction boundary conditions.

Equation	θ_0	θ_x	θ_y	θ_{xy}	θ_t	a_x	a_y	a_{xy}	a_t	L
$\rho_{N_2}(x,y)$	0.75	0.0	1.0	0.0	–	1.2	0.0	0.0	–	1.0
$\rho_N(x,y)$	0.75	0.0	1.0	0.0	–	1.2	0.0	0.0	–	1.0
$u(x,y)$	-5500.0	0.0	0.0	0.0	–	0.0	0.0	0.0	–	1.0
$v(x,y)$	500.0	0.0	0.0	0.0	–	0.0	0.0	0.0	–	1.0
$T_{tr}(x,y)$	700.0	67.5	169.0	127.0	–	2.2	1.4	1.0	–	1.0
$T_s(x,y,t)$	700.0	67.5	169.0	127.0	100.0	2.2	1.4	1.0	2400.0	1.0

Table C.10: Constants used in the manufactured solution for the coupled Le-MANS/thermal response test case.

APPENDIX D

Material Properties

This appendix lists the thermophysical properties of carbon-carbon that are used for the IRV-2 simulations in Chapter V. These quantities are obtained from Amar [5].

Temperature [K]	Specific heat $J/kg-K$	Conductivity $W/m-K$	Emissivity
255.56	544.284	64.176	0.8
533.33	921.096	64.799	0.8
811.11	1318.842	53.583	0.8
1088.89	1624.478	46.107	0.8
1366.67	1775.203	39.253	0.8
1644.44	1925.928	34.891	0.8
1922.22	2009.664	31.776	0.8
2200.00	2080.840	30.530	0.8
2477.78	2122.708	29.907	0.8
2755.56	2143.642	29.907	0.8
3033.33	2177.136	29.284	0.8
5555.56	2177.136	29.284	0.8

Table D.1: Thermophysical properties of carbon-carbon.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] U.S. standard atmosphere, 1976, 1976. NASA-TM-X-74335.

[2] Lapack homepage, August 2014.

[3] Metallic materials and elements for aerospace vehicle structures, Jan. 2003. MIL-HDBK-5J.

[4] Hicham Alkandry, Erin D. Farbar, and Iain D. Boyd. Evaluation of finite-rate surface chemistry models for simulation of the stardust reentry capsule. In *43rd AIAA Thermophysics Conference*, New Orleans, Louisiana, June 25-28 2012. AIAA paper 2012-2874.

[5] A. J. Amar. Modeling of one-dimensional ablation with porous flow using finite control volume procedure. Master's thesis, North Carolina State University, 2006.

[6] Adam J. Amar, Nathan D. Calvert, and Benjamin S. Kirk. Development and verification of the charring ablating thermal protection implicit system solver. In *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, Orlando, Florida, January 4-7 2011. AIAA paper 2011-144.

[7] John D. Jr. Anderson. *Modern Compressible Flow with Historical Perspective*. McGraw Hill, 3 edition, 2003.

[8] H. Ashley and G. Zartarian. Piston theory - a new aerodynamic tool for the aeroelastician. *Journal of the Aeronautical Sciences*, 23(12):1109–1118, 1956.

[9] C Bailey and M Cross. A finite volume procedure to solve elastic solid mechanics problems in three dimensions on an unstructured mesh. *International Journal for Numerical Method in Engineering*, 38:1757–1776, 1995.

[10] B. R. Baliga and S. V. Patankar. A control volume finite-element method for two-dimensional fluid flow and heat transfer. *Numerical Heat Transfer*, 6(3):245–261, 1983.

[11] Bantwal Rabindranath Baliga. *A Control-Volume Based Finite-Element Method for Convective Heat and Mass Transfer*. PhD thesis, University of Minnesota, 1978.

[12] R. E. Bartels, C. L. Rumsey, and R. T. Biedron. CFL3D user's manual - general usage and aeroelastic analysis (version 6.4). Technical report, 2006. NASA TM 2006-214301.

[13] Klaus-Jürgen Bathe. *Finite Element Procedures*. Prentice Hall, 1996.

[14] R. L. Bisplinghoff. Some structural and aeroelastic considerations of high-speed flight. *Journal of the Aeronautical Sciences*, 23(4):289–321, 1956. The Nineteenth Wright Brothers Lecture.

[15] B. F. Blackwell and R. E. Hogan. Numerical solution of axisymmetric heat conduction problems using finite control volume technique. *Journal of Thermophysics and Heat Transfer*, 7(3):462–471, 1993.

[16] F.G. Blottner, Margaret Johnson, and Molly Ellis. Chemically reacting viscous flow program for multi-component gas mixtures. Technical report, Sandia Laboratories, Albuquerque, New Mexico, December 1971. SC-RR-70-754.

[17] B. A. Boley. Approximate analyses of thermally induced vibrations. *Journal of Applied Mechanics*, 39(1):212–216, 1972.

[18] R.B. Bond, C.C. Ober, P.M. Knupp, and S.W. Bova. Manufactured solution for computational fluid dynamics boundary condition verification. *AIAA Journal*, 45(9), 2007.

[19] Ryan B. Bond, Patrick M. Knupp, and Curtis C. Ober. A manufactured solution for verifying cfd boundary conditions. In *34th AIAA Fluid Dynamics Conference and Exhibit*, Portland, Oregon, June 28 - July 1 2004. AIAA paper 2004-2029.

[20] Ryan B. Bond, Patrick M. Knupp, and Curtis C. Ober. A manufactured solution for verifying cfd boundary conditions, part ii. In *43rd AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, January 10-13 2005. AIAA paper 2005-88.

[21] Ryan B. Bond, Patrick M. Knupp, and Curtis C. Ober. A manufactured solution for verifying cfd boundary conditions, part iii. In *36th AIAA Fluid Dynamics Conference and Exhibit*, San Francisco, CA, June 5-8 2006. AIAA paper 2005-3722.

[22] René de Borst, Mike A. Crisfield, Joris J. C. Remmers, and Clemens V. Verhoosel. *Non-linear Finite Element Analysis of Solids and Structures*. John Wiley & Sons, Ltd., 2012.

[23] Graham V. Candler and Robert W. MacCormack. Computation of weakly ionized hypersonic flows in thermochemical nonequilibrium. *Journal of Thermophysics*, 5(3):266–273, 1991.

[24] Aniruddha Choudhary, Christopher J. Roy, Edward A. Luke, and Subrahmanya P Veluri. Issues in verifying boundary conditions for 3d unstructured cfd codes. In *20th AIAA Computational Fluid Dynamics Conference*, Honolulu, Hawaii, June 27-30 2011. AIAA paper 2011-3868.

[25] Andrew R. Crowell, Brent A. Miller, and Jack J. McNamara. Computational modeling for conjugate heat transfer of shock-surface interactions on compliant skin panels. In *52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Denver, CO, April 4-7 2011. AIAA paper 2011-2017.

[26] Adam J. Culler. *Coupled Fluid-Thermal-Structural Modeling and Analysis of Hypersonic Flight Vehicle Structures*. PhD thesis, The Ohio State University, 2010.

[27] Adam J. Culler and Jack J. McNamara. Studies on fluid-thermal-structural coupling for aerothemoelasticity in hypersonic flow. *AIAA Journal*, 48(8), 2010.

[28] Adam J. Culler and Jack J. McNamara. Impact of fluid-thermal-structural coupling on response prediction of hypersonic skin panels. *AIAA Journal*, 49(11), 2011.

[29] John A. Dec. *Three Dimensional Finite Element Ablative Thermal Response Analysis Applied to Heatshield Penetration Design*. PhD thesis, Georgia Institute of Technology, 2010.

[30] Pramote Dechaumphai, Earl A. Thornton, and Allan R. Wieting. Flow-thermal-structural study of aerodynamically heated leading edges. *Journal of Spacecraft and Rockets*, 26(4):201–209, 1989.

[31] Pramote Dechaumphai, Allan R. Wieting, and Ajay K. Pandey. Fluid-thermal-structural interaction of aerodynamically heated leading edges. In *30th Structures, Structural Dynamics, and Materials Conference*, 1989. AIAA paper 1989-1227.

[32] Robert V. Doggett, Jr., Rodney H. Ricketts, T. E. Noll, and John B. Malone. Nasp aeroservothermoelasticity studies. Technical report, April 1991. NASA TM-104058.

[33] E. H. Dowell. Nonlinear oscillations of a fluttering plate. *AIAA Journal*, 4(7):1267–1275, 1966.

[34] Marie-Claude Druguet, Graham V. Candler, and Ioannis Nompelis. Effect of numerics on navier-stokes computations of hypersonic double-cone flows. *AIAA Journal*, 43(3), 2005.

[35] John Dugundji. Theoretical considerations of panel flutter at high supersonic mach numbers. *AIAA Journal*, 4(7):1257–1266, 1966.

[36] E. R. G. Eckert. Engineering relations for heat transfer and friction in high-velocity laminar and turbulent boundary-layer flow over surfaces with constant pressure and temperature. *Transactions of the ASME*, 78(6):1273–1283, 1956.

[37] S. Étienne, A. Garon, and D. Pelletier. Some manufactured solutions for verification of fluid-structure interaction codes. *Computers and Structures*, 106-107, 2012.

[38] S. Étienne, D. Pelletier, and A. Garon. A monolithic formulation for steady-state fluid-structure interaction problems. In *34th AIAA Fluid Dynamics Conference and Exhibit*, Portland, OR, June 28 - July 1 2004. AIAA paper 2004-2239.

[39] N. A. Fallah, C. Bailey, M. Cross, and G. A. Taylor. Comparison of finite element and finite volume methods application in geometrically nonlinear stress analysis. *Applied Mathematical Modelling*, 24(7):439–455, 2000.

[40] Erin Farbar, Hicham Alkandry, Jonathan Wiebenga, and Iain D. Boyd. Simulation of ablating hypersonic vehicles with finite-rate surface chemistry. In *11th AIAA/ASME Joint Thermophysics and Heat Transfer Conference*, Atlanta, GA, June 16-20 2014. AIAA paper 2014-2124.

[41] Gerald E. Farin. *Curves and Surfaces for Computed Aided Geometric Design: A Practical Guide*. Boston: Academic Press, 1988.

[42] D. Folkner, A. Katz, and V. Sankaran. Design and verification methodology of boundary conditions for finite volume schemes. *Computers & Fluids*, 96:264–275, 2014.

[43] Peretz P. Friedmann. Renaissance of aeroelasticity and its future. *Journal of Aircraft*, 36(1):105–121, 1999.

[44] M. W. Gee, U. Küttler, and W. A. Wall. Truly monolithic algebraic multigrid for fluid-structure interaction. *International Journal for Numerical Methods in Engineering*, 85:987–1016, 2011.

[45] Peter A. Gnoffo, Roop N. Gupta, and Judy L. Shinn. Conservation equations and physical models for hypersonic air flows in thermal and chemical nonequilibrium. Technical report, NASA, February 1989. NASA-TP-2867.

[46] Charles. Hirsch. *Numerical Computation of Internal & External Flows, Volume 2: Computational Methods for Inviscid and Viscous Flows*. John Wiley & Sons, 1 edition, 1990.

[47] Charles. Hirsch. *Numerical Computation of Internal & External Flows*. Butterworth-Heinemann, 2 edition, 2007.

[48] R. E. Hogan, B. F. Blackwell, and R. J. Cochran. Application of moving grid control volume finite element method to ablation problems. *Journal of Thermophysics and Heat Transfer*, 10(2):312–319, April-June 1996.

177

[49] B. Hübner, E. Walhorn, and D. Dinkler. A monolithic approach to fluid-structure interaction using space-time finite elements. *Computer Methods in Applied Mechanics and Engineering*, 193:2087–2104, 2004.

[50] P. Jawahar and Hemant Kamath. A high-resolution procedure for euler and navier-stokes computations on unstructured grids. *Journal of Computational Physics*, 164:165–203, 2000.

[51] Ojas Joshi and Pénélope Leyland. Stability analysis of a partitioned fluid-structure thermal coupling algorithm. In *43rd AIAA Thermophysics Conference*, New Orleans, LA, June 25-28 2012. AIAA paper 2012-3185.

[52] H .H. C. King, K. K. Muramoto, A. L. Murray, and S. W. Pronchick. *ABRES Shape Change Code (ASCC86) - Technical Report and User's Manual*. Acurex Corp., Aerotherm Div., Mountain View, CA, December 1986. Rept. BMO TR-87-57.

[53] Patrick Knupp and Kambiz Salari. *Verification of Computer Codes in Computational Science and Engineering*. Chapman & Hall/CRC, 2003.

[54] Dean A. Kontinos and Grant Palmer. Numerical simulation of metallic thermal protection system panel bowing. *Journal of Spacecraft and Rockets*, 36(6):842–849, 1999.

[55] David W. Kuntz, Basil Hassan, and Donald L. Potter. Predictions of ablating hypersonic vehicles using an iterative coupled fluid/thermal approach. *Journal of Thermophysics and Heat Transfer*, 15(2):129–139, 2001.

[56] Jong-Hun Lee. *Basic Governing Equations for the Flight Regimes of Aeroassisted Orbital Transfer Vehicles*, volume 96 of *Progress in Astronautics and Aeronautics*, pages 3–53. AIAA, 1985. in Thermal Design of Aeroassisted Orbital Transfer Vehicles.

[57] M. J. Lighthill. Oscillating airfoils at high mach number. *Journal of the Aeronautical Sciences*, 20(6):402–406, June 1953.

[58] Rainald Löhner, Chi Yang, Juan Cebral, Joseph D. Baum, Hong Luo, Daniele Pelessone, and Charles Charman. Fluid-structure-thermal interaction using a loose coupling algorithm and adaptive unstructured grids. In *29th AIAA Fluid Dynamics Conference*, Albuquerque, NM, 1998. AIAA paper 1998-2419.

[59] D. R. Lynch and K. O'Neill. Elastic grid deformation for moving boundary problems in two space dimensions. *Finite Elements in Water Resources*, 2:7.111–7.120, 1980.

[60] Robert W. MacCormack and Graham V. Candler. The solution of the navier-stokes equations using gauss-seidel line relaxation. *Computers and Fluids*, 17(1):135–150, 1989.

[61] Mal, Ajit K. *Deformation of Elastic Solids.* Prentice-Hall, Inc., 1991.

[62] A. Martin and I. D. Boyd. Simulation of pyrolysis gas within a thermal protection system. In *40th AIAA Thermophysics Conference*, Seattle, Washington, June 23-26 2008. AIAA paper 2008-3805.

[63] A. Martin and I. D. Boyd. Implicit implementation of material response and moving meshes for hypersonic re-entry ablation. In *47th AIAA Aerospace Sciences Meeting and Exhibit*, Orlando, Florida, January 5-8 2009. AIAA paper 2009-0670.

[64] A. Martin and I. D. Boyd. Strongly coupled computation of material response and nonequilibrium flow for hypersonic ablation. In *41st AIAA Thermophysics Conference*, San Antonio, Texas, June 22-25 2009. AIAA paper 2009-3597.

[65] A. Martin and I. D. Boyd. Mesh tailoring for strongly coupled computation of ablative material in nonequilibrium hypersonic flow. In *10th AIAA/ASME Joint Thermophysics and Heat Transfer Conference*, Chicago, Illinois, June 28 - July 1 2010. AIAA paper 2010-5062.

[66] Alexandre Martin, Leonardo C. Scalabrin, and Iain D. Boyd. High performance modeling of atmospheric re-entry vehicles. *Journal of Physics: Conference Series*, 341, 2012. Article 012002.

[67] Jack J. McNamara and Peretz P. Friedmann. Aeroelastic and aerothermoelastic analysis in hypersonic flow: Past, present, and future. *AIAA Journal*, 49(6):1089–1122, 2011.

[68] Megson, T.H.G. *Aircraft Structures for Engineering Students.* Elsevier, 4 edition, 2007.

[69] C. Mei, K. Abdel-Motagaly, and R. Chen. Reveiw of nonlinear panel flutter at supersonic and hypersonic speeds. *Applied Mechanics Reviews*, 52(10):321–332, 1999.

[70] Brent A. Miller, Andrew R. Crowell, and Jack J. McNamara. Loosely coupled time-marching of fluid-thermal-structural interactions. In *54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Boston, MA, April 8-11 2013. AIAA paper 2013-1666.

[71] C. B. Moyer and R. A. Rindal. An analysis of the coupled chemically reacting boundary layer and charring ablator, part ii: Finite difference solution for the in-depth response of charring materials considering surface chemical and energy balances. Technical report, June 1968. NASA CR-1061.

[72] David E. Myers, Carl J. Martin, and Max L. Blosser. Parametric weight comparison of advanced metallic, ceramic tile, and ceramic blanket thermal protection systems. Technical report, NASA, 2000. NASA/TM-2000-210289.

[73] Nathan M. Newmark. A method of computation for structural dynamics. *ASCE Journal of the Engineering Mechanics Division*, 85:67–94, 1959.

[74] William L. Oberkampf and Timothy G. Trucano. Verification and validation in computational fluid dynamics. *Progress in Aerospace Sciences*, 38:209–272, 2002.

[75] George C. Olsen and R. E. Smith. Analysis of aerothermal loads on spherical dome protuberances. *AIAA Journal*, 23(5):650–656, 1985.

[76] David Olynick, Y. K. Chen, and Michael E. Tauber. Aerothermodynamics of the stardust sample return capsule. *Journal of Spacecraft and Rockets*, 36(3):442–462, 1999.

[77] Chul Park. *Nonequilibrium Hypersonic Aerothermodynamics*. John Wiley & Sons, 1990.

[78] C. A. Powars and R. M. Kendall. *Aerotherm Chemical Equilibrium (ACE) Computer Program - User's Manual*. Aerotherm Corporation, Mountain View, CA, May 1969.

[79] Willam H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in Fortran 77: The Art of Scientific Computing*. Cambridge University Press, 2nd edition, 1992.

[80] Patrick J. Roache. *Verification and Validation in Compuational Science and Engineering*. Hermosa Publishers, 1998.

[81] Patrick J. Roache. Verification of codes and calculations. *AIAA Journal*, 36(5), 1998.

[82] C. J. Roy, C. C. Nelson, T. M. Smith, and C. C. Ober. Verification of euler/navier-stokes codes using the method of manufactured solutions. *International Journal for Numerical Methods in Fluids*, 44:599–620, 2004.

[83] Christopher J. Roy. Review of code and solution verification procedures for computational simulation. *Journal of Computational Physics*, 205(1):131–156, 2005.

[84] Christopher J. Roy, Mary A. McWherter-Payne, and William L. Oberkampf. Verification and validation for laminar hypersonic flowfields, part 1: Verification. *AIAA Journal*, 41(10):1934–1943, 2003.

[85] Youcef Saad and Martin H. Schultz. Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, July 1986.

[86] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, 2nd edition, 2003.

[87] L. D. Scalabrin. *Numerical Simulation of Weakly Ionized Hypersonic Flow Over Reentry Capsules.* PhD thesis, University of Michigan, 2007.

[88] A. K. Slone, C. Bailey, and M. Cross. Dynamic solid mechanics using finite volume methods. *Applied Mathematical Modeling*, 27:69–87, 2003.

[89] Josepsh L. Steger and R. F. Warming. Flux vector splitting of the inviscid gasdynamic equations with application to finite-difference methods. *Journal of Computational Physics*, 40:263–293, 1981.

[90] Stanly Steinberg and Patrick J. Roache. Symbolic manipulation and computational fluid dynamics. *Journal of Computational Physics*, 57(2):251–284, 1985.

[91] Kenneth Sutton and Peter A. Gnoffo. Multi-component diffusion with application to computational aerothermodynamics. In *AIAA/ASME Joint Thermophysics and Heat Transfer Conference*, Albuquerque, NM, June 15-18 1998. AIAA paper 1998-2575.

[92] Tecplot, Inc. *Tecplot 360.* Tecplot, Inc., Bellevue, WA, 12.2.0 edition, 2010.

[93] Earl A. Thornton. Thermal structures: Four decades of progress. *Journal of Aircraft*, 29(3), 1992.

[94] Earl A. Thornton. *Thermal Structures for Aerospace Applications.* American Institute of Aeronautics and Astronautics, 1996.

[95] Earl A. Thornton and Pramote Dechaumphai. Coupled flow, thermal, and structural analysis of aerodynamically heated panels. *Journal of Aircraft*, 25(11):1052–1059, 1988.

[96] Eleuterio F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction.* Springer, 2009.

[97] R. W. Truitt. *Fundamentals of Aerodynamic Heating.* Ronald Press, New York, 1960.

[98] A. S. Usmani, J. M. Rotter, S. Lamont, A. M. Sanad, and M. Gillie. Fundamental principles of structural behaviour under thermal effects. *Fire Safety Journal*, 36:721–744, 2001.

[99] E. R. Van Driest. The problem of aerodynamic heating. *Aeronautical Engineering Review*, 15(10):26–41, 1956.

[100] Bram van Leer. Towards the ultimate conservative difference scheme v. a second-order sequel to godunov's method. *Journal of Computational Physics*, 32:101–136, 1979.

[101] Subrahmanya P. Veluri, Christopher J. Roy, Shelley Hebert, and Edward, A. Luke. Verification of the loci-chem cfd code using the method of manufactured solutions. In *46th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, January 7-10 2008. AIAA paper 2008-661.

[102] Walter G. Vincenti and Charles H. Kruger. *Introduction to Physical Gas Dynamics*. Krieger Publishing Company, 1965.

[103] Marcel Vinokur. On one-dimensional stretching functions for finite-difference calculations. *Journal of Computational Physics*, 50:215–234, 1983.

[104] Vaughn R. Voller. *Basic Control Volume Finite Element Methods for Fluids and Solids*. World Scientific Publishing Co. Pte. Ltd., 2009.

[105] Frank M. White. *Viscous Fluid Flow*. McGraw Hill, 3 edition, 2006.

[106] Jonathan E. Wiebenga and Iain D. Boyd. Computation of multi-dimensional material response coupled to hypersonic flow. In *43rd AIAA Thermophysics Conference*, New Orleans, LA, June 25-28 2012. AIAA paper 2012-2873.

[107] C. R. Wilke. A viscosity equation for gas mixtures. *The Journal of Chemical Physics*, 18(4), April 1950.

[108] Wolfram Research, Inc. *Mathematica*. Wolfram Research, Inc., Champaign, Illinois, version 8.0 edition, 2010.

[109] Michael James Wright. *A Family of Data-Parallel Relaxation Methods for the Navier-Stokes Equations*. PhD thesis, University of Minnesota, 1997.

[110] O. C. Zienkiewicz and R. L. Taylor. *The Finite Element Method*. McGraw-Hill, fourth edition, 1989.

ABSTRACT

High-Fidelity Material Response Modeling as Part of an Aerothermoelastic
Framework for Hypersonic Flows

by

Jonathan E. Wiebenga

Chair: Iain D. Boyd

The development of a numerical code capable of simulating the multidimensional thermal and structural response of materials exposed to hypersonic flow is presented. The control volume finite element method (CVFEM) is used for the spatial discretization, and implicit time integration is used in both the thermal and structural portions of the code. Fourier's law is used to compute heat fluxes, and the thermal portion of the code can model materials with temperature-dependent, anisotropic properties. A total Lagrangian structural mechanics formulation is used along with the Green-Lagrange strain tensor and the generalized Hooke's law, to allow for simulation of large elastic deformations of temperature-dependent, orthotropic materials. The Method of Manufactured Solutions (MMS) is used to verify the implementation of several aspects of the material response code.

The material response code is coupled with a Computational Fluid Dynamics (CFD) code using a partitioned framework, in order to study quasi-static aerothermal and aerothermoelastic problems that arise in hypersonic flow. Characteristic times for the flow component of the coupled problem are assumed to be much faster than the thermal and structural time scales, allowing for steady state flow solutions to be used. The thermal and structural material response is updated as the flow solution

converges to steady state. The temperature solution is used to compute thermal stresses in the structural response. The MMS is used to verify the CFD code and some aspects of the coupling procedure.

Two test cases are presented to demonstrate the performance of the material response code and the coupling framework. The first case simulates the aerothermal response, including surface ablation, of a reentry vehicle. Comparisons are made with results from other codes, and it is found that accounting for multidimensional heat transport within the vehicle leads to higher surface temperatures, but lower surface recession. The second test case models a thermally insulated compliant panel in an hypersonic flow. A strong dependence on the time between computing updated flow solutions exists, even for cases that are dynamically stable. Additionally, dynamic panel response is observed for some test conditions, and the impact of this behavior on the quasi-static coupling framework is discussed.