

5-12-2023

Monolithic multiphysics simulation of hypersonic aerothermoelasticity using a hybridized discontinuous Galerkin method

William Paul England
Mississippi State University, wpe23@msstate.edu

Follow this and additional works at: <https://scholarsjunction.msstate.edu/td>



Part of the [Fluid Dynamics Commons](#), [Numerical Analysis and Scientific Computing Commons](#), and the [Partial Differential Equations Commons](#)

Recommended Citation

England, William Paul, "Monolithic multiphysics simulation of hypersonic aerothermoelasticity using a hybridized discontinuous Galerkin method" (2023). *Theses and Dissertations*. 5741.
<https://scholarsjunction.msstate.edu/td/5741>

This Dissertation - Open Access is brought to you for free and open access by the Theses and Dissertations at Scholars Junction. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of Scholars Junction. For more information, please contact scholcomm@msstate.libanswers.com.

Monolithic multiphysics simulation of hypersonic aerothermoelasticity using a hybridized
discontinuous Galerkin method

By

William Paul England

Approved by:

Ioana Banicescu (Major Professor)

Adrian Sescu

Shanti Bhushan

Justin Foster

Ian Dettwiller

Adrian Sescu (Graduate Coordinator)

Jason M. Keith (Dean, Bagley College of
Engineering)

A Dissertation

Submitted to the Faculty of

Mississippi State University

in Partial Fulfillment of the Requirements

for the Degree of Doctor of Philosophy

in Computational Engineering

in the James Worth Bagley College of Engineering

Mississippi State, Mississippi

May 2023

Copyright by
William Paul England
2023

Name: William Paul England

Date of Degree: May 12, 2023

Institution: Mississippi State University

Major Field: Computational Engineering

Major Professor: Ioana Banicescu

Title of Study: Monolithic multiphysics simulation of hypersonic aerothermoelasticity using a hybridized discontinuous Galerkin method

Pages of Study: 110

Candidate for Degree of Doctor of Philosophy

This work presents implementation of a hybridized discontinuous Galerkin (DG) method for robust simulation of the hypersonic aerothermoelastic multiphysics system. Simulation of hypersonic vehicles requires accurate resolution of complex multiphysics interactions including the effects of high-speed turbulent flow, extreme heating, and vehicle deformation due to considerable pressure loads and thermal stresses. However, the state-of-the-art procedures for hypersonic aerothermoelasticity are comprised of low-fidelity approaches and partitioned coupling schemes. These approaches preclude robust design and analysis of hypersonic vehicles for a number of reasons. First, low-fidelity approaches limit their application to simple geometries and lack the ability to capture small scale flow features (e.g. turbulence, shocks, and boundary layers) which greatly degrades modeling robustness and solution accuracy. Second, partitioned coupling approaches can introduce considerable temporal and spatial inaccuracies which are not trivially remedied. In light of these barriers, we propose development of a monolithically-coupled hybridized DG approach to enable robust design and analysis of hypersonic vehicles with arbitrary geometries.

Monolithic coupling methods implement a coupled multiphysics system as a single, or monolithic, equation system to be resolved by a single simulation approach. Further, monolithic approaches are free from the physical inaccuracies and instabilities imposed by partitioned approaches and enable time-accurate evolution of the coupled physics system. In this work, a DG method is considered due to its ability to accurately resolve second-order partial differential equations (PDEs) of all classes. We note that the hypersonic aerothermoelastic system is composed of PDEs of all three classes. Hybridized DG methods are specifically considered due to their exceptional computational efficiency compared to traditional DG methods. It is expected that our monolithic hybridized DG implementation of the hypersonic aerothermoelastic system will 1) provide the physical accuracy necessary to capture complex physical features, 2) be free from any spatial and temporal inaccuracies or instabilities inherent to partitioned coupling procedures, 3) represent a transition to high-fidelity simulation methods for hypersonic aerothermoelasticity, and 4) enable efficient analysis of hypersonic aerothermoelastic effects on arbitrary geometries.

DEDICATION

To my wife Bethani, mi amor ahora and siempre

TABLE OF CONTENTS

DEDICATION	ii
LIST OF TABLES	vi
LIST OF FIGURES	vii
CHAPTER	
I. INTRODUCTION	1
II. BACKGROUND	5
2.1 Hypersonics	5
2.1.1 Hypersonic regime	5
2.1.2 Hypersonic modeling challenges	7
2.2 Multiphysics coupling	10
2.2.1 Partitioned approaches	12
2.2.1.1 Staggered approach	13
2.2.1.2 Fixed point approach	16
2.2.1.3 Partitioned conclusion	18
2.2.2 Monolithic approach	18
2.3 Hypersonic aerothermoelasticity modeling	21
2.4 Hypersonic aerothermoelasticity proposed approach	25
2.4.1 Discontinuous Galerkin	26
2.4.2 Hybridized discontinuous Galerkin	27
III. APPROACH	32
3.1 Balance equation	32
3.1.1 Hybridized DG discretization	33
3.1.1.1 Finite element triangulation	33
3.1.1.2 Finite element spaces	34
3.1.1.3 Inner products	35
3.1.1.4 Discretization	35

3.1.1.5	Numerical flux	36
3.1.1.6	Boundary conditions	37
3.2	Fluid domain	38
3.2.1	Continuous equations	38
3.2.2	Non-dimensionalization	40
3.2.3	HDG stabilization	41
3.3	Solid domain	42
3.3.1	Continuous equations	42
3.3.2	Non-dimensionalization	43
3.3.3	HDG Stabilization	45
3.4	Thermal domain	45
3.4.1	Continuous equations	45
3.4.2	Non-dimensionalization	46
3.4.3	HDG stabilization	46
3.5	Coupling	46
3.5.1	Arbitrary Lagrange-Euler	47
3.5.2	Continuous equations	49
3.5.3	Common time and length scales	53
3.5.4	Hybridized DG discretization	54
IV.	IMPLEMENTATION	55
4.1	Nonlinear solver	55
4.1.1	Static condensation	56
4.1.2	Parallelization	57
4.1.3	Automatic differentiation	59
4.2	Linear solver	62
4.3	Multiphysics considerations	62
4.3.1	Physical domain	62
4.3.2	Equation systems	63
4.4	Temporal	64
4.4.1	Time integrators	65
4.4.2	Pseudo-Transient Continuation	66
V.	RESULTS	70
5.1	Methodology	70
5.2	Verification	71
5.2.1	Fluid	72
5.3	Linear elasticity	78
5.4	Thermal	83
5.5	Hypersonic aerothermoelasticity on a cylinder	86
5.5.1	Problem description	87

5.5.2	Solution procedure	90
5.6	Hypersonic aerothermoelasticity on a hollow cylinder	93
5.6.1	Problem description	93
VI.	CONCLUSION	97
	REFERENCES	100

LIST OF TABLES

5.1	Degrees of freedom for each MMS mesh. The number of degrees of freedom must be multiplied by the number of components solved for by each of the physical equations to get an accurate degree of freedom count in each simulation. These values are 4, 4, and 1 for the fluid, solid, thermal equation systems, respectively.	72
5.2	Configuration for the compressible Navier-Stokes MMS verification study.	73
5.3	Configuration for the linear elasticity MMS verification study.	78
5.4	Configuration for the heat equation MMS verification study.	83
5.5	Configuration of the fluid domain. Values with an ∞ superscript denote freestream values which are used as an initial condition for the PTC procedure.	89
5.6	Configuration for the solid domain. A value of 1,000 K was chosen for the initial temperature of the solid for the PTC procedure.	90
5.7	Values of density, pressure, and temperature at the stagnation point of the cylinder's surface.	92
5.8	Configuration of the fluid domain for the hollow cylinder case. Values with an ∞ superscript denote freestream values which are used as an initial condition for the PTC procedure.	95
5.9	Configuration for the solid domain for the hollow cylinder case. A value of 3,000 K was chosen for the initial temperature of the solid for the PTC procedure.	95

LIST OF FIGURES

2.1	Two iterations of a staggered coupling approach for a FSI system. The (leading) fluid system is advanced from state u_f^n to state u_f^{n+1} . The updated fluid state is then transferred to the (lagging) structural physics system where it is used in the update from state u_s^n to u_s^{n+1} . Finally, the updated structural state is transferred to the fluid system for computation of the future fluid state u_f^{n+2}	15
2.2	Illustration of element and trace degrees of freedom in the hybridized DG method. Due to the discontinuous nature of DG methods, neighboring degrees of freedom u_1 and u_2 are independent. The hybridized DG method introduces trace variables \hat{u} defined on the common element interface (e.g. $\hat{u}_{1,2}$) which connect the solutions using flux-like terms and approximate Riemann solvers.	28
5.1	Manufactured solutions for the fluid velocity components v_1 and v_2 , respectively, evaluated at $t = 0.5s$	74
5.2	Solutions computed on the considered meshes for the v_1 fluid velocity component evaluated at time $t = 0.5s$	76
5.3	Solutions computed on the considered meshes for the v_2 fluid velocity component evaluated at time $t = 0.5s$	77
5.4	Convergence of the L^2 error between the manufactured and computed compressible Navier-Stokes solutions for the considered meshes computed at time $t = 0.5s$	78
5.5	Manufactured solutions for the solid displacement components u_1 and u_2 , respectively, evaluated at $t = 20s$	79
5.6	Solutions computed on the considered meshes for the u_1 solid displacement component evaluated at time $t = 20s$	81
5.7	Solutions computed on the considered meshes for the u_2 solid displacement component evaluated at time $t = 20s$	82
5.8	Convergence of the L^2 error between the manufactured and computed linear elasticity solutions for the considered meshes computed at time $t = 2.0s$	83

5.9	Manufactured solution for the temperature T evaluated at $t = 1.6s$	84
5.10	Solutions computed on the considered meshes for the temperature T evaluated at time $t = 1.6s$	85
5.11	Convergence of the average L^2 error between the manufactured and computed heat equation solutions for the considered meshes computed at time $t = 1.6s$	86
5.12	Geometry for the cylinder. The radius of the cylinder is 0.2 m.	88
5.13	Mesh employed for the simulation of the cylinder.	88
5.14	Steady-state solution for fluid-only variables on a cylinder subject to Mach 5 flow. The solution for Mach number (top left), non-dimensional pressure (top right) and non-dimensional density (bottom) are presented.	91
5.15	Steady-state solution for variables shared by the fluid and solid domain on a cylinder subject to Mach 5 flow. The solution for temperature (K) (left) and displacement (m) (right) are presented.	91
5.16	Mesh employed for the simulation of the hollow cylinder	94
5.17	Steady-state solution for temperature (K) (left column) and displacement (m) (right column) variables for with displacements scaled by 10 (top row) and undeformed (bottom row). The discontinuous nature of the solution spaces are seen in the deformed visualizations.	96

CHAPTER I

INTRODUCTION

There has recently been renewed interest in the design and analysis of vehicles which operate in the hypersonic, i.e. Mach 5 and beyond, flow regime. Hypersonic flight vehicles include hypersonic weapons systems, next-generation hypersonic boost and glide vehicles, and reentry aircraft. Still, greater knowledge of the hypersonic regime is needed to facilitate the design of next-generation hypersonic vehicles. Unfortunately, the lack of experimental data and considerable cost of hypersonic flight experiments has slowed the development of next-generation hypersonic vehicles. Modeling and simulation is positioned to fill existing knowledge gaps in the hypersonic regime and enable accurate and robust design and analysis of hypersonic vehicles.

The hypersonic regime is inherently a multiphysics regime dominated by the aerothermoelastic effects of fluid dynamics, structural mechanics, and heat transfer [85, 88] and is characterized by extreme heating and considerable fluid pressures which in turn produce large strains and stresses in a hypersonic vehicle. It is well known that these phenomena can have devastating effects on the vehicle and lead to onset flutter for even nominal temperatures [118, 33]. Further, deformation of the vehicle augments the surrounding flow which can in turn cause significant changes in flight performance. It is necessary to employ a simulation method that is able to accurately capture these complex multiphysics interactions observed in hypersonic aerothermoelasticity.

While simulation of the individual physics in aerothermoelastic problems have been well explored using a variety of methods, coupling of the hypersonic aerothermoelastic interactions requires special attention. By “coupling,” we denote the modeling of the interactions of the constituent physics which compose a multiphysics system. To this end, there exist two major approaches, namely partitioned and monolithic approaches [70]. Partitioned approaches model interactions through transfer of data resolved by single-physics solvers across physical boundaries. Partitioned approaches enjoy wide use in many physical simulation domains due largely to their relative ease of implementation [44]. However, certain partitioned methods suffer temporal instabilities due to the inherent leading and lagging of the constituent physics [54, 120] as well as spatial difficulties incurred by interpolation schemes between the non-conforming meshes [43] employed by the individual solvers. Monolithic approaches remedy the problems observed in partitioned approaches by instead solving a single equation system comprised of all physical equations on a single mesh. Due to this, monolithic methods can exhibit time-accurate results and incur no spatial interpolation errors. These qualities facilitate more rigorous analysis of discretizations of the physical equations. Still, the software implementation of monolithic methods is often more difficult than that of partitioned methods as development of a dedicated simulation software implementing the monolithic equation system and solver is often necessary.

Currently, hypersonic aerothermoelasticity simulation is widely pursued with low-fidelity or model reduction approaches on simple geometries [31, 85, 66] or at supersonic Mach numbers using a partitioned coupling scheme [118]. However, these methods are unable to facilitate accurate design and analysis of hypersonic vehicles for various reasons. First, low-fidelity methods preclude effective design and analysis of hypersonic vehicles due to physical inaccuracies stemming from

assumptions which are valid for only a few test geometries. Low-fidelity methods are also unable to generalize to arbitrary vehicle geometries and flow domains. Second, partitioned methods can introduce temporal and spatial inaccuracies in the multiphysics simulation. Spatial errors due to interpolation pose considerable challenges in ensuring physical requirements (e.g. conservation) are preserved. Attempts to remedy the temporal inaccuracies associated with partitioned methods can require rigorous tuning of inter-physics transfer schedules or implementation of fixed-point iterations with slow rates of convergence.

To address the problems associated with low-fidelity and partitioned approaches, we propose coupling the hypersonic aerothermoelastic equations using a monolithically-coupled hybridized discontinuous Galerkin (DG) [109, 111] method. The major effort of this work is the development of a hybridized DG code which implements the proposed monolithic hypersonic aerothermoelasticity system. DG methods are classes of finite element methods (FEM) which require the approximate solution space to be only piecewise continuous. Due to this property, DG methods are able to effectively resolve all three classes of second-order partial differential equations (PDEs) encountered in continuum physics. We note that each PDE class is represented in the hypersonic aerothermoelastic system. Further, a hybridized DG method is employed due to its ability to greatly reduce the number of globally coupled degrees of freedom compared to traditional DG methods [23]. This fact extends the reach of DG methods to large-scale problems commonly encountered in physical problems of interest [93] as hybridized DG methods are competitive with traditionally employed continuous Galerkin (CG) and finite volume methods (FVM) in terms of computational expense. Indeed, hybridized DG methods have already demonstrated success for multiphysics systems. Sheldon et. al [109] simulated monolithic fluid-structure interaction (FSI)

of incompressible flows with hyper-elastic materials. Their results were shown to be in excellent agreement with the FSI benchmark due to Turek and Hron [119]. Finally, simulation of the hypersonic aerothermoelastic system using a monolithic hybridized DG method is expected to 1) provide the physical accuracy necessary to capture all complex physical features, 2) be free from any spatial and temporal inaccuracies or instabilities inherent to the partitioned coupling procedure, 3) represent a transition to high-fidelity simulation methods for hypersonic aerothermoelasticity, and 4) enable efficient analysis of hypersonic aerothermoelastic effects on arbitrary geometries.

This dissertation is laid out as follows: Chapter II begins this work with a review of the physical considerations and modeling strategies for the hypersonic regime, an overview of commonly applied multiphysics coupling schemes, and a more detailed explanation of DG methods with special attention placed on the formulation of hybridized DG methods. Chapter III discusses the modeling approach and corresponding hybridized DG equation system employed for hypersonic aerothermoelasticity. Chapter IV provides description of the algorithms used to solve the monolithic hybridized DG system. Numerical experiments and their results are presented in Chapter V. We conclude with closing remarks in Chapter VI.

CHAPTER II

BACKGROUND

2.1 Hypersonics

2.1.1 Hypersonic regime

Hypersonic flows are typically defined as flows with a Mach number greater than 5. Hypersonic flows inherit much of the qualitative behavior of supersonic flows, generally defined as flows in the Mach 1-5 regime, including compressible flow effects, shock formation and interaction, and turbulence. In addition to these qualities, the hypersonic regime introduces the effects of extreme heating due to considerable convection near the surface hypersonic vehicle [85, 65, 41]. This heating results in the generation of large heat fluxes on the vehicle surface and the accumulation of large stresses in the flight vehicle via heat conduction[88]. Hypersonic vehicles must be engineered to be resilient to extreme temperatures and appreciable fluid pressures which cause structural bending and onset flutter [118, 64, 6]. It is common for hypersonic vehicles to be equipped with a thermal protection system on the vehicle surface to allow for controlled material degradation due to heating. At large hypersonic Mach numbers, the near-body heating is sufficient to cause the dissociation and ionization of the fluid molecules and, accordingly, the formation of a plasma layer near the vehicle.

As in lower speed flight regimes, a continuum assumption of all flow variables remains valid for much of the hypersonic regime. Due to this, the Navier-Stokes equations are employed as

the governing fluid equations in the hypersonic regime. We note, however, that the continuum assumption begins to fail for higher speed hypersonic flows due to a sharp increase in the mean free path of the fluid particles leading to a significant decrease in fluid density. Physical descriptions of these low-density flows must make use of the kinetic theory of particles. In this work, we restrict attention to the regime of hypersonic flows where the continuum assumption is valid.

Due to the devastating effects hypersonic heat loads are able to impose on the structure of a vehicle [38, 84], hypersonic aerothermoelasticity analyses may consider all forms of heat transfer, i.e. conduction, convection, and radiation, to ensure their accuracy, with the majority of the aerothermal heating occurring on the nose and leading edges of the vehicle [127]. Conduction and convection are governed by Fourier's law of conduction and Newton's law of cooling, respectively. Radiative heat transfer can be significant in the hypersonic regime and depends strongly on the material properties of the vehicle surface. We note that radiation evolves proportional to the quartic power of flow and surface temperatures. Similar to the Navier-Stokes equations, the governing equations for each form of heat transfer rely heavily on the continuum assumption of the fluid and solid.

Structural deformation and bending of a vehicle due to large fluid pressure and internal thermal strains and stresses is a hallmark of the hypersonic regime. The evolution of the structural displacements are governed by the typical elastodynamic equations. The elastodynamic equations are a differential statement of Newton's second law of motion written in terms of the temporal evolution of body stress due to strains and external loads. The stress in the body is related to solid displacement via the internal strain which is itself proportional to the spatial gradient of the solid displacements. Still, the relationship between stress and strain is material dependent. For

small applied loads, this relationship is often taken to be linear, causing the material to be assumed elastic. However, for stress-strain relationships depending on the displacements, nonlinearities in the governing equations are introduced and plastic deformation is permissible.

Physics-based high-fidelity modeling of the hypersonic regime is necessary to better understand the complex interactions of hypersonic aerothermoelasticity and to facilitate effective and reliable design and analysis of hypersonic vehicles. It is necessary to model the flow regime, structural deformation, and heat transfer in a coupled simulation. Unlike lower speed flight regimes, the hypersonic regime is plagued by a considerable lack of experimental flight data. High-fidelity modeling is positioned to fill in the knowledge gaps for the hypersonic regime. Still, high-fidelity is not free of challenges. We now review some modeling concerns for the hypersonic regime.

2.1.2 Hypersonic modeling challenges

Modeling of high-speed flows has been well-explored in the literature. However, modeling of the aerothermoelastic effects which dominate the hypersonic regime has historically received little attention [85]. Aerothermoelasticity modeling is comprised of unique challenges which can limit physical accuracy and computational efficiency of existing modeling approaches.

In many practical supersonic and lower Mach number hypersonic flow modeling applications, an ideal gas assumption is employed as the equation of state to relate fluid pressure, density, and temperature and ultimately close the Navier-Stokes equations. However, extreme heating in the hypersonic regime may necessitate the use of real gas models to better capture the temperature dependence in the equation of state. There exist many real gas models developed under assumptions of the gas under consideration, complicating the choice of a real gas model. Further, modeling

real gas effects in hypersonic flows is not a straightforward endeavor and may require special consideration of flight conditions and fluid properties.

The hypersonic regime is comprised of considerable shock layers, shock interactions, and turbulence. Modeling of these phenomena requires special attention on grid-based methods. Shocks are near-discontinuous changes in flow properties on the order of nanometers, often several orders of magnitude smaller than the physical cell size. Models for shock capturing and turbulence must be introduced to resolve their effects. Many methods have been introduced to increase shock resolution on grids, including near-shock grid refinement and shock-capturing methods commonly based on the addition of artificial diffusion to the viscous stress term in the Navier-Stokes equations.

It is necessary to model the effects of turbulence at the point of laminar-turbulence transition. Turbulence evolves across many turbulent length scales, the smallest of which are often not resolvable on even well-refined grids. Similar to shock capturing methods, turbulence has also been investigated at considerable length from which three main approaches for turbulent flow simulation have arisen, listed in order of increasing physical fidelity and computational cost: Reynolds-Averaged Navier-Stokes (RANS), Large Eddy Simulation (LES) and Direct Numerical Simulation (DNS). While DNS simulates turbulence using high-order numerical methods on very fine grids, RANS and LES impose models to resolve fine turbulent flow features with comparatively less computational expense.

The modeling of real gas effects, shocks, and turbulence can add additional nonlinearities to the augmented Navier-Stokes equations causing accurate and stable modeling of these effects to be greatly complicated. In addition to these difficulties, the hypersonic speeds of the fluid can greatly reduce the time step needed for stable time evolution in unsteady solutions, potentially causing a

hypersonic flow simulation to require a large number of solver iterations across prohibitively-many time steps.

The extreme heating prevalent in the hypersonic regime necessitates the modeling of heat transfer of all kinds including conduction, convection, and radiation. Each form of heat transfer introduces unique complexities. Conduction modeling requires knowledge of temperature-varying material properties often determined by experiment. Convection modeling requires determination of a convection coefficient which may depend nonlinearly on flow properties. Radiation modeling may require the computation of viewfactors between grid cells. Further, radiation modeling introduces nonlinearities proportional to the quartic power of temperature.

The Navier-Stokes equations are notorious for their varied behavior in different areas of the fluid domain. For example, near-body flow demonstrates strongly elliptic behavior due to the dominance of the viscous stress term compared to the convective term. Far-field flow is strongly hyperbolic as viscous effects are minimal and the nonlinear convective term dominates. The Peclet number Pe is a non-dimensional number defined as the ratio of the convection transport rate to the diffusive transport rate and may be used as an indicator of the dominating flow effects and thus the dynamical behavior of the Navier-Stokes equations. Resolution of flows across a wide range of Pe numbers is a necessary capability of a high-fidelity aerodynamics model.

The nonlinear nature of the three physical systems causes individual physical simulation to be a considerable challenge in itself. Coupling of the equations to produce a unified, physically accurate system adds further complexity. To this end, we continue with a review of multiphysics coupling methods.

2.2 Multiphysics coupling

The hypersonic aerothermoelasticity system is inherently a multiphysics system. In this section we present an overview of multiphysics coupling approaches providing motivation and drawbacks for each approach. Throughout this section, we make use of fluid-structure interaction (FSI) as an example problem and provide some discussion about the application of the various multiphysics coupling techniques to the FSI problem. We use FSI coupling approaches as examples due to the availability of FSI studies in the literature and due to the similarities between FSI and aerothermoelasticity for which fewer studies exist.

FSI approaches are concerned with the simulation of the coupled interactions of a malleable structure subject to the effects of dynamic fluid. In FSI approaches, it is assumed that the structure is not rigid and will deform under the pressure loads acting on the structure's surface. However, deformation of the fluid domain must be modeled in an appropriate way since the deformation of the solid displaces the nearby fluid. It will later be discussed that the method of fluid and structural deformation may depend on the employed coupling approach.

Throughout this section we use the notation \mathbf{u}_f and \mathbf{u}_s to refer to the fluid and structural solution, respectively. We denote the fluid-structure coupling interface as Γ . Certain coupling methods will make use of a third solution defined on Γ which we denote by \mathbf{u}_Γ . We use \mathbf{r}_f , \mathbf{r}_s , and \mathbf{r}_Γ to refer to the discrete residuals of appropriate governing equations defined on the fluid, solid, and coupling interface, respectively. Finally, despite the following schemes being presented in the context of FSI, generalization to other multiphysics systems or multiphysics systems which consider three or more physics is straightforward.

Choosing an appropriate coupling approach frequently requires consideration of problem-dependent information and the time budget allotted for development of the coupling scheme. Analysis of the Jacobian of the residual for a multiphysics problem can provide insight to determine the coupling scheme which best fits the goals of the practitioner. We illustrate this Jacobian for the FSI system as follows:

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_{f,f} & \mathbf{J}_{f,s} \\ \mathbf{J}_{s,f} & \mathbf{J}_{s,s} \end{bmatrix} \quad (2.1)$$

In general, the entries of the Jacobian of a multiphysics problem are given by $J_{ij} = \frac{\partial r_i}{\partial u_j}$. We note that the diagonal elements of the Jacobian are the sensitivities of the individual physics with respect to their physical variables. The off-diagonal terms correspond to the inter-physics couplings. It is apparent that if the off-diagonal terms of (2.1) are of negligible magnitude compared to the diagonal terms, the coupling is weak. Multiphysics systems which possess weakly-coupled physics may be able to disregard the coupling of such physics entirely to realize a time savings with nominal loss in accuracy.

For multiphysics problems similar to FSI which perform coupling through the transfer of physical data on the coupling boundary, a secondary formulation is possible. It is possible to decompose the multiphysics problem into subproblems on the respective domains plus a new problem defined on the multiphysics boundary. In this case, the multiphysics Jacobian is an “arrow-like” matrix. In the case of FSI, this Jacobian is given by

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_{f,f} & \mathbf{0} & \mathbf{J}_{f,\Gamma} \\ \mathbf{0} & \mathbf{J}_{s,s} & \mathbf{J}_{s,\Gamma} \\ \mathbf{J}_{\Gamma,f} & \mathbf{J}_{\Gamma,s} & \mathbf{J}_{\Gamma,\Gamma} \end{bmatrix} \quad (2.2)$$

As mentioned in Chapter I, multiphysics coupling schemes can be broadly categorized as partitioned or monolithic schemes. We also make a distinction between two classifications of partitioned coupling schemes: staggered and fixed point coupling schemes, sometimes referred to as loosely and tightly coupled partitioned schemes in the literature. A major distinction between partitioned and monolithic schemes can be seen using the multiphysics Jacobians. Typically, partitioned schemes model or omit the effects of the coupling Jacobians, while monolithic methods frequently compute the full Jacobians.

Still, much research has been performed using both approaches from both classes. For example, partitioned [42, 14, 35, 44, 45, 73, 121] and monolithic [109, 111, 108, 63, 35, 5] approaches have been extensively explored to model FSI. To motivate the use of a monolithic coupling approach in this work, we briefly provide some overview of partitioned and monolithic coupling approaches, placing special consideration on the application of such methods for FSI coupling.

2.2.1 Partitioned approaches

Partitioned coupling approaches involve the coupling of various single-physics solvers through transfer of solution data across shared physical boundaries. Partitioned approaches are especially common in the multiphysics literature compared to monolithic methods due to their ability to employ existing single-physics simulation codes which can be specifically tailored to resolve complex effects in the single-physics system. Due to this, in a partitioned method, it is only

necessary to derive and implement the coupling procedure. Coupling is performed by the transfer of physical data on the coupling interface. Data transferred from one domain is used to enforce boundary conditions on the other.

A difficulty encountered in partitioned coupling approaches is the need to develop accurate interpolation methods for physical data transferred along a coupling interface since the locations where the solutions are computed may not conform between the meshes on the coupling interface. Erroneous interpolation schemes along non-conforming meshes may lead to computation of unphysical solutions [43]. This problem is of special importance when the underlying physics relies on the transferred data being conserved, e.g. the transfer of heat flux between two heated domains. Interpolation schemes may not be able to ensure conservation of such data and may cause spurious creation and deletion of transferred data. Methods to combat these issues for FSI problems are presented in [43], though the methods can not guarantee conservation. Beyond this difficulty, it is readily observed that ensuring the stability of a partitioned scheme is also itself a challenge [45].

2.2.1.1 Staggered approach

In a staggered coupling approach, the transient simulation of a multiphysics system is performed by sequentially computing the solutions to the single-physics problems in a prescribed schedule. Within a time step the execution of a single-physics solver may employ boundary conditions on the coupling interface computed by a previous single-physics solver within the current or previous time step. An area of research for staggered schemes is the development of effective staggered transfer schedules and determination of the necessary data to transfer on the coupling interface. In the case that the evolution of one physical domain occurs on a smaller time scale than another domain, a

characteristic transfer time may also be chosen such that the more quickly evolving physical system may be evolved for many time steps before transferring boundary data to the slower physical system. Thus, many possible transfer schedules for a multiphysics problem employing a staggered coupling method may exist. The interested reader may refer to [45] for description of the various operations which define staggered approaches.

For the case of the FSI system, a seemingly valid transfer schedule at a time step n may be the following:

1. Evolve the fluid state from u_f^n to u_f^{n+1}
2. Transfer the fluid pressures on the coupling interface to the structure to be imposed as boundary conditions
3. Evolve the structure state from u_s^n to u_s^{n+1}
4. Transfer the structure's displacement and velocity on the coupling interface to the fluid
5. Update the fluid mesh according to the structure's displacement
6. Impose the structure's velocity as the fluid's velocity on the coupling interface

However, in the case of an incompressible fluid, staggered schemes for FSI are unstable [54] and require additional stabilization methods, e.g. an under-resolution method [86]. Indeed, staggered approaches for general problems can exhibit considerable instabilities [45]. It is observed that the instability of a staggered FSI approach grows quickly for large fluid-structure mass ratios [54]. Compressible flows, however, require a different stability analysis. It was shown in [120] that the stability of FSI approaches using compressible flows scale is a function of the time step.

While the staggered approach has the attractive property of being perhaps the simplest partitioned coupling approach, it suffers from certain drawbacks. First, a staggered approach may introduce numerical instabilities and physical inconsistencies [45]. Second, it is not apparent if in

general a staggered approach associates with a convergence property for the coupled solution on physical boundaries [45]. Since the single-physics solvers are executed sequentially, the staggered approach gives rise to “leading” and “lagging” physics systems, as opposed to all participating physics evolving simultaneously in time. As demonstrated in Figure 2.1, a leading physics system will transfer physical variables computed at a future time point to a coupled solver at a previous time point. Besides this approach potentially introducing temporal inaccuracies, it is unclear if this approach will converge to a physically valid state for a chosen time step. Indeed, choice of an appropriate time step may require additional analyses which consider the strength of the coupling determined by the Jacobian of the multiphysics system. Finally, it is worth noting that methods for combatting the temporal inaccuracies imposed by partitioned methods have been investigated [44]. Methods which adapt the time step of the fluid and structural solvers have also received attention [83, 39].

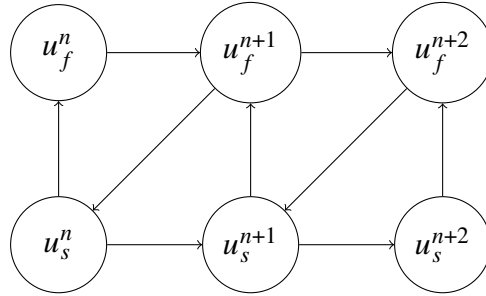


Figure 2.1: Two iterations of a staggered coupling approach for a FSI system. The (leading) fluid system is advanced from state u_f^n to state u_f^{n+1} . The updated fluid state is then transferred to the (lagging) structural physics system where it is used in the update from state u_s^n to u_s^{n+1} . Finally, the updated structural state is transferred to the fluid system for computation of the future fluid state u_f^{n+2} .

2.2.1.2 Fixed point approach

Perhaps a more mathematically rigorous approach to implement a partitioned coupling approach is through a fixed point partitioned approach. Fixed point partitioned approaches solve for the boundary state \mathbf{u}_Γ at each time step by using an iterative fixed point procedure. Given an initial guess \mathbf{u}_Γ^0 to the coupling interface state at a given time step, fixed point approaches consider an iterative mapping for \mathbf{u}_Γ given by

$$\mathbf{u}_\Gamma^{n+1} = \Phi \left(\mathbf{u}_f^n, \mathbf{u}_s^n, \mathbf{u}_\Gamma^n \right) \quad (2.3)$$

where Φ is the fixed point mapping to be chosen which enforces coupling conditions on the coupling interface. Of course, the convergence of the coupling interface state \mathbf{u}_Γ requires that the mapping Φ be a contraction mapping which may be difficult to formulate for general problems. Convergence of (2.3) may not even be achievable for certain choices of Φ . Still, even with a well-posed fixed point scheme for a multiphysics problem, convergence to a solution within a reasonable tolerance may require prohibitively many iterations and impose a large computational burden. Indeed, many fixed point approaches require the evaluation of the subdomain solutions at every iteration. This problem is complicated further by the frequent unavailability of the Jacobian information expressed in (2.1) which could be used to expedite the convergence of (2.3) and (2.4). Due to this, the number of iterations required to achieve convergence can grow prohibitively large and computationally expensive since each iteration requires a simulation of all participating single-physics solvers.

To alleviate the computational cost of fixed point methods, it is common to employ under-relaxation techniques for one or all physical disciplines which expedite the fixed point procedure [86, 108, 61, 73, 9]. The under-relaxed form of (2.3) is given by

$$\mathbf{u}_{\Gamma}^{n+1} = \omega \Phi \left(\mathbf{u}_f^n, \mathbf{u}_s^n, \mathbf{u}_{\Gamma}^n \right) + (1 - \omega) \mathbf{u}_{\Gamma}^n \quad (2.4)$$

where ω is an appropriately chosen relaxation parameter. Approaches to determine ω for a fixed point FSI approach was pursued in [73]. The investigated methods included steepest descent and Aitkens Δ^2 method [2].

Determining an appropriate fixed point mapping Φ for a multiphysics problem is the main effort for the fixed point approach. To this end, it is common to employ domain decomposition (DD) techniques [117, 61, 89, 9] to derive an effective fixed point scheme since the extension of DD methods to multiphysics problems is natural. DD methods traditionally decompose a single-physics problem defined on a single domain into subproblems defined on either overlapping or non-overlapping subdomains plus a new problem defined on the interfaces of the subdomains. It is straightforward to apply DD methods to multiphysics problems by taking the subdomains in a DD formulation to be the physical subdomains on which a single-physics problem is solved. This approach results in the subdomain interfaces becoming multiphysics coupling interfaces. The resulting equations on the interfaces of the subdomains define the coupling conditions for a multiphysics problem. Many DD methods require iteration of the resulting coupled problem to ensure convergence.

FSI methods frequently employ a DD-inspired Dirichlet-Neumann coupling on the multiphysics boundary [108]. In this coupling approach, the solid and fluid velocities are imposed to be equal (the Dirichlet condition) and the tractions along the outward-pointing interface normals are required to balance (the Neumann condition). It is apparent that the displacement of the structure should displace the nearby fluid which may require re-meshing the fluid domain each iteration.

Partitioned coupling of the FSI problem can lead to unexpected physical effects. An example of this is the added mass effect where fluid mass is spuriously created or dissipated in incompressible flows [54, 120]. The impact of the added mass effect is related to the ratio of fluid and structural mass. Compressible flows also exhibit an added mass effect. However, it was found that the impact of the effect was related to the magnitude of the time step for transient flows [120].

2.2.1.3 Partitioned conclusion

Finally, partitioned methods are desirable due to their ability to employ existing single-physics simulation codes leading to their relative ease of implementation compared to monolithic methods. Staggered methods are perhaps the most computationally inexpensive partitioned approach. However, it is unclear if in general a staggered method is stable or accurate. Fixed point methods, on the other hand, are generally equipped with a notion of convergence, but can require an excessive number of iterations to arrive at a physical solution each time step. Still, common to both methods is the difficulty of developing accurate interpolation schemes between non-conforming meshes. This problem is especially significant for multiphysics problems which must maintain conservation of physical data transferred across the coupling boundary. Finally, for certain multiphysics systems employing a partitioned method, a fixed point approach may be preferred to a staggered approach since fixed point methods are equipped with a notion of convergence, e.g. by measure of the L^2 error $\left\| \mathbf{u}_\Gamma^n - \Phi \left(\mathbf{u}_f^n, \mathbf{u}_s^n, \mathbf{u}_\Gamma^n \right) \right\|$.

2.2.2 Monolithic approach

Monolithic coupling approaches perform multiphysics coupling by formulating the multiphysics system as a single equation system to be solved on a single mesh. In the monolithic approach, all

coupling is preformed in the continuous domain, prior to spatial and temporal discretization. Due to this, monolithic methods are free from much of the error generated by partitioned methods. Unlike a staggered partitioned approach, monolithic methods impose no inaccuracies in time evolution. No fixed point iterative scheme defined on the coupling interface is needed to ensure coupling convergence. Monolithic approaches often evolve on a single computational grid, eliminating the need for devising complex inter-mesh interpolation schemes.

A major drawback of monolithic methods noted in the literature is their lack of modularity to employ arbitrary single-physics solvers for solution of the multiphysics system [85, 44]. However, multiphysics coupling performed within a single simulation code has certain performance advantages. For example, monolithic methods perform all inter-physics data transfer in memory, while partitioned methods may require file-based data transfers. Due to the memory efficiency, monolithic methods are expected to better equip multiphysics systems for exascale computing [70].

Many monolithic approaches of FSI have been developed [109, 111, 63, 35, 5, 57]. Besides having to develop appropriate fluid and structure solvers, an additional difficulty for monolithic FSI approaches is accurately formulating the deformation of the fluid domain. While partitioned solvers may simply re-mesh the fluid domain each coupling iteration, monolithic approaches resolve all physics on a single mesh. Due to this, an additional deformation field is included in the fluid equations to compute the deformation of the fluid domain.

A difficulty with this approach, however, is that the fluid and structure equations evolve in differing reference frames. To remedy this problem, an Arbitrary Lagrange-Euler (ALE) approach [37] is often employed to unify the fluid and structure equations into a common reference frame. ALE approaches have found many applications including monolithic coupling of FSI [109, 35,

5, 57, 55], adaptive mesh refinement and optimization [4], and mesh motion under prescribed transformations [93].

In addition to the burden of developing a new simulation code for a monolithic multiphysics system, additional difficulties arise in determining appropriate methods to solve the monolithic system. In particular, a difficulty for monolithic solvers is the devising of appropriate preconditioners which generally must be formulated specifically for the multiphysics problem [70]. Studies for monolithic FSI preconditioners approaches have been investigated [82, 57].

With respect to FSI, A universal benchmark for incompressible, laminar FSI problems was introduced by Turek and Hron [119]. The authors also presented a monolithic FSI coupling approach with attention placed on biomechanics applications [63]. This work formulated the FSI system using a finite element method using an ALE formulation to make the fluid and solid reference frames consistent. The resulting nonlinear system was solved using a damped Newton's method with a line search to accelerate convergence. A multigrid solver was used to solve the resulting Newton system. While the authors note the accuracy and robustness of the scheme, they stress the need for efficient and parallel solvers to expedite its solution.

An overview of partitioned and monolithic FSI approaches can be found in [108]. A comparison the performance of a partitioned FSI solver and a monolithic FSI solver was performed in [35]. The authors found that, while the partitioned method did sometimes outperform the monolithic method, certain partitioned test cases did not converge. Recently, a mixed partitioned-monolithichic coupling approach for low Mach thermal FSI approaches was pursued in [60] where two monolithically coupled systems were coupled with a partitioned approach. The method was found to be robust for the considered experiments. Finally, a monolithic coupling approach was

employed to implement a solver for a capsule/parachute system [16]. Notable in this work is the development of a coupling graph which details the coupling and dependencies between the various physics models for computation of the physical variables and their adjoint variables.

2.3 Hypersonic aerothermoelasticity modeling

Hypersonic aerothermoelasticity modeling has witnessed a resurgence in the recent literature, with a major focus being the coupling procedures for the three physical disciplines. Partitioned staggered coupling methods are the prevailing hypersonic aerothermoelasticity approaches, with many authors replacing certain physical equations with low-fidelity physics-based models, empirical models, or reduced order models to achieve a cost savings.

A review of research for hypersonic aerothermoelasticity is presented in [85]. Therein the authors note three potential avenues for simulation of hypersonic aerothermoelasticity: low-fidelity approaches, high-fidelity modeling, and reduced-order modeling using surrogate modeling techniques. They note the need for staggered approaches to perform two-phase aeroelastic-aerothermal coupling as opposed to independent three coupling of fluid, thermal, and solid dynamics. It was found that an aeroelastic-aerothermal coupling better resolved the aerodynamic heating and solid temperature and displacement response for flows over a skin panel. Still, the authors note the need for advancement of high-fidelity simulation methods to better understand the physical characteristics of the hypersonic regime.

Follow-up work to [85] is presented in [31] where a staggered partitioned approach for hypersonic aerothermoelasticity across a flat plate is employed for two-way aeroelastic-aerothermal coupling. In this lower-fidelity approach, approximations for the aerodynamic heating are obtained

using Eckert's reference enthalpy method which provides a low-fidelity model of the thermal boundary layer heating and is used to determine an approximate aerodynamic heat flux. This enthalpy method was determined from incompressible flow data and augmented to compensate for compressible flow effects. The aeroelastic model employed a finite element formulation of the equations of motion. The aeroelastic solution was represented a a combination of assumed sinusoidal modes plus a quadratic polynomial. The final coupled system was solved on coarse grids using an explicit time integrator.

The fluid, structural, and thermal dynamics in hypersonic flow evolve at disparate time scales, potentially varying by several orders of magnitude. The differences in these time scales necessitate the use of long-time simulations to accurately capture the multiphysics effects for a hypersonic vehicle [65]. Due to this, much of the present research focuses attention on computationally efficient methods to greatly the reduce simulation time. These methods often achieve reduced computational time at the sacrifice of accuracy. It is common in the literature to replace high-fidelity simulation method with low-order approximations and empirical relationships derived from flow states of questionable applicability to the hypersonic regime.

Various empirical models are employed to efficiently obtain reasonable physical heat flux and pressure distributions on a vehicle surface. Among the commonly applied empirical pressure models are Newtonian Impact Theory, Classical Piston Theory, and Local Piston Theory. Culler et. al [32] employed Eckert's reference enthalpy method to predict surface heating on a hypersonic panel. We note, however, the application of the presented empirical models depends heavily on the geometry, often finding the most accurate application on flat plates and panels. Still some work has been performed using high-fidelity discretizations for each of the physics [126, 89].

Reduced order models (ROMs) with a variety of formulations have become a mainstay of the hypersonic aerothermoelasticity literature [66, 40, 41, 30, 71] in attempt to greater alleviate the computational burden in simulation. ROMs are a class of surrogate models which are often constructed using data generated from a full order model. With respect to hypersonic aerothermoelasticity, ROMs are commonly employed for the fluid simulation portion and are frequently constructed using proper orthogonal decomposition. A machine learning approach is presented in [122] to expedite aerothermal heating prediction. An intrusive Least-Squares Petrov-Galerkin [15] has also been pursued for high-fidelity hypersonic model reduction [11].

Partitioned coupling methods are near universally applied for the resolution of multiphysics hypersonic flows. Much of the literature cites the robustness of partitioned methods for interchange of the constitutive single-physics solvers as a major motivation for their use. Some work employing DD techniques has been employed [9, 89]. To the author's knowledge, only a single monolithic method has appeared in the literature. However, only hypersonic aeroelasticity was studied in this monolithic approach [71].

Few studies investigate the effects of the hypersonic aerothermoelastic system on a full vehicle. Instead, hypersonic panels [64, 31, 116] and control surfaces [104] are the prevailing domain finding application. The investigation of hypersonic panels can greatly ease investigation for a variety of reasons including 1) the availability of structural modes for determining structural deformation and time scales (some studies derive a structural solution entirely from these modes), 2) simplified conduction modeling (e.g. only modeling through-thickness conduction), 3) simplification of fluid-solid interface modeling via application of empirical models for surface heating and pressure

distributions, and 4) reduced complexity in constructing accurate ROMs due to the simplicity of the geometry.

Work has been done to integrate hypersonic aerothermoelasticity into design of experiments frameworks. These applications include design optimization [113], trajectory optimization [29], uncertainty propagation [76, 75], and the devising of control laws [123, 79]. Work for the formulation of similarity scaling laws for experimental validation using, e.g., wind tunnels has been pursued [64, 13]. The convergence of aerothermoelasticity methods has also been analyzed using both finite element and finite volume discretizations for the fluid equations [89].

A partitioned approach for effective simulation of aerothermoelasticity in the supersonic and hypersonic regime is presented by Tran et al. [118]. The presented work builds on previous work performed in [77] in which a three-field formulation for aeroelasticity simulation was developed. The fields of consideration were the fluid, structure, and fluid mesh motion fields. This work extends this previous work to include thermal contributions as a fourth field, motivated by observed onset flutter induced by aerothermal heating in a wing subject to $\text{Ma } 2$, 500°F flow.

This work formulates the fluid momentum using an ALE formulation to allow the fluid domain to experience deformation due to the displacement of the structure. As is typical in the literature [109, 111, 77], the mesh motion is described via a linear elastostatic equation. The stiffness of the mesh is purely fictitious, but is able to vary with time. The structural equation is a nonlinear elastodynamic equation. This equation is coupled to the thermal unknowns through addition of a $\mathbf{C} (T_S - T_{\text{ref}})$ term where \mathbf{C} is the thermal coupling matrix and T_S and T_{ref} are the structural and zero thermal stress temperatures, respectively. Transient conduction is the only heat transfer mechanism considered. At the fluid-structure interface, the boundary displacement and velocities are required

to match the structural displacement and velocities. Similar conditions for the fluid and structural temperature, heat flux, and stresses are applied. All equations are non-dimensionalized.

This work formulates the coupling of the FSTI using a partitioned staggered method. Initial conditions for fluid, structural, mesh, and thermal equations are applied. The coupling procedure is outlined as 1) the fluid solution is advanced 2) aerodynamic forces and heat fluxes are transferred to the structure 3) the structural temperature is advanced using the received heat fluxes 4) the temperature field is transferred to the structure 5) the structural displacement is computed using the fluid and thermal forces and 6) the fluid mesh is updated using the structural displacements. The fluid equation is solved with a finite volume method where, notably, the diffusive fluxes are approximated using a weak formulation. The heat and structural equations employ a finite element scheme. Finally, the mesh motion uses a torsional spring formulation due to [42].

Still, the problem of inter-mesh data transfer remains. For non-conforming meshes, conservation of transferred data appears to be enforced weakly. The method is tested on a section of F-16 wing and a flat plate. In both cases, the free stream flow is set to be Ma 2. Thus, no hypersonic flows are considered. Still, in the case of the flat plate, flutter is observed due largely to the contribution of aerothermal heating. However, the accuracy of the method is questionable in some regions. For example, in the simulation of the F-16 wing, the authors note, “skin temperature can be over-predicted if the heat transfer model accounts for the structure only”.

2.4 Hypersonic aerothermoelasticity proposed approach

Implementation of the hypersonic aerothermoelasticity system using a monolithic simulation method represents a significant departure from the prevailing modeling techniques in the presented

literature in terms of computational accuracy and expense. A monolithic coupling approach is expected to be more accurate than the commonly applied staggered methods. Further, simulation of physics is performed using a high-fidelity approach. However, to simulate monolithic hypersonic aerothermoelasticity, it is necessary to consider a discretization approach that is able to resolve the underlying physical equations. It is for this reason we consider DG methods.

2.4.1 Discontinuous Galerkin

DG methods are classes of high-order finite element methods which do not enforce continuity of the solution on element interfaces. Instead, DG methods enforce continuity of the solution within each element and model flux transfer on the element interfaces. A consequence of this is that DG methods frequently possess significantly more unknowns than traditional finite element methods due to solution nodes on the element interfaces being doubly defined. DG methods have also referred to as high-order methods since the order of the finite element solution spaces can be arbitrarily large and can be adapted during a simulation [58, 59].

DG methods were first devised by Reed and Hill for simulation of neutron transport [102] and have since been applied to many problems of interest. DG methods extend the application of finite element methods beyond what traditional continuous finite element methods typically allow while maintaining the capability of continuous finite element methods to accurately simulate elliptic problems [112]. Of particular interest, DG methods have been successfully applied to hyperbolic and fluid flow problems of various kinds [7, 62, 107] including entropy-stable [17, 18, 20] and entropy-bounded [80] solutions, a feat not typically shared by traditional continuous finite element methods.

2.4.2 Hybridized discontinuous Galerkin

DG methods notoriously require the solution for significantly more degrees of freedom compared to continuous finite element and FVM formulations, limiting their application to problems of academic interest. Hybridized DG methods were introduced in 2009 by Cockburn in [26] and explored further in [23] for the solution of elliptic PDEs. The methods were found to greatly reduce the number of globally coupled degrees of freedom and reduce the computational cost while maintaining high order accuracy. This computational cost savings offered by hybridized DG methods has attracted considerable research since their inception.

Hybridized DG methods reduce the computational cost compared to traditional DG methods by considering additional unknowns defined on element interfaces called *trace variables* along with additional equations which control the flux on the interfaces. Figure 2.2 illustrates the configuration of the degrees of freedom in the hybridized DG method. Due to the discontinuous nature of the element unknowns, a global system can be formulated in terms of only the trace unknowns through employment of static condensation. This global system is found to be a greatly reduced system of equations while maintaining the sparsity of the original system. Due to this, the hybridized DG method makes DG methods considerably more competitive with traditional mesh-based PDE solution techniques.

A second attractive property of hybridized DG methods is that they possess a superconvergence property which causes that the convergence of a hybridized DG method to increase by an order. Superconvergence is achieved via a problem dependent post-processing which is performed on individual elements and can be executed in parallel. However, it is necessary to derive such a

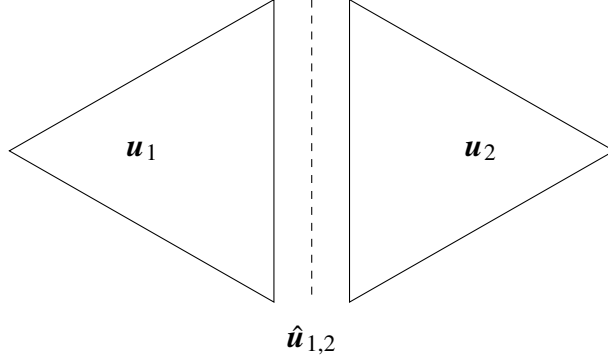


Figure 2.2: Illustration of element and trace degrees of freedom in the hybridized DG method. Due to the discontinuous nature of DG methods, neighboring degrees of freedom u_1 and u_2 are independent. The hybridized DG method introduces trace variables \hat{u} defined on the common element interface (e.g. $\hat{u}_{1,2}$) which connect the solutions using flux-like terms and approximate Riemann solvers.

post-processing technique for individual PDEs. Such techniques have been derived for various problems of interest [94, 95, 96, 24, 74].

Much research has been done to apply the hybridized DG method to many physical problems of interest which evolve on complex domains. In particular, hybridized DG methods have been extended to perform simulations for problems with feature overset meshes [68], mesh transformations [100, 97], adaptive meshing [125, 105, 81], non-conforming meshes [114, 98], and domain decomposition approaches [78, 106, 92, 49].

As mentioned, like continuous finite methods, hybridized DG methods maintain the ability to effectively resolve elliptic problems. The seminal works on hybridized DG methods focused on resolving second-order elliptic problems [26, 25]. Further has been performed to model structural deformation including hybridized DG methods linear [47, 24, 96] and nonlinear elasticity [47, 28, 115].

A distinguishing factor between continuous and discontinuous finite elements is the ability of the latter to compute high order solutions for hyperbolic [27] and advection-dominated problems. In particular, hybridized DG methods have demonstrated success in resolving the shallow water equations [8, 67], the incompressible [97, 59] and compressible [124, 90, 81, 51] Navier-Stokes equations. Magnetohydrodynamics problems consider an electromagnetic fluid and have also received treatment [92, 22]. Important to the present study is the simulation of hypersonic flows with a hybridized DG method [81].

Of particular interest in computational fluid dynamics is the resolution of shocks and the accurate simulation of turbulence. Shock-capturing methods have been developed for DG methods, predominantly by employing artificial viscosity approaches [53, 21, 101]. The simulation of turbulence with hybridized DG methods have been developed for both RANS [125] and LES [49] formulations. Many additional studies have also been performed to determine more advanced methods to model turbulence with application to hybridized DG methods [52, 48, 50, 91]. It has been shown that an implicit LES approach using a high order hybridized DG predicted turbulent transition over an airfoil with an order of magnitude fewer globally coupled degrees of freedom than a second-order FVM approach [49].

Hybridized DG method to simulate monolithic multiphysics systems. We note the seminal work performed by Sheldon et al. [109] to construct a monolithic FSI system using the hybridized DG method on a single mesh. In this approach, every element of the mesh was labeled as either a fluid element or a solid element. This labelling determined the physical equation system applied in the element. The work employed the incompressible Navier-Stokes equations for the fluid simulation and the elastodynamics equations with a hyperelastic stress tensor. The global degrees of freedom

for this work were later decreased by about 50% in a later study [111] through elimination of the globally coupled solid displacement degrees of freedom and by decreasing the polynomial order of the fictitious fluid mesh equations. Negligible degradation of solution accuracy was observed.

We now provide a demonstration of the ability of hybridized DG methods to greatly reduce the globally-coupled degrees of freedom compared to other DG methods, making hybridized DG competitive with continuous finite element and FVM methods. A more formal mathematical description of hybridized DG methods is presented in Ch. III. We consider the discrete residual $\mathbf{R}(\mathbf{u}, \hat{\mathbf{u}}) = \begin{bmatrix} \mathbf{r}(\mathbf{u}, \hat{\mathbf{u}}) & \hat{\mathbf{r}}(\mathbf{u}, \hat{\mathbf{u}}) \end{bmatrix}^T$ of the weak form of a PDE discretized via a hybridized DG method where $\mathbf{r}(\mathbf{u}, \hat{\mathbf{u}})$ and $\hat{\mathbf{r}}(\mathbf{u}, \hat{\mathbf{u}})$ represent the discrete residuals defined on the element and trace unknowns, respectively. We seek the solution $\mathbf{u}^*, \hat{\mathbf{u}}^*$ defined on the element and element trace, respectively, such that $\mathbf{R}(\mathbf{u}^*, \hat{\mathbf{u}}^*)$ vanishes. To this end, we apply Newton's method which results in the following linear system:

$$\begin{bmatrix} \nabla_{\mathbf{u}} \mathbf{r} & \nabla_{\hat{\mathbf{u}}} \mathbf{r} \\ \nabla_{\mathbf{u}} \hat{\mathbf{r}} & \nabla_{\hat{\mathbf{u}}} \hat{\mathbf{r}} \end{bmatrix} \begin{bmatrix} \delta \mathbf{u} \\ \delta \hat{\mathbf{u}} \end{bmatrix} = - \begin{bmatrix} \mathbf{r} \\ \hat{\mathbf{r}} \end{bmatrix} \quad (2.5)$$

To simplify the notation used in the following chapters, the linear system in (2.5) is typically written as

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \delta \mathbf{u} \\ \delta \hat{\mathbf{u}} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix} \quad (2.6)$$

Hybridized DG methods greatly reduce the Newton system in (2.6) by eliminating $\delta \mathbf{u}$ via static condensation. After employing static condensation, we can consider the following reduced global system written in terms of only the increments to trace unknowns $\hat{\mathbf{u}}$ with

$$\mathbf{K}\delta\hat{\mathbf{u}} = \mathbf{h} \tag{2.7}$$

$$\mathbf{K} = \mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{C} \quad \mathbf{h} = \mathbf{g} - \mathbf{C}\mathbf{A}^{-1}\mathbf{f}$$

We note that this reduced global system is sparse. After solving for $\hat{\mathbf{u}}$, the element increments \mathbf{u} are computed in an element-local procedure. Thus, the hybridized DG method exchanges the solution of a large linear system (2.6) for the solution of a considerably smaller linear system (2.7) and independent element-local updates which are trivially parallelized.

CHAPTER III

APPROACH

In this chapter we present the coupling approach for the hypersonic aerothermoelasticity system. The presented approach considers the coupled contributions and interactions of hypersonic flow, nonlinear structural deformations, and heat conduction. To this end, three separate PDEs are presented and their interactions on the fluid-solid interface are described. The constitutive PDEs are also non-dimensionalized through appropriate choice of physics-based scaling factors. Non-dimensionalization is often necessary for physics simulation codes due to the wide range of magnitudes of physical variables computed in a simulation which may lead to solver instability in approximate numbering systems like floating point. We note that non-dimensionalization also results in a characteristic non-dimensional time for each PDE, providing insight on the relative time scales encountered in the evolution of the individual and coupled physical systems.

3.1 Balance equation

To facilitate simplified analysis and ease of implementation of the coupled multiphysics system, the PDEs defined on the fluid and solid domains are presented in a common form, namely as a set of balance equations given by

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{u}) + \nabla \cdot \mathbf{G}(\mathbf{u}, \nabla \mathbf{u}) = \mathbf{f} \quad (3.1)$$

where \mathbf{u} is the vector of state variables, ∇ is the usual spatial gradient operator, \mathbf{F} and \mathbf{G} are (potentially nonlinear) convective and diffusive flux functions, and \mathbf{f} is a source term. We note that only the diffusive flux is a function of the spatial gradient of the state variables. We refer to (3.1) as the strong form of the balance equation.

3.1.1 Hybridized DG discretization

Determining a function \mathbf{u} satisfying (3.1) is greatly complicated by the strong continuity requirements imposed by the differential operators. Further, the solution \mathbf{u} is in general required to exist in an infinite-dimensional function space. These constraints preclude determination of a solution to the strong form. For this reason, we consider computing an approximate solution of the strong form using a hybridized DG finite element discretization. The finite element solution will be shown to reside in a finite-dimensional subset of the infinite-dimensional continuous solution space with weakened continuity requirements. This weak solution is said to satisfy a weak form of (3.1) realized by the hybridized DG method which will be presented shortly. We must first provide technical description of the necessary ingredients to derive the weak form, including the spatial domain and finite element function spaces.

3.1.1.1 Finite element triangulation

We consider a continuous d -dimensional domain $\Omega \subset \mathbb{R}^d$ with boundary $\partial\Omega$. Finite element methods operate on a triangulation \mathcal{T}_h of Ω consisting of non-overlapping subdomains K referred to here as elements. Further, we denote the set of element boundaries of \mathcal{T}_h by $\partial\mathcal{T}_h = \{\partial K : K \in \mathcal{T}_h\}$. We denote by \mathbf{n} the outward pointing unit normal vector to an element boundary ∂K . We make a

distinction between interior element boundaries and element boundaries contained in $\partial\Omega$, as these boundaries will receive different treatment as will be seen in subsequent sections.

3.1.1.2 Finite element spaces

We consider the polynomial space $\mathcal{P}^k(S)$ consisting of the polynomials of degree of at most k defined on an arbitrary domain S . The finite element spaces considered are then given by

$$\begin{aligned}\mathcal{W}_h^k &= \{r \in L^2(\mathcal{T}_h) : r|_K \in \mathcal{P}^k(K), \forall K \in \mathcal{T}_h\} \\ \mathcal{V}_h^k &= \{\mathbf{r} \in \left(L^2(\mathcal{T}_h)\right)^m : \mathbf{r}|_K \in \left(\mathcal{P}^k(K)\right)^m, \forall K \in \mathcal{T}_h\} \\ \mathcal{Q}_h^k &= \{\mathbf{R} \in \left(L^2(\mathcal{T}_h)\right)^{m \times d} : \mathbf{R}|_K \in \left(\mathcal{P}^k(K)\right)^{m \times d}, \forall K \in \mathcal{T}_h\}\end{aligned}$$

where we have used lowercase, bold lowercase, and bold uppercase letters to represent scalar-, vector-, and matrix-valued functions, respectively. Hybridized DG methods also consider solution variables defined on the edges of the triangulation referred to as hybrid or trace variables. For this reason, we also define the finite element space \mathcal{M}_h^k given by

$$\mathcal{M}_h^k = \{\boldsymbol{\mu} \in \left(L^2(\partial\mathcal{T}_h)\right)^m : \boldsymbol{\mu}|_K \in \left(\mathcal{P}^k(K)\right)^m, \forall \partial K \in \mathcal{T}_h\}$$

Choice of a basis for a finite element space permits computation of arbitrary functions in the finite element space. For example, for an n -dimensional space V_h spanned by a basis $\boldsymbol{\phi}_i$ for $i \in [1, n]$, a function $\mathbf{u}_h \in V_h$ is computed via

$$\mathbf{u}_h(\mathbf{x}) = \sum_{i=1}^n u_i \boldsymbol{\phi}_i(\mathbf{x}) \tag{3.2}$$

3.1.1.3 Inner products

Finite element spaces are also inner product spaces. DG methods consider finite element spaces which are continuous within an element K but discontinuous on the element boundary ∂K . For this reason, the inner products defined on the DG finite element spaces are given by

$$\begin{aligned} (a, b)_{\mathcal{T}_h} &= \sum_{K \in \mathcal{T}_h} (a, b)_K & (\mathbf{a}, \mathbf{b})_{\mathcal{T}_h} &= \sum_{K \in \mathcal{T}_h} (\mathbf{a}, \mathbf{b})_K & (\mathbf{A}, \mathbf{B})_{\mathcal{T}_h} &= \sum_{K \in \mathcal{T}_h} (\mathbf{A}, \mathbf{B})_K \\ \langle a, b \rangle_{\partial \mathcal{T}_h} &= \sum_{\partial K \in \partial \mathcal{T}_h} \langle a, b \rangle_{\partial K} & \langle \mathbf{a}, \mathbf{b} \rangle_{\partial \mathcal{T}_h} &= \sum_{\partial K \in \partial \mathcal{T}_h} \langle \mathbf{a}, \mathbf{b} \rangle_{\partial K} & \langle \mathbf{A}, \mathbf{B} \rangle_{\partial \mathcal{T}_h} &= \sum_{\partial K \in \partial \mathcal{T}_h} \langle \mathbf{A}, \mathbf{B} \rangle_{\partial K} \end{aligned}$$

where we have used the notation $(\cdot, \cdot)_K$ and $\langle \cdot, \cdot \rangle_{\partial K}$ to denote the usual L^2 inner products defined on the element and element boundary respectively. These L^2 inner products are given by

$$\begin{aligned} (a, b)_K &= \int_K ab & (\mathbf{a}, \mathbf{b})_K &= \int_K \mathbf{a} \cdot \mathbf{b} & (\mathbf{A}, \mathbf{B})_K &= \int_K \mathbf{A} : \mathbf{B} \\ \langle a, b \rangle_{\partial K} &= \int_{\partial K} ab & \langle \mathbf{a}, \mathbf{b} \rangle_{\partial K} &= \int_{\partial K} \mathbf{a} \cdot \mathbf{b} & \langle \mathbf{A}, \mathbf{B} \rangle_{\partial K} &= \int_{\partial K} \mathbf{A} : \mathbf{B} \end{aligned}$$

with $\mathbf{A} : \mathbf{B} = \text{tr}(\mathbf{A}^T \mathbf{B})$ denoting the tensor product of \mathbf{A} and \mathbf{B} .

3.1.1.4 Discretization

We now discuss the hybridized DG discretization of the model problem (3.1). We begin by restating (3.1) as a first order system resulting in

$$\mathbf{q} = \nabla \mathbf{u} \tag{3.3a}$$

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{F}(\mathbf{u}) + \mathbf{G}(\mathbf{u}, \mathbf{q})) = \mathbf{f} \tag{3.3b}$$

where we have introduced the variable \mathbf{q} and grouped the flux terms. We refer to (3.3) as the first order strong form of the balance equation. We seek to determine a finite-dimensional approximation of (3.3) by forming the corresponding weak form. The weak form is realized by multiplying each equation by a test function and integrating. The discrete problem reads, find $(\mathbf{u}_h, \mathbf{q}_h, \hat{\mathbf{u}}_h) \in \mathcal{V}_h \times \mathcal{Q}_h \times \mathcal{M}_h$ such that

$$(\mathbf{q}_h, \mathbf{r})_{\mathcal{T}_h} + (\mathbf{u}_h, \nabla \cdot \mathbf{r})_{\mathcal{T}_h} - \langle \hat{\mathbf{u}}_h, \mathbf{r} \cdot \mathbf{n} \rangle_{\partial \mathcal{T}_h} = 0 \quad (3.4a)$$

$$\left(\frac{\partial \mathbf{u}_h}{\partial t}, \mathbf{w} \right)_{\mathcal{T}_h} - (\mathbf{F}(\mathbf{u}_h) + \mathbf{G}(\mathbf{u}_h, \mathbf{q}_h), \nabla \mathbf{w})_{\mathcal{T}_h} + \left\langle \hat{\mathbf{f}}_h(\hat{\mathbf{u}}_h, \mathbf{u}_h, \mathbf{q}_h), \mathbf{w} \right\rangle_{\partial \mathcal{T}_h} = 0 \quad (3.4b)$$

$$\left\langle \hat{\mathbf{f}}_h(\hat{\mathbf{u}}_h, \mathbf{u}_h, \mathbf{q}_h), \boldsymbol{\mu} \right\rangle_{\partial \mathcal{T}_h \setminus \partial \Omega} + \left\langle \hat{\mathbf{b}}_h(\hat{\mathbf{u}}_h, \mathbf{u}, \mathbf{q}_h), \boldsymbol{\mu} \right\rangle_{\partial \Omega} = 0 \quad (3.4c)$$

for all $(\mathbf{r}, \mathbf{w}, \boldsymbol{\mu}) \in \mathcal{V}_h \times \mathcal{Q}_h \times \mathcal{M}_h$ where $\hat{\mathbf{f}}_h$ is the numerical flux function which remains to be defined and $\hat{\mathbf{b}}_h$ are the boundary conditions. We note that the partial derivatives in (3.3) are shifted to the test functions by partial integration.

3.1.1.5 Numerical flux

The numerical flux imposes single-valuedness of the trace variable $\hat{\mathbf{u}}$. Indeed, the numerical flux function results in the hybridized DG scheme being a conservative scheme. In this work, the numerical flux in hybridized DG methods takes the form

$$\hat{\mathbf{f}}_h(\hat{\mathbf{u}}_h, \mathbf{u}_h, \mathbf{q}_h) = (\mathbf{F}(\hat{\mathbf{u}}_h) + \mathbf{G}(\hat{\mathbf{u}}_h, \mathbf{q}_h)) \cdot \mathbf{n} + \boldsymbol{\sigma}(\hat{\mathbf{u}}_h, \mathbf{u}_h, \mathbf{q}_h, \mathbf{n})(\mathbf{u}_h - \hat{\mathbf{u}}_h) \quad (3.5)$$

where $\boldsymbol{\sigma}$ is the so-called stabilization matrix introduced by the hybridized DG method and is often determined using knowledge of the simulated physics. The choice of the stabilization matrix can

have considerable effect on the accuracy and stability of a hybridized DG scheme. We refer the interested reader to Cockburn’s description of hybridized DG methods [23] for greater discussion on the derivation and role of the stabilization matrix.

3.1.1.6 Boundary conditions

Boundary conditions in the hybridized DG method are weakly satisfied resulting in the inclusion of boundary trace variables in the nonlinear solution procedure. Hybridized DG methods have been formulated to satisfy Dirichlet and Neumann boundary conditions [26, 93]. In this work, we only consider Dirichlet conditions and omit description of Neumann boundary condition enforcement. The introduction of the trace variables allow hybridized DG methods to consider state-based boundary conditions [46], a feature not shared by conventional DG methods. A state-based boundary condition is defined as

$$\hat{\mathbf{b}}_h(\hat{\mathbf{u}}, \mathbf{u}, \mathbf{q}) = \hat{\mathbf{u}} - \mathbf{u}_b \quad (3.6)$$

for a target state \mathbf{u}_b .

In the sections that follow, we introduce the governing equations for each physical domain and demonstrate how they fit in the model form (3.1) through judicious choice of convective flux, diffusive flux, and source terms. Each section concludes with discussion concerning the determination of appropriate stabilization functions employed by the numerical flux. Finally, we present the coupling scheme employed to generate the monolithic aerothermoelastic equation system. We omit the fluid and solid subscripts f and s on variables and material properties to

simplify notation in the following sections. These subscripts are included for clarity in the coupling discussion.

3.2 Fluid domain

As discussed in Chapter II, hypersonic flows exhibit complex flow phenomena including shock formation, thick boundary layers, and significant fluid compression. Hypersonic flows of especially high Mach number can result in extremely low density flows which require consideration of individual particles. In this thesis, we restrict attention to lower speed hypersonic flows (i.e. around Mach 5) such that the continuum assumption remains valid. In this regime, the Navier-Stokes equations find extensive use and are well-suited for determining the evolution of fluid momentum in a compressible fluid. The Navier-Stokes equations are also equipped with equations describing the conservation of mass and energy.

3.2.1 Continuous equations

The temporal and spatial state of the fluid is expressed via the volumetric transport of fluid mass, momentum, and energy described by a state vector $\mathbf{u} = \begin{bmatrix} \rho & \rho v_i & \rho E \end{bmatrix}^T$ with $m_f = d + 2$ components. We write the Navier-Stokes system in the balance form given by (3.1) with the (m_f, d) fluxes given by

$$\mathbf{F} = \begin{bmatrix} \rho v_i \\ \rho v_i v_j + \delta_{ij} p \\ v_i (\rho E + p) \end{bmatrix} \quad (3.7a)$$

$$\mathbf{G} = \begin{bmatrix} 0 \\ \tau_{ij} \\ v_j \tau_{ij} + \kappa \frac{\partial T}{\partial x_i} \end{bmatrix} \quad (3.7b)$$

where d is the spatial dimension, p is the pressure, τ is the stress rate tensor, κ is the thermal conductivity of the fluid, and $i, j \in [1, d]$. We consider a stress rate tensor of the form

$$\tau = \mu \left[\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) + \frac{2}{3} \frac{\partial v_k}{\partial x_k} \right] \quad (3.8)$$

where μ is the dynamic viscosity and we have used Einstein notation to indicate summation. The pressure is determined by an equation of state and remains to be specified. The ideal gas law provides an accurate model of pressure for a wide range of flow conditions. Still, real gas laws are sometimes considered for hypersonic flows. In this work, we consider lower speed hypersonic flows such that the ideal gas law is assumed to remain valid. The fluid pressure and temperature are thusly given by

$$p = (\gamma - 1) \left(\rho E - \frac{1}{2} \rho v_k^2 \right) \quad (3.9a)$$

$$T = \frac{p}{(\gamma - 1) \rho C_v} \quad (3.9b)$$

where γ is the specific heat ratio and C_v is the specific heat of the fluid at constant volume. For flows with nominal temperature variation, fluid properties including the specific heat ratio, thermal conductivity, and dynamic viscosity are considered to be constant. However, due to the considerable thermal variations in hypersonic flows, it can be necessary to consider thermally-dependent values of fluid properties. A source term f can also be provided, though, in this work, no source term is considered.

3.2.2 Non-dimensionalization

In particular, we consider the following non-dimensionalization

$$\tilde{x}_i = \frac{x_i}{\bar{L}} \quad \tilde{\rho} = \frac{\rho}{\bar{\rho}} \quad \tilde{v}_i = \frac{v_i}{\bar{v}} \quad \tilde{t} = \frac{t}{\bar{t}} \quad (3.10a)$$

$$\tilde{p} = \frac{p}{\bar{\rho}\bar{u}^2} \quad \tilde{\rho E} = \frac{\rho E}{\bar{\rho}\bar{u}^2} \quad \tilde{T} = \frac{\bar{C}_v T}{\bar{v}^2} \quad (3.10b)$$

$$\tilde{\mu} = \frac{\mu}{\bar{\mu}} \quad \tilde{\gamma} = \frac{\gamma}{\bar{\gamma}} \quad \tilde{C}_v = \frac{C_v}{\bar{C}_v} \quad \tilde{\kappa} = \frac{\kappa}{\bar{\kappa}} \quad (3.10c)$$

where $\tilde{\square}$ represents a non-dimensional value and $\bar{\square}$ is the dimensional scaling value. The first row of non-dimensional variables (3.10a) considers the primitive flow variables and time scaling. The second row (3.10b) considers flow variables whose scaling is determined from the primitive flow variables. The third row (3.10c) considers material properties. In this work we take $\bar{\rho} = \rho_\infty$ and $\bar{v} = v_\infty$ where \square_∞ denotes the freestream value. The characteristic length \bar{L} is problem dependent and is usually determined from the flow geometry. The characteristic time is taken to be $\bar{t} = \frac{\bar{v}}{\bar{L}}$ and is used to also define the time scale of the fluid problem. Finally, the scaling parameters for each

of the temperature-varying material properties are taken to be their respective values at freestream conditions.

Substituting these values into the dimensional Navier-Stokes fluxes (3.7) yields the non-dimensional fluxes given by

$$\tilde{\mathbf{F}} = \begin{bmatrix} \widetilde{\rho v_i} \\ \widetilde{\rho v_i v_j} + \delta_{ij} \widetilde{p} \\ \widetilde{v_i} \left(\widetilde{\rho E} + \widetilde{p} \right) \end{bmatrix} \quad (3.11a)$$

$$\tilde{\mathbf{G}} = \begin{bmatrix} 0 \\ \widetilde{\tau_{ij}} \\ \widetilde{v_j} \widetilde{\tau_{ij}} + \frac{\gamma}{\text{RePr}} \frac{\partial \widetilde{T}}{\partial \widetilde{x_i}} \end{bmatrix} \quad (3.11b)$$

$$\widetilde{\tau_{ij}} = \frac{1}{\text{Re}} \left[\left(\frac{\partial \widetilde{v_i}}{\partial \widetilde{x_j}} + \frac{\partial \widetilde{v_j}}{\partial \widetilde{x_i}} \right) + \frac{2}{3} \frac{\partial \widetilde{v_k}}{\partial \widetilde{x_k}} \right] \quad (3.11c)$$

where $\text{Re} = \frac{\bar{\rho} \bar{v} \bar{L}}{\bar{\mu}}$ is the Reynolds number and $\text{Pr} = \frac{\bar{\gamma} \bar{C}_v \bar{\mu}}{\bar{k}}$ is the Prandtl number evaluated at \bar{T} .

3.2.3 HDG stabilization

The stabilization matrix in the hybridized DG Navier-Stokes equations defines an implicit Riemann solver [50]. In this work we consider Lax-Friedrichs scheme [93, 49] given by

$$\boldsymbol{\sigma} = \lambda_{\max} (\mathbf{A} (\hat{\mathbf{u}}) \cdot \mathbf{n}) \mathbf{I} \quad (3.12)$$

where \mathbf{A} is the convective flux Jacobian with entries $\mathbf{A}(\mathbf{u})_{ij} = \frac{(\mathbf{F}(\mathbf{u}) \cdot \mathbf{n})_i}{u_j}$.

3.3 Solid domain

3.3.1 Continuous equations

For the solid domain, we firstly assume the application of a material capable of withstanding the pressure loads applied by the hypersonic flow such that yielding does not occur and secondly that structural displacements due to the pressure loading are sufficiently small so that infinitesimal strain theory can be accurately applied. Under these assumptions, we apply the linear elastodynamics equation for computation of the structural deformation. The elastodynamics equation is given by

$$\rho \frac{\partial^2 \mathbf{u}}{\partial t^2} = \nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{b} \quad (3.13)$$

where ρ is the solid material density, \mathbf{u} is the solid displacement, $\boldsymbol{\sigma}$ is internal stress tensor, and \mathbf{b} is an acceleration due to an external loading. In order to represent (3.13) in the form (3.1), first consider a rendering of (3.13) via the introduction of a solid velocity \mathbf{v} . With the inclusion of \mathbf{v} , (3.13) can be written as

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{v} \quad (3.14a)$$

$$\rho \frac{\partial \mathbf{v}}{\partial t} = \nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{b} \quad (3.14b)$$

Determination of the internal stress tensor $\boldsymbol{\sigma}$ requires consideration of the deformation regime and the microstructure of the material and is an ongoing area of research. However, under the assumption of elastic deformation, the stress tensor $\boldsymbol{\sigma}$ is well-defined and is frequently assumed to vary linearly with the material strain $\boldsymbol{\epsilon}$ via the relation $\boldsymbol{\sigma} = \mathbf{C}:\boldsymbol{\epsilon}$ where \mathbf{C} is the Hooke tensor whose entries include the influence of the considered material. We consider this formulation of the stress tensor in this work. Finally, we write the strain as a function of the displacement gradients via

$$\boldsymbol{\epsilon} = \frac{1}{2} \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right) \quad (3.15)$$

We note that (3.15) reveals the dependence of the internal solid stress on the solid displacements, fully determining (3.13). For an isotropic material, the Hooke tensor is symmetric and the stress tensor can be simplified considerably, resulting in

$$\begin{aligned} \boldsymbol{\sigma} &= 2\mu\boldsymbol{\epsilon} + \lambda\mathbf{I} (\nabla \cdot \mathbf{u}) \\ &= \mu \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right) + \lambda\mathbf{I} (\nabla \cdot \mathbf{u}) \end{aligned} \quad (3.16)$$

where λ and μ are Lamé's first and second parameters, respectively. Finally, (3.14) can be written in the model form (3.1) via

$$\mathbf{F} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad (3.17a)$$

$$\mathbf{G} = \begin{bmatrix} \mathbf{0} \\ \frac{1}{\rho} \boldsymbol{\sigma} \end{bmatrix} \quad (3.17b)$$

$$\mathbf{f} = \begin{bmatrix} \mathbf{v} \\ \mathbf{b} \end{bmatrix} \quad (3.17c)$$

3.3.2 Non-dimensionalization

The elastodynamics system is non-dimensionalized by

$$\tilde{x}_i = \frac{x_i}{\bar{L}} \quad \tilde{u}_i = \frac{u_i}{\bar{\epsilon}\bar{L}} \quad \tilde{v}_i = \frac{v_i\bar{t}}{\bar{u}} \quad \tilde{t} = \frac{t}{\bar{t}} \quad (3.18a)$$

$$\tilde{\epsilon} = \frac{\epsilon}{\bar{\epsilon}} \quad \tilde{\sigma} = \frac{\sigma}{\bar{\epsilon}\bar{E}} \quad (3.18b)$$

$$\tilde{\lambda} = \frac{\lambda}{\bar{\lambda}} \quad \tilde{\mu} = \frac{\mu}{\bar{\mu}} \quad (3.18c)$$

where \bar{E} is Young's modulus for the material given by $\bar{E} = \frac{\bar{\mu}(3\bar{\lambda}+2\bar{\mu})}{\bar{\lambda}+\bar{\mu}}$. The non-dimensional time is given by $\bar{t} = \bar{L}\sqrt{\frac{\rho}{E}}$. Non-dimensionalization of (3.14) yields

$$\frac{\partial \tilde{\mathbf{u}}}{\partial \tilde{t}} = \tilde{\mathbf{v}} \quad (3.19a)$$

$$\frac{\partial \tilde{\mathbf{v}}}{\partial \tilde{t}} = \tilde{\nabla} \cdot \tilde{\boldsymbol{\sigma}} + \frac{\tilde{t}^2 \tilde{\mathbf{b}}}{\bar{u}} \quad (3.19b)$$

Equation (3.14) can now be expressed as a balance equation with variables

$$\mathbf{u}_s = \begin{bmatrix} u_i \\ v_i \end{bmatrix} \quad (3.20)$$

$$\tilde{\mathbf{F}} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad (3.21a)$$

$$\tilde{\mathbf{G}} = \begin{bmatrix} \mathbf{0} \\ \tilde{\boldsymbol{\sigma}} \end{bmatrix} \quad (3.21b)$$

$$\tilde{\mathbf{f}} = \begin{bmatrix} \tilde{\mathbf{v}} \\ \frac{\tilde{t}^2}{\bar{u}} \tilde{\mathbf{b}} \end{bmatrix} \quad (3.21c)$$

3.3.3 HDG Stabilization

Stabilization of the first order solid equations requires choice of stabilization factors for both the displacement and velocity relationships with appropriate units. Elliptic problems are seemingly less sensitive to the stabilization and can permit a wide range of stabilization factors. Further, we are considering non-dimensional values. For these reasons, we simply take the stabilization factor for both equations to be unity which has performed well in practice.

3.4 Thermal domain

In this work, we consider heat conduction through the solid. Many hypersonic aerothermoelasticity models focus on surface heating prediction and fail to model heat transfer within the body. However, the solid temperature distribution can have considerable effect on the material strength. Indeed, the rigidity of the considered solid materials decreases considerably over the temperature ranges common for hypersonic flows.

3.4.1 Continuous equations

With these concerns, we model heat transfer in the solid using Fourier's law of thermal conduction, expressed as a PDE via

$$\rho c_p \frac{\partial T}{\partial t} = \nabla \cdot (\kappa \nabla T) \quad (3.22)$$

where T is the solid temperature, ρ is the density of the solid, c_p is the thermally-varying specific heat capacity at constant pressure, and κ is the thermally-varying thermal conductivity. (3.22) nearly fits the form of (3.1) with no convective flux and the diffusive flux $G = \kappa \nabla T$. However, the temporal term has a leading coefficient ρc_p which will be removed in non-dimensionalization.

3.4.2 Non-dimensionalization

We consider a straightforward non-dimensionalization of (3.22) given by

$$\tilde{x}_i = \frac{x_i}{\bar{L}} \quad \tilde{T} = \frac{T}{\bar{T}} \quad \tilde{t} = \frac{t}{\bar{t}} \quad (3.23a)$$

$$\tilde{c}_p = \frac{c_p}{\bar{c}_p} \quad \tilde{\kappa} = \frac{\kappa}{\bar{\kappa}} \quad (3.23b)$$

where \bar{L} is a characteristic thermal length scale, \bar{T} is taken to be the initial temperature of solid, \bar{c}_p and $\bar{\kappa}$ are the specific heat and thermal conductivity evaluated at \bar{T} . Finally, the non-dimensional time is determined by the non-dimensional Fourier number $\text{Fo} = \frac{\bar{\kappa}\bar{t}}{\rho\bar{c}_p\bar{L}^2}$, resulting in $\bar{t} = \frac{\text{Fo}\bar{L}^2\rho\bar{c}_p}{\bar{\kappa}}$.

The non-dimensionalized heat equation is given by

$$\frac{\partial \tilde{T}}{\partial \tilde{t}} = \tilde{\nabla} \cdot (\tilde{\kappa} \tilde{\nabla} \tilde{T}) \quad (3.24)$$

It is easy to see that (3.24) satisfies the model form with diffusive flux $\tilde{G} = \tilde{\kappa} \tilde{\nabla} \tilde{T}$.

3.4.3 HDG stabilization

We employ the same approach for stabilization of the thermal equation system as we do for the structural equation system and simply take the stabilization factor to be unity.

3.5 Coupling

In the previous sections we presented the individual equation systems which are employed for the simulation of hypersonic aerothermoelasticity. In this section, we focus on the monolithic approach used to couple them. We take a similar approach as the previous sections and present the

coupling formulation in a continuous solution domain. We conclude with the section by detailing how the continuous coupling formulation is integrated within the hybridized DG model problem.

The solid-thermal coupling is the most readily achieved coupling. The addition of a thermal strain term to (3.15) is all that is requires. The thermal strain is given by

$$\epsilon_{s,\text{thermal}} = \alpha_s (T_s - T_{s,0}) \quad (3.25)$$

where α_s is referred to as the thermal expansion coefficient and $T_{s,0}$ is a reference temperature taken to be temperature of the undeformed structure. The total strain is simply the sum of the internal and thermal strain.

$$\epsilon_{s,\text{total}} = \epsilon_{s,\text{internal}} + \epsilon_{s,\text{thermal}} \quad (3.26)$$

3.5.1 Arbitrary Lagrange-Euler

Coupling of the three physical systems requires that all three equation systems are evolved in a common reference frame. We note that the Navier-Stokes equation system evolves in a Eulerian reference frame. In this reference frame, measurements of physical values are taken at static points which are not moved for the entirety of the simulation. In contrast, the elasticity and heat equations evolve in a Lagrangian reference frame. A Lagrangian reference frame considers physical values along points which are deformed throughout a simulation. To ensure a common reference is employed between all three physical systems, we employ the Arbitrary Lagrange-Euler (ALE) technique.

ALE methods reframe PDEs defined on either Lagrangian or Eulerian reference frames into a common reference frame by a transformation of the PDEs. ALE methods have become a tool of choice for monolithic fluid-structural interaction simulation [109, 111, 119], though they also enable accurate simulation in many other physical domains, including the resolution of traveling discontinuities, e.g. blast waves.

In this work, we transform the Navier-Stokes equations to a Lagrangian frame and employ the elasticity and heat equations as previously presented. The necessary ingredients for the ALE transformation of the Navier-Stokes equations is a mapping of the fluid domain from its initial reference configuration to a deformed configuration as the simulation time advances. To this end, we include an elastostatics PDE for the computation of fluid mesh displacements given by

$$\nabla \cdot \left[\mu_f \left(\nabla \mathbf{d}_f + (\nabla \mathbf{d}_f)^T \right) + \lambda_f \mathbf{I} (\nabla \cdot \mathbf{d}_f) \right] = 0 \quad (3.27)$$

where \mathbf{d}_f are the fluid mesh displacements. We note that (3.27) are simply the elastodynamics equations (3.13) lacking the temporal term on the left-hand side and with a zero source time. Employing (3.27) for fluid mesh deformation, has the attractive property that the nodes are diffused throughout the domain due to the elliptic nature of the PDE. In contrast to the elastodynamics equations applied on the solid domain, the fluid mesh Lamé parameters in (3.27) can be freely chosen by the analyst due to the physically fictitious nature of the fluid mesh deformation.

We follow the descriptions of the ALE transformation given in [100, 93]. We first introduce the following notation: values defined on the reference domain are represented with uppercase letters values defined on the deformed domain are denoted with lowercase letters. A spatial coordinate \mathbf{x} at a time t can be written in terms of its corresponding reference coordinate \mathbf{X} by

$$\mathbf{x}(\mathbf{X}, t) = \mathbf{X} + \mathbf{u}(t) \quad (3.28)$$

It is useful to consider the deformation gradient \mathbf{P} given by

$$\begin{aligned} \mathbf{P} &= \nabla_{\mathbf{X}} \mathbf{x} \\ &= \mathbf{I} + \nabla_{\mathbf{X}} \mathbf{u} \end{aligned} \quad (3.29)$$

We denote the determinant of the deformation gradient by $J = \det \mathbf{P}$. We follow the approaches of [100] to transform the model problem (3.4) with the ALE method resulting in

$$\mathbf{q} = \nabla \mathbf{u} \quad (3.30a)$$

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot [J\mathbf{P}^{-1}(\mathbf{F}(\mathbf{u}) + \mathbf{G}(\mathbf{u}, \mathbf{q})) - \mathbf{u}J\mathbf{P}^{-1}\mathbf{v}] = \mathbf{f} \quad (3.30b)$$

where \mathbf{v} is the mesh velocity which in this work is computed using a first order approximation from the mesh displacements. In general, the transformation $J\mathbf{P}^{-1}$ is used to map from the Eulerian frame to the Lagrangian frame.

3.5.2 Continuous equations

All multiphysics interactions occur on the fluid-solid interface. In this work, we enforce a two-way coupling scheme on the fluid-solid interface, such that both the fluid and solid domains impact each other. It is common in partitioned (FSI) coupling schemes to include one-way coupling schemes to simplify the physical scheme or reduce the implementation burden.

All coupling conditions are enforced on the Lagrangian frame after mapping of the fluid variables the ALE mapping. In particular, we require the following conditions on the fluid-solid interface

$$\begin{aligned}
\mathbf{d}_f &= \mathbf{d}_s \\
\mathbf{v}_f &= \mathbf{v}_s \\
T_f &= T_s \\
\boldsymbol{\sigma}_f \cdot \mathbf{n} &= \boldsymbol{\sigma}_s \cdot \mathbf{n} \\
\mathbf{q}_f \cdot \mathbf{n} &= \mathbf{q}_s \cdot \mathbf{n}
\end{aligned} \tag{3.31}$$

With the exception of the displacement coupling, the physics on the fluid domain and the solid domain are two-way coupled, meaning that these conditions are applied from both the perspectives of the fluid and the solid domains. The displacements are only one-way coupled, i.e. only from the fluid perspective. This is due to the fluid mesh lacking any physical basis, so the fluid mesh displacements do not inform the solid displacements. Instead, the solid displacements uniquely determine the fluid mesh displacements.

The second condition enforces continuity of the fluid velocity and the solid velocity, resulting in satisfaction of a no-slip condition. The third condition enforces continuity of the temperature field. The fourth condition imposes coupling of the fluid and solid traction and can be viewed as either a momentum flux condition or a forcing condition. Finally, the fifth condition enforces continuity of the heat fluxes and can be viewed as a thermal energy flux condition.

We make a distinction between the first three conditions in (3.31), which we refer to as *state-based* conditions, and the final two conditions which we refer to as *flux-based* conditions.

State-based conditions are enforced by the addition of an interface term requiring equality in the balance variables between the fluid and solid domains. Since the two domains are composed of differing state variables, analogues of the state variables for the opposite physics must be made. More concretely, we have from the perspective of the fluid domain

$$\begin{bmatrix} \widehat{\rho}_f \\ \widehat{\rho u}_{f,i} \\ \widehat{\rho E}_f \\ \widehat{d}_f \end{bmatrix} = \begin{bmatrix} \rho_f \\ \rho_f u_{s,i} \\ \rho E_f \\ d_s \end{bmatrix} \quad (3.32)$$

and from the perspective of the solid domain

$$\begin{bmatrix} \widehat{d}_{s,i} \\ \widehat{u}_{s,i} \\ \widehat{T}_s \end{bmatrix} = \begin{bmatrix} d_{s,i} \\ \frac{\rho u_{f,i}}{\rho_f} \\ T_f \end{bmatrix} \quad (3.33)$$

We make note of the extrapolation of solid displacement in (3.33) instead of coupling the fluid mesh displacements. It is necessary to express the state-based conditions in non-dimensional variables, resulting in

$$\begin{bmatrix} \widetilde{\rho}_f \\ \widetilde{\rho u}_{f,i} \\ \widetilde{\rho E}_f \\ \widetilde{d}_f \end{bmatrix} = \begin{bmatrix} \widetilde{\rho}_f \\ \widetilde{\rho}_f \widetilde{u}_{s,i} \frac{\bar{u}_s}{\bar{u}_f} \\ \widetilde{\rho E}_f \\ \widetilde{d}_s \frac{\bar{d}_s}{\bar{d}_f} \end{bmatrix} \quad (3.34)$$

$$\begin{bmatrix} \widetilde{d}_{s,i} \\ \widetilde{u}_{f,i} \\ \widetilde{T}_s \end{bmatrix} = \begin{bmatrix} \widetilde{d}_{s,i} \\ \frac{\widetilde{\rho u}_{f,i} \bar{u}_f}{\widetilde{\rho}_f \bar{u}_s} \\ \widetilde{T}_f \frac{\bar{T}_f}{\bar{T}_s} \end{bmatrix} \quad (3.35)$$

for the fluid and solid variables, respectively. The flux-based conditions receive a treatment similar to the state-based conditions. The key difference is that the flux conditions must match flux descriptions for the fluid and solid domains. For example, the fluid flux condition on the interface is given by

$$\begin{bmatrix} 0 \\ \boldsymbol{\sigma}_f \cdot \mathbf{n} \\ \mathbf{q}_f \cdot \mathbf{n} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \boldsymbol{\sigma}_s \cdot \mathbf{n} \\ \mathbf{q}_s \cdot \mathbf{n} \\ 0 \end{bmatrix} \quad (3.36)$$

while the solid flux condition on the interface is given by

$$\begin{bmatrix} 0 \\ \boldsymbol{\sigma}_s \cdot \mathbf{n} \\ \mathbf{q}_s \cdot \mathbf{n} \end{bmatrix} = \begin{bmatrix} 0 \\ \boldsymbol{\sigma}_f \cdot \mathbf{n} \\ \mathbf{q}_f \cdot \mathbf{n} \end{bmatrix} \quad (3.37)$$

We non-dimensionalize the flux conditions with the following scaling values

$$\bar{q}_f = \frac{\bar{\kappa}_f \bar{T}_f}{\bar{L}_f} \quad \bar{q}_s = \frac{\bar{\kappa}_s \bar{T}_s}{\bar{L}_s} \quad (3.38)$$

$$\bar{\sigma}_f = \frac{\bar{\mu}_f \bar{u}_f}{\bar{L}_f} \quad \bar{\sigma}_s = \bar{\epsilon}_s \bar{E}_s \quad (3.39)$$

which yields the non-dimensional flux conditions

$$\begin{bmatrix} 0 \\ \tilde{\sigma}_f \cdot \mathbf{n} \\ \tilde{q}_f \cdot \mathbf{n} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\sigma}_s \cdot \mathbf{n} \frac{\tilde{\sigma}_s}{\tilde{\sigma}_f} \\ \tilde{q}_s \cdot \mathbf{n} \frac{\tilde{q}_s}{\tilde{q}_f} \\ 0 \end{bmatrix} \quad (3.40)$$

$$\begin{bmatrix} 0 \\ \tilde{\sigma}_s \cdot \mathbf{n} \\ \tilde{q}_s \cdot \mathbf{n} \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\sigma}_f \cdot \mathbf{n} \frac{\tilde{\sigma}_f}{\tilde{\sigma}_s} \\ \tilde{q}_f \cdot \mathbf{n} \frac{\tilde{q}_f}{\tilde{q}_s} \end{bmatrix} \quad (3.41)$$

on the fluid and solid interfaces respectively.

3.5.3 Common time and length scales

Despite the fluid, solid, and thermal equations each requiring a distinct characteristic length, we consider a single length scale \bar{L} and set $\bar{L} = \bar{L}_f = \bar{L}_s = \bar{L}_t$ to ensure consistency of the spatial differential operator $\tilde{\nabla}$. In practice, the characteristic length is related to some length scale of the solid geometry.

Since we consider a monolithic system where fluid, solid, and thermal equations are composed into a single model equation, it is necessary to reconcile the non-dimensional time variables \tilde{t} for

application of a time integrator. As in the determination of the length scale, a single time scale \bar{t}^* must be chosen and scale each balance equation. In this work, we take \bar{t}^* to be the fluid time scale \bar{t}_f since the fluid time scale is expected to be the smallest time scale. Scaling of a balance equation with a constitutive time scale \bar{t} only requires multiplication of each term by the non-dimensional factor $\frac{\bar{t}}{\bar{t}^*}$.

3.5.4 Hybridized DG discretization

Finally, coupling of fluid and solid domains is performed by the addition of interface terms to the trace equations in the fluid and solid balance equations. The coupling conditions act as constraints on the solution defined on the fluid-solid interface from the fluid and solid perspectives, Γ_f and Γ_s . The fluid and solid coupling conditions are given by

$$\langle \mathbf{F}_f(\mathbf{u}_f, \mathbf{q}_f) \cdot \mathbf{n} + \sigma_f(\mathbf{u}_f - \hat{\mathbf{u}}_f), \mu_f \rangle_{\Gamma_f} + \langle \mathbf{T}_f \mathbf{F}_s(\mathbf{u}_s, \mathbf{q}_s) \cdot \mathbf{n} + \sigma_f(\mathbf{P}_f \mathbf{u}_s - \hat{\mathbf{u}}_f), \mu_f \rangle_{\Gamma_s} = 0 \quad (3.42a)$$

$$\langle \mathbf{F}_s(\mathbf{u}_s, \mathbf{q}_s) \cdot \mathbf{n} + \sigma_s(\mathbf{u}_s - \hat{\mathbf{u}}_s), \mu_s \rangle_{\Gamma_s} + \langle \mathbf{T}_s \mathbf{F}_f(\mathbf{u}_f, \mathbf{q}_f) \cdot \mathbf{n} + \sigma_s(\mathbf{P}_s \mathbf{u}_f - \hat{\mathbf{u}}_s), \mu_s \rangle_{\Gamma_f} = 0 \quad (3.42b)$$

where \mathbf{T}_d and \mathbf{P}_d project the flux and state, respectively, to domain d and \mathbf{F}_f and \mathbf{F}_s are the fluid and solid flux constraints, respectively.

CHAPTER IV

IMPLEMENTATION

In this chapter, we present the implementation of the hybridized DG discretization hypersonic aerothermoelasticity system presented in Chapter III. As mentioned in Chapter I, the major effort of this work was the development of a hybridized DG framework for monolithic multiphysics simulation using the Rust [1] programming language. Discussion of the nonlinear solver, linear solver, and temporal discretization employed in this code is provided in this chapter. Considerations to enable simulation of arbitrary multiphysics equation systems are also presented. Special attention is given to the efficient resolution of hybridized DG equation systems. While this chapter does include certain detail about the developed hybridized DG solver, discussion of code structure is largely omitted.

4.1 Nonlinear solver

Finite element theory relies heavily on the linearity of the considered strong form. However, many PDE systems are nonlinear with respect to their solution. Formation of the weak form for a nonlinear strong form has not yet resulted in a finite element system. Instead, it is necessary to determine a linearization of the nonlinear weak form.

4.1.1 Static condensation

As discussed in Chapter II, a major benefit of hybridized DG methods is that they employ static condensation to exchange computational effort solving a globally-coupled system of equations written in terms of element unknowns for a significantly smaller globally-coupled system of equations written in terms of interface unknowns and an additional element-local procedure to compute the element unknowns. Since the computational complexity of iterative and direct linear solvers is almost never linear with respect to the unknowns, decreasing the number of globally coupled unknowns can greatly decrease the computational effort in solving the nonlinear equation system. The additional problem of determining the element unknowns can be performed in parallel and thus contributes an often negligible amount of computation expense.

In the present implementation, we take advantage of opportunities for parallelism. A significant portion of the computational expense of many implicit solvers is the formation of a linearization of the underlying nonlinear problem. Hybridized DG methods require the use of a Newton method to achieve the promised computational efficiency and thus require the assembly of a Newton system. As noted in Chapter II, hybridized DG methods consider a split residual problem where a residual is computed both for the element and trace unknowns, respectively. We recall the Newton system generated by a hybridized DG method given in (2.6).

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \delta \mathbf{u} \\ \delta \hat{\mathbf{u}} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix} \quad (4.1)$$

In practice, formation of the global matrices \mathbf{A} , \mathbf{B} , \mathbf{C} , and \mathbf{D} is not necessary. Instead, the linear system in (4.1) can be computed on each individual element. Static condensation of the element-

local Newton system is then performed to produce an element-local reduced Newton system as given by (2.7).

We note that for especially low polynomial orders, computation of the matrix A^{-1} on individual elements may be tractable by analytic matrix inverse formulae or via a computational matrix inverse method. However, an increase in the polynomial order causes the dimension of the element-local A to increase significantly and the use of traditional matrix inverse algorithms may result in undue computational expense. In the present implementation, the element-local matrix K is computed by instead solving the linear system $AX = B$ for the matrix X and then computing $K = D - CX$. It is assumed that this intermediate linear system is small enough to be efficiently solved by a direct solver for typically considered polynomial orders. Computation of K is efficiently performed using the BLAS [10] `gemm` routine. A similar approach is applied for the computation of the element-local h .

4.1.2 Parallelization

It is apparent that an element's residual and Jacobian can be computed independently of those of every other element. Under this consideration, we compute these systems across the threads available on the present system. To this end, we employ the use of the Rust library `rayon` [36] to enable efficient thread-based parallelization for the computation of the element Newton systems via a non-invasive API requiring surprisingly minimal code changes. To enable thread-based parallelization, `rayon` employs a work-stealing thread balancing strategy. At the start of the program `rayon` creates a thread pool. When thread parallel work needs to be performed, `rayon` distributes the work across the threads available in the thread pool for processing. If a participating

thread exhausts its assigned work, it reaches into the yet un-processed work queue of another thread and steals a task for processing. The provided work-stealing strategy is an effective load balancing heuristic for a large amount of computational work that seeks to decrease the length of the critical path and ensure that each processor operates at peak utilization.

After an element Newton system is computed on a thread, it must be assembled into the global Newton system (2.7) The global Newton system is a data structure shared by all threads participating in computation of the element Newton systems. Due to this, accessing the global Newton system from the individual threads requires access of a shared synchronization primitive (e.g. a mutex) to prevent data races by ensuring exclusive mutable access to the global Newton system by a single thread at any instance in the program's execution.

Early implementations of the present hybridized DG framework performed computation of the element Newton systems and assembly into the global system in a single pass. However, this approach was found to greatly reduce parallel efficiency as many threads would finish computation of the element Newton system in a similar amount of time and then have to wait idly to access the global Newton system. Since computation of an element Newton system is a much more computationally-demanding task, this approach effectively reduced the amount of parallel workers to be one.

One approach to prevent this problem would be to separate the computation of the element Newton systems and global assembly into two passes. In this approach, the participating threads would compute the element Newton systems and efficiently store them in a single buffer. The element Newton systems could then be assembled into the global Newton system serially. We pursue an approach similar to this. However, the assembly into the global Newton system is

performed in parallel. To achieve parallelism, `rayon` was used to distribute element Newton systems and form various thread-local global assemblies. These global assemblies are combined in parallel to produce a single global assembly. This approach ensures all cores remain busy with necessary work and idling due to contention among threads trying to access a single global assembly is prevented.

4.1.3 Automatic differentiation

Simulation codes often require the computation of the Jacobian of the (often nonlinear) residual. However, the Jacobian is often error-prone to implement in software or difficult to derive analytically. For these reasons, several approaches are employed by simulation codes to prevent explicit implementation of the Jacobian altogether, including Jacobian-free Newton Krylov (JFNK) [19, 12, 72] and automatic differentiation approaches. JFNK approaches take advantage of the fact that the Jacobian in a Newton-Krylov method (i.e. a Newton system solved with a Krylov subspace method) is needed only to compute Krylov vectors via Jacobian-vector products. JFNK approaches compute these products through a finite difference approximation expressed in terms of the residual only. Despite the ability of the JFNK approaches to eliminate computation of the Jacobian, JFNK methods are a class of quasi-Newton methods and accordingly forego the well-known quadratic convergence of the Newton method.

In this work we instead consider an automatic differentiation (AD) approach for computation of the global Jacobian. AD methods have found considerable use in many domains including machine learning [99], AD approaches algorithmically compute derivatives of functions by making heavy use of the chain rule. AD methods do not rely on approximation, e.g by finite difference, but

instead compute exact derivatives of desired functions. Two modes of AD methods exist: forward-mode and reverse-mode. The major difference between the two modes of AD is their efficiency in computing large amounts of derivatives of a single function. Typically, forward-mode AD approaches perform well when the number of outputs of the differentiated function is significantly fewer than the number of inputs. Oppositely, reverse-mode AD approaches perform well when the differentiated function possesses significantly more outputs than inputs.

Due to the recognition that the number of unknowns is significantly fewer than the number of derivatives necessary to compute, in this work we pursue a reverse-mode AD approach for the computation of the element-local Jacobians. Indeed, if an element possesses N unknowns, computation of the Jacobian requires $O(N^2)$ unknowns. Reverse-mode AD works by storing a computation graph of elementary mathematical expressions which can be readily differentiated. The vertices of the computation graph are mathematical expressions, while the edges are the arguments of the expressions which may themselves also be expressions. Computation of the partial derivative of an expression with respect to another expression x is performed by traversing the graph and computing the partial derivative of the traversed nodes with respect to x . The final partial derivative is assembled through consideration of the partial derivatives of its neighbors via the chain rule.

It was found that forming the Jacobian using expressions written in terms of the element and trace unknowns produced correct Jacobians but also incurred a large computational expense. While it was expected that computation of the Jacobian would represent a significant portion of the execution cost of a simulation, it was found that the Jacobian computation represented the greatest computational expense. This was due in large part to the expression tree being exceptionally large.

Computation of each entry in the Jacobian required a unique graph traversal, the computational cost of which is sensitive to memory access and allocation patterns in the graph data structure.

One remedy to this issue was to greatly reduce the size of the expression graph by computing some portions of the Jacobian analytically. We note that the entries of the element Jacobian can be expressed as

$$\begin{aligned} J_{ij} &= \frac{\partial r_i}{\partial u_j} \\ &= \frac{\partial r_i}{\partial u_h} \frac{\partial u_h}{\partial u_j} \end{aligned} \tag{4.2}$$

where we recall that u_i is the i_{th} degree of a freedom for a physical variable $u_h \in V_h$ which is given by $u_h(x) = \sum_i u_i \phi_i(x)$. In the present implementation, only the $\frac{\partial r_i}{\partial u_h}$ terms are computed using reverse-mode AD and the $\frac{\partial u_h}{\partial u_j}$ terms are computed analytically.

We conclude this section by stressing the importance of an efficient implementation of the AD data structures. Traversal of the expression graph often requires random access of a large number of graph nodes. If element-local Jacobians are computed on various threads, the combined size of the element-local AD graphs may exceed the available cache memory and require storage in RAM. In this case, the program may become memory bound due to the relatively large cost of frequent random access of expression nodes in RAM. This problem was remedied in part by caching the result of an expression and its adjoint derivatives. Finally, to prevent the large cost of memory allocation, the backing store for the expression graph is allocated once and reused throughout the execution of the program.

4.2 Linear solver

In this work, we employ the iterative General Minimized Residual (GMRES) [103] solver for the solution of the global trace system. GMRES is a member of the so-called Krylov subspace solution methods for which it can be shown that the error in the solution depends heavily on the condition number of the matrix. For this reason, it is necessary to precondition the linear system to obtain a solution in a reasonable amount of computation time. Since preconditioners are often determined by taking into account the underlying physics, determination of preconditioners for monolithic multiphysics systems [82] can be difficult. We have found that the incomplete LU decomposition [87] performs well on isolated fluid and elasticity problems and is a decent preconditioner for the hypersonic aerothermoelasticity system and is employed in this work.

4.3 Multiphysics considerations

The presented HDG implementation was designed with special consideration of monolithic multiphysics equation systems. The necessary ingredients to this end are as follows 1) decomposition of the finite element triangulation into physical subdomains, 2) application of a distinct equation system to each physical subdomain, and 3) addition of interface coupling conditions on multiphysics boundaries. We shall provide greater description of these ingredients in the following subsections.

4.3.1 Physical domain

Multiphysics systems consider physical domains where distinct physical laws are supposed to govern. Typically, partitioned approaches allow various domains to be defined within the simulation codes which perform a single-physics simulation. However, monolithic approaches

consider a single equation system defined on a single triangulation, requiring the prescription of subdomains for the constituent physics to be evolved. In the current approach, these physical domains are prescribed by the imported mesh file. Each physical domain in the mesh file must be provided with a name that identifies the domain. During the mesh import process, domain names are mapped to unique numeric identifiers.

4.3.2 Equation systems

Unique equation systems can be applied on the various domains. In the present implementation, equation systems are recognized to simply be a set of equations and equations are recognized to be a set of terms. Each term in a finite element method is an inner product. Inner products require one of only a few operations on test functions, e.g. gradient evaluation or divergence evaluation. Further, the inner products defined on elements and interfaces differ only in their domains.

A hybridized DG method requires equations for both the element and trace unknowns. Application of element equations is straightforward since no coupling between neighboring element equations arises due to the discontinuous nature of the element solutions. However, interface conditions require special care. In particular, three interface conditions must be handled: conditions on boundary interfaces, conditions on interfaces dividing elements with the same the element equations, and conditions on interfaces dividing elements with different element equations. Application of boundary conditions can be considered as a boundary element neighboring a ghost element which possesses the desired boundary state. Application of the equations defined on interfaces with the same element equations is straightforward: the same interface condition is applied on the interface from the perspective of both elements. However, conditions defined on interfaces

dividing elements with different element equations are harder to implement. In the present implementation, we assume that all interface conditions can be expressed as a combination of state equivalence condition or a flux balance condition. This formulation corresponds to a combination of Dirichlet and Neumann conditions being applied on these interfaces as seen in (3.42).

4.4 Temporal

Under the consideration of time-varying solutions, the hybridized DG method gives rise to a differential-algebraic (DAE) system of index 1. Indeed, the semi-discretized model problem (3.4) yields an ODE for \mathbf{u} and constraints for the gradient of the state vector \mathbf{q} and the approximation of the trace of the state vector $\hat{\mathbf{u}}$. Conventional time integration methods can be applied for the evolution of (3.4b) subject to the constraints (3.4a) and (3.4c). Explicit time integration methods for DG methods have the benefit that each element can be evolved independently of every other element due to the discontinuous nature of the solution space. However, explicit methods often require restrictive time steps for many problems of considerable size and forego the computational savings enjoyed by implicit methods due to elimination of element variables from the global problem.

We consider (3.4) in the more concise DAE form

$$\begin{aligned} \mathbf{M} \frac{\partial \mathbf{u}}{\partial t} &= \mathbf{f}(\mathbf{u}_h, \mathbf{q}_h, \hat{\mathbf{u}}_h) \\ \text{s.t. } \mathbf{g}(\mathbf{u}_h, \mathbf{q}_h, \hat{\mathbf{u}}_h) &= \mathbf{0} \end{aligned} \tag{4.3}$$

where \mathbf{M} is the so-called finite element mass matrix with entries $M_{ij} = (\boldsymbol{\phi}_i, \boldsymbol{\phi}_j)_K$ for $\boldsymbol{\phi}_i, \boldsymbol{\phi}_j \in V_h$, \mathbf{u} are the coefficients which determine $\mathbf{u}_h \in V_h$, \mathbf{f} encodes the dynamics in (3.4b), and \mathbf{g} encodes

the constraints imposed by (3.4a) and (3.4c). We note that \mathbf{M} reduces to the identity matrix in the case that \mathbf{u}_h is spanned by an orthonormal polynomial basis.

4.4.1 Time integrators

Implicit time integrations methods commonly applied to hybridized DG systems include backward difference formulae (BDF) [34, 56] and diagonally-implicit Runge Kutta (DIRK) methods [3]. BDF methods of order n approximate the temporal term of (4.3) with a Lagrange polynomial determined from the solution of the previous n time steps. For an early time step with index $i < n$, a BDF method of order i is employed. In particular, BDF methods discretize the time derivative $\frac{\partial \mathbf{u}}{\partial t} \approx \sum_i^n c_i \mathbf{u}_i$ where the coefficients c_i are determined to ensure desirable qualities for the temporal discretization, e.g. stability. The order of accuracy of BDF methods have been shown to increase with order of the BDF method for BDF orders less than 6. However, only BDF methods of up to order 3 have been shown to be A-stable. The BDF1 method results in the Backward Euler method which, in addition to being A-stable, is also L-stable.

DIRK methods are Runge Kutta (RK) methods whose A matrix in the Butcher tableau is a lower-diagonal matrix with non-zero entries in the diagonal. Though not discussed in much detail in this work, DIRK methods can be algorithmically viewed as repeated application of a BDF method within a single time step. The introduction of non-zero diagonal entries transition an explicit RK method to an implicit RK method. RK methods operate in stages within a single time step and require the solution of the ODE at each stage. For implicit methods, RK methods may introduce considerable computational expense due to a q -stage RK method requiring the inversion of q linear systems at each time step. For linear systems resulting from the application of a Newton

method, it has been shown that computation of each time step may be expedited by lagging the Jacobian matrix for several Newton iterations or even time steps [46]. Quasi-Newton methods may also provide computational savings, though they lose the quadratic convergence which make Newton methods attractive. For transient simulation of a hybridized DG DAE system employing a DIRK method, computation of the constraint variables for the final DIRK stage may be ignored to achieve a computational savings or computed if the constraint variables are needed for another purpose, e.g. visualization [93, 46].

4.4.2 Pseudo-Transient Continuation

The disparate time scales of the hypersonic aerothermoelasticity equations warrant long-time integration for the accurate capture of all constituent physics. The evolution of a hypersonic fluid on a fine grid may require time steps on the order of microseconds, while the conduction of the heat through solid may require time steps on the order of milliseconds or even seconds. Still, temporal evolution of the hypersonic aerothermoelasticity system is bounded by the smallest time step which, in practice, is often the fluid time step. Simulations spanning even a few seconds can require a number of time steps on the order of $1e6$, rendering repeated time accurate simulation computationally intractable.

It is in light of this difficulty that we instead shift attention to steady-state simulation of the monolithic hypersonic aerothermoelasticity system. Still, steady-state simulation for large nonlinear systems of equations such as the hypersonic aerothermoelasticity system may not be as simple as disregarding the temporal term. Eliminating the temporal term from the equation system may cause considerable difficulty in resolving complex features not present in the initial guess to

the steady state solution for iterative nonlinear solvers, frequently leading to divergence. Among such complex features for hypersonic aerothermoelasticity are the emergence of shocks of various kinds and shapes, thick boundary layers, and complex surface heat flux distributions.

Despite the computational cost of transient simulation compared to steady-state simulation, it is apparent that time integration methods contribute considerable regularization to linearizations of the nonlinear systems of equations realized under temporal discretization. For example, discretization of the temporal term can work to increase the diagonal dominance of the Jacobian of the nonlinear residual when Newton's method is applied to linearize the nonlinear algebraic system, greatly easing the solution of the Newton system.

One approach for reaching a steady-state simulation for nonlinear differential equations is Pseudo-Transient Continuation (PTC) [69]. PTC considers the model steady state problem

$$\mathbf{f}(\mathbf{u}) = \mathbf{0} \quad (4.4)$$

where \mathbf{f} encodes, e.g., the dynamics of a semi-discretized PDE with solution \mathbf{u} . The PTC algorithm adds a fictitious temporal term with a pseudo time variable τ resulting in the model initial value problem

$$\frac{\partial \mathbf{u}}{\partial \tau} = \mathbf{f}(\mathbf{u}) \quad (4.5)$$

The PTC algorithm computes the steady-state solution \mathbf{u} by solving (4.5) via the iteration

$$\left(\frac{1}{\Delta\tau_n} \mathbf{I} + \mathbf{J}(\mathbf{u}_n) \right) \delta\mathbf{u}_n = \mathbf{f}(\mathbf{u}_n) \quad (4.6a)$$

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \delta\mathbf{u}_n \quad (4.6b)$$

where n denotes the iteration index, $\Delta\tau_n$ is the n^{th} time step, \mathbf{J} is the Jacobian of \mathbf{f} , and $\delta\mathbf{u}_n$ is the solution increment to be computed by solving (4.6a). The PTC algorithm can be derived by employing Backward Euler time integration to (4.5) and solving the resulting algebraic system with a single-iteration Newton method. The initial guess \mathbf{u}_0 is specified by the analyst. It is desired, though not always necessary in practice, that \mathbf{u}_0 satisfy boundary conditions encoded in \mathbf{f} . The aim of PTC is to take the time step $\Delta\tau_n$ to be small in early iterations and gradually increase it throughout the execution of the algorithm. We note that in the limit of $\Delta\tau_n \rightarrow \infty$, the diagonal addition $\frac{1}{\Delta\tau_n} \mathbf{I}$ vanishes and (4.6) becomes a typical Newton solve for the steady-state problem (4.4).

It remains to specify the methods of incrementing the time between iterations. Various heuristics have been proposed to this end. In this work, we consider the switched evolution relaxation [69] approach given by

$$\Delta\tau_n = \Delta\tau_0 \frac{\|\mathbf{f}(\mathbf{u}_0)\|}{\|\mathbf{f}(\mathbf{u}_n)\|} \quad (4.7)$$

It is apparent (4.7) is equivalent to

$$\Delta\tau_n = \Delta\tau_{n-1} \frac{\|\mathbf{f}(\mathbf{u}_{n-1})\|}{\|\mathbf{f}(\mathbf{u}_n)\|} \quad (4.8)$$

We make a final addition to limit the size of the time step by finally considering

$$\Delta\tau_n^* = \min(\Delta\tau_n, \Delta\tau_{max}) \quad (4.9)$$

where, for stability purposes, we have specified a maximum time step which should not be exceeded by the SER method. Finally, in this work, we couple the PTC algorithm with polynomial cycling, where the order of the polynomial finite element solution space is increased between subsequent runs. In this approach, the steady-state solution is first computed using a zeroth order polynomial space. The computed solution is then used as the initial guess to the steady-state solution of a first order polynomial space. Polynomial cycling allows convergence of large physical features using lower-order polynomial spaces. Fine physical phenomena can then be resolved as the polynomial order is increased.

CHAPTER V

RESULTS

5.1 Methodology

Due to the complex nature of hypersonic aerothermoelasticity simulation, a small number of physical and numerical experiments have been conducted, providing limited opportunities for verification and validation of the presented monolithic approach. As described in previous chapters, many of the numerical experiments presented in the literature either rely heavily on results generated on simple, flat geometries to limit the number of important structural modes or employ either reduced-order models or simple empirical relationships for the computation of pressure loads and heat distributions.

It is under these concerns that we present a numerical experiment that, to the knowledge of the author, have not been performed in the relevant literature. Verification of the monolithic approach is instead demonstrated via verification of the constituent physics models using the method of manufactured solutions. Grid convergence studies of the constituent physics are also presented. Verification of the single-physics modeling methods in multiphysics systems have been performed in other multiphysics coupling studies [110, 126].

The method of manufactured solutions (MMS) is a simulation code verification technique where a manufactured solution to a simulated PDE is inserted into the PDE. In general, the manufactured solution will not satisfy the PDE and will result in the production of a source term. This source

term can then be added to the code and the simulation is executed. The computed result for a correctly implemented code will be the manufactured solution. Boundary and initial conditions in MMS are simply the manufactured solution evaluated at the domain boundaries and the initial time, respectively.

5.2 Verification

MMS simulations for fluid, solid, and thermal simulations are performed on a square structured domain with a side length of 3π m. We focus on verifying the results of the individual solvers instead of the coupled equation system due to the considerable difficulty in determining an appropriate source term. Several MMS simulations are performed on successively refined grids to facilitate mesh convergence studies. Each grid is a structured grid of triangular elements. The number of elements is quadrupled between each grid by doubling the number element in x and y directions.

Transient simulations are employed for all three verification test cases using a BDF(1) time integrator which was found to provide more accurate results than higher order BDF methods. Polynomials of order 3 are employed for all MMS verification cases. The hybridized DG method realizes a cost savings by performing static condensation on the Newton system computed after spatio-temporal discretization of the considered PDE producing a reduced linear system. In all MMS verification cases, this reduced linear system is solved using a GMRES solver coupled with an ILU preconditioner with two levels of fill-in. The Krylov vectors computed by the GMRES method are discarded and restarted every 50 iterations. The Newton method is determined to have converged within a time step if the L^2 norm of the full nonlinear residual is found to be less than a

tolerance of $1e-12$ or if 25 Newton iterations have elapsed. Finally, all variables in the verification study are taken to be non-dimensional.

Table 5.1: Degrees of freedom for each MMS mesh. The number of degrees of freedom must be multiplied by the number of components solved for by each of the physical equations to get an accurate degree of freedom count in each simulation. These values are 4, 4, and 1 for the fluid, solid, thermal equation systems, respectively.

Mesh	Element unknowns	Trace unknowns
4	320	44
8	1,280	832
16	5,120	20480
32	20,480	12,544
64	81,920	49,664

5.2.1 Fluid

A MMS verification test was performed on the compressible Navier-Stokes equations discretized with a hybridized DG method as demonstrated in Ch. III. The configuration for the simulation is presented in Table 5.2.

The manufactured solution for the compressible Navier-Stokes equations written in terms of primary variables is given in (5.1). The same manufactured solution is rewritten in terms of the conservation variables in (5.2). In the results that follow, we take T_0 to be unity. The exact solutions for the velocity components v_i at time 0.5 s are presented in Figure 5.1.

Table 5.2: Configuration for the compressible Navier-Stokes MMS verification study.

Title	Symbol	Value
Domain height	H	3π m
Domain width	L	3π m
Polynomial order	k	3
Density	ρ	1 kg m^{-3}
Pressure	p	1 Pa
Length scale	\bar{L}	1 m
Mach number	Ma	1
Reynolds number	Re	$1.10\text{e}5$
Specific heat ratio	γ	1.2
Specific heat	c_v	$5 \text{ J kg}^{-1} \text{ K}^{-1}$
Dynamic viscosity	μ	0.7 Pa s
Prandtl number	Pr	0.7
Time step	Δt	$9.13\text{e-}3$ s

$$\rho = 1$$

$$v_1 = \sin x \sin t \tag{5.1}$$

$$v_2 = \cos y \sin t$$

$$T = \sin t + T_0$$

$$\begin{bmatrix} \rho \\ \rho v_1 \\ \rho v_2 \\ \rho E \end{bmatrix} = \begin{bmatrix} 1 \\ \sin x \sin t \\ \cos y \sin t \\ \frac{1}{2} \sin^2 t \left(\sin^2 x + \cos^2 y \right) + \sin t + T_0 \end{bmatrix} \tag{5.2}$$

Inserting (5.2) into the non-dimensional compressible Navier-Stokes system yields the following source term.

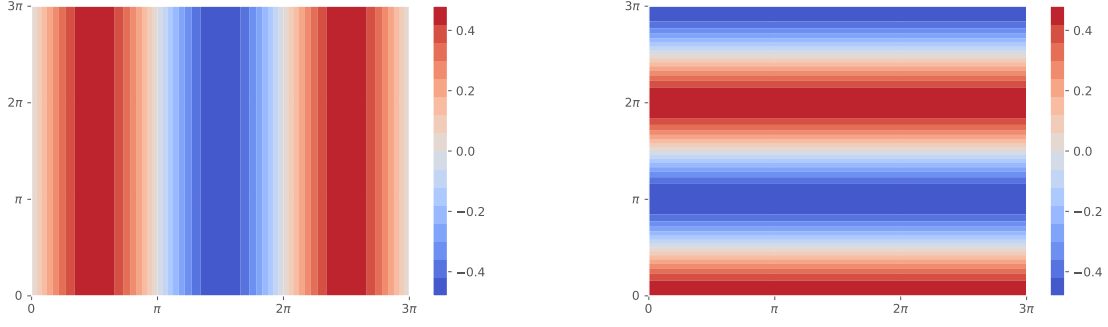


Figure 5.1: Manufactured solutions for the fluid velocity components v_1 and v_2 , respectively, evaluated at $t = 0.5s$.

$$\mathbf{S} = \begin{bmatrix} \sin t (\cos x - \sin y) \\ \sin x \left(2 \sin^2 t \cos x - \sin^2 t \sin y + \cos t \right) + \frac{1}{\text{Re}} \frac{8}{3} \sin t \sin x \\ \cos y \left(\sin^2 t \cos x - 2 \sin^2 t \sin y + \cos t \right) + \frac{1}{\text{Re}} \frac{8}{3} \sin t \cos y \\ \rho E_s \end{bmatrix} \quad (5.3)$$

where ρE_s is given by

$$\begin{aligned} \rho E_s = & \gamma \left(T_0 \sin t + \sin^2 t \right) (\cos x - \sin y) \\ & + \frac{1}{2} \sin^3 t \left(-\sin^2 x \sin y + 2 \sin^2 x \cos x - 2 \sin y \cos^2 y + \cos x \cos^2 y \right) \\ & + \sin t \left(\sin^2 x \cos t + \cos t \cos^2 y \right) \\ & + \frac{1}{\text{Re}} \frac{4}{3} \sin^2 t \left(2 \sin^2 x - 2 \sin^2 y + \sin y \cos x - 2 \cos^2 x + 2 \cos^2 y \right) \\ & + \cos t \end{aligned}$$

We note the considerable nonlinearity in the resulting source term. Indeed, retrieving the manufactured solution presented a challenge for the implementation. It is expected that this is due

to inaccuracies in the integration rules used to form the Newton system and time integrator. It was found that decreasing the time step yielded more accurate solutions, but greatly increased the computational burden. The time step was taken to be $\Delta t = 9.13e - 3$ s which provided a good trade off between accuracy and expense.

The solutions computed by the hybridized DG implementation are presented in Figure 5.2 and Figure 5.3. Mesh convergence is presented in Figure 5.4. It is found that refining the meshes results in a monotonic decrease in the error of all primary variables.

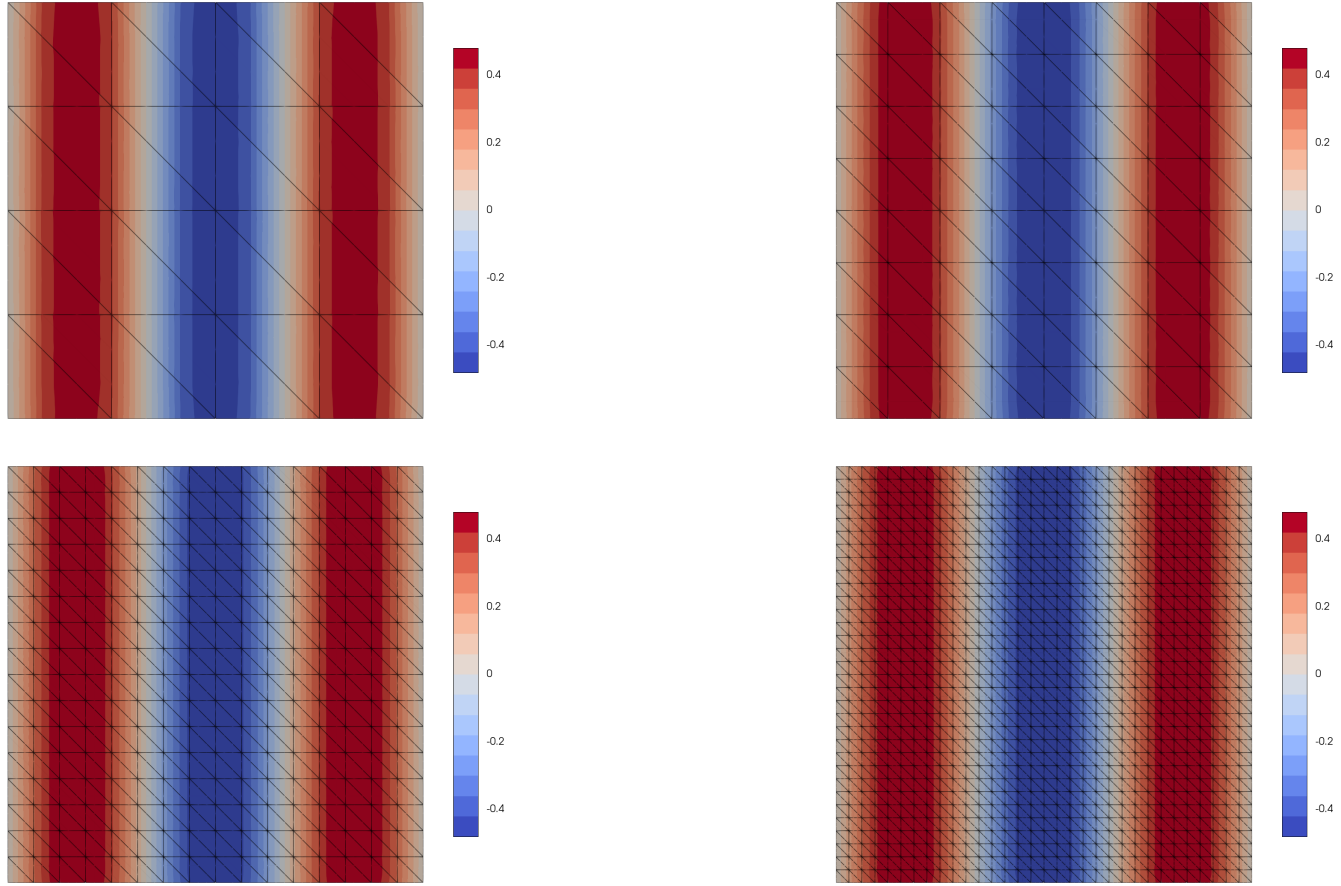


Figure 5.2: Solutions computed on the considered meshes for the v_1 fluid velocity component evaluated at time $t = 0.5$ s.

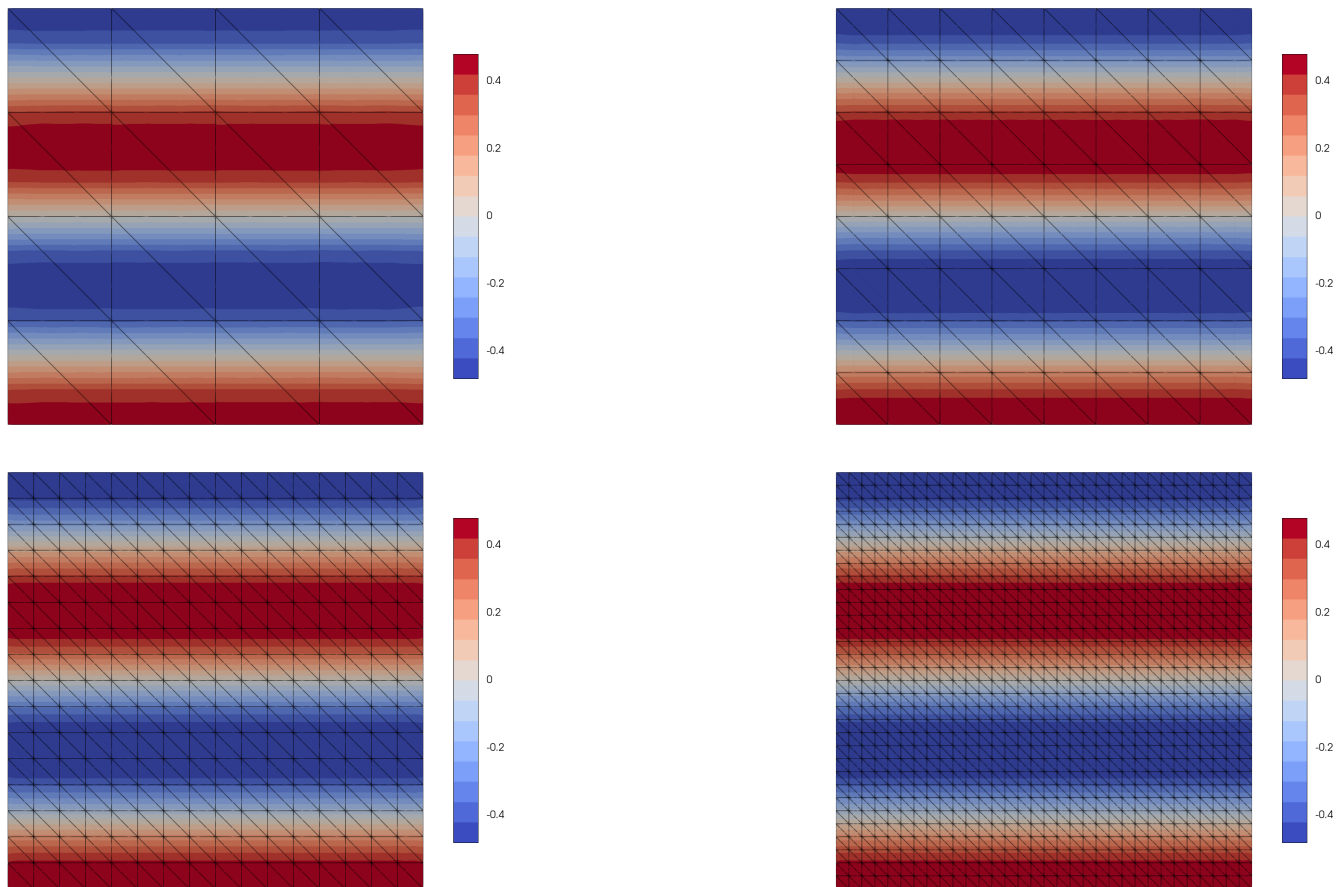


Figure 5.3: Solutions computed on the considered meshes for the v_2 fluid velocity component evaluated at time $t = 0.5$ s.

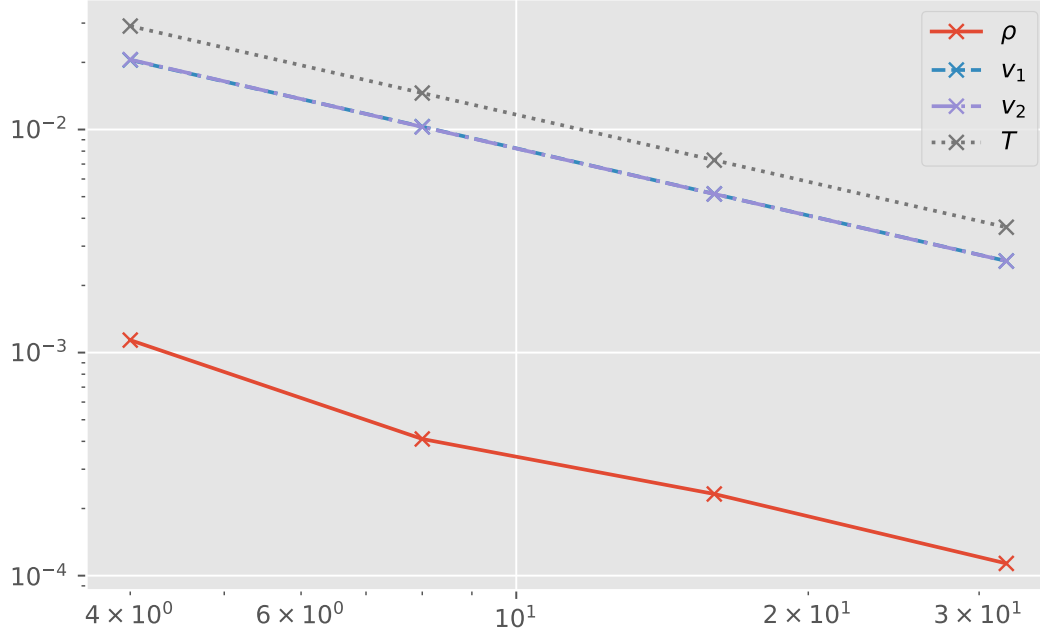


Figure 5.4: Convergence of the L^2 error between the manufactured and computed compressible Navier-Stokes solutions for the considered meshes computed at time $t = 0.5$ s.

5.3 Linear elasticity

We consider a MMS verification for the linear elasticity equations. The configuration for the MMS simulation is given in Table 5.3.

Table 5.3: Configuration for the linear elasticity MMS verification study.

Title	Symbol	Value
Domain height	H	3π m
Domain width	L	3π m
Polynomial order	k	3
Density	ρ	1 kg m^{-3}
Lamé's first parameter	λ	0.4 Pa
Lamé's second parameter	μ	0.4 Pa
Time step	Δt	0.1 s

The manufactured solution for the solid displacement field is given by

$$\begin{aligned} u_1 &= \sin x \cos y (1 - e^{-t}) \\ u_2 &= \sin y \cos x (1 - e^{-t}) \end{aligned} \tag{5.4}$$

The manufactured solution is presented in Figure 5.5. We recall that the linear elasticity system is expressed as a first order in time system of equations. Therefore, the source term for the linear elasticity equations is applied to the velocity equation and not the displacement equation. Plugging in the manufactured solution into linear elasticity equations yield the following source term given in (5.5).

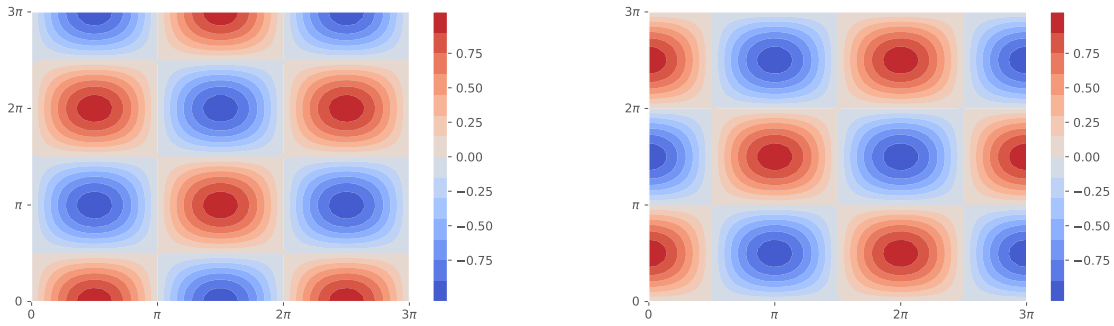


Figure 5.5: Manufactured solutions for the solid displacement components u_1 and u_2 , respectively, evaluated at $t = 20$ s.

$$\begin{aligned} v_1 &= -\frac{1}{\beta} ((2\lambda + 4\mu) (1 - e^{-t}) - 1) e^{-t} \sin x \cos y \\ v_2 &= -\frac{1}{\beta} ((2\lambda + 4\mu) (1 - e^{-t}) - 1) e^{-t} \sin y \cos x \end{aligned} \tag{5.5}$$

Finally, the computed solutions for the solid displacement components are presented in Figure 5.6 and Figure 5.7, respectively. The convergence of the computed solution is presented in Figure

5.8. It is found the refining the meshes results in monotonic decrease of the solid displacement error.

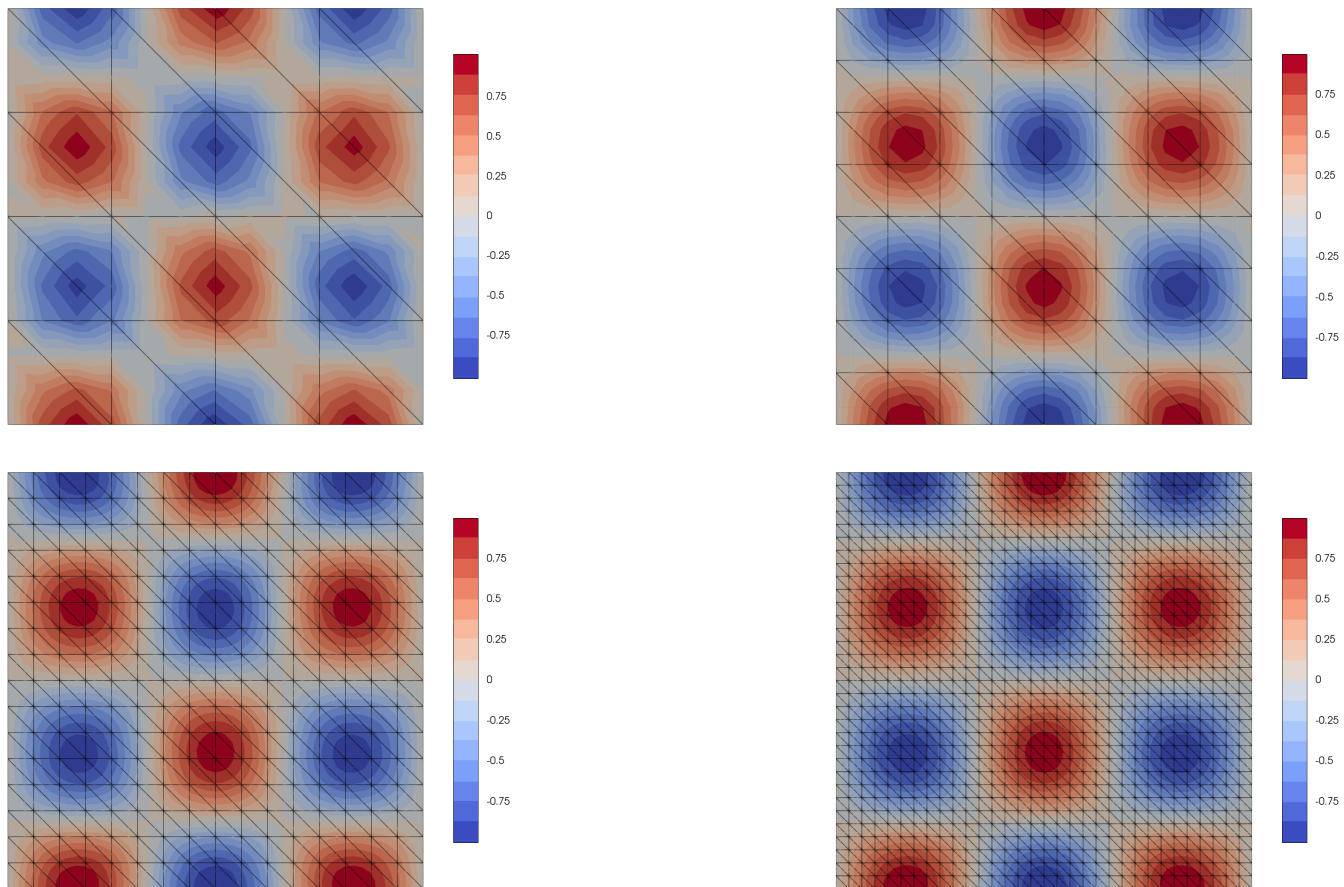


Figure 5.6: Solutions computed on the considered meshes for the u_1 solid displacement component evaluated at time $t = 20$ s.

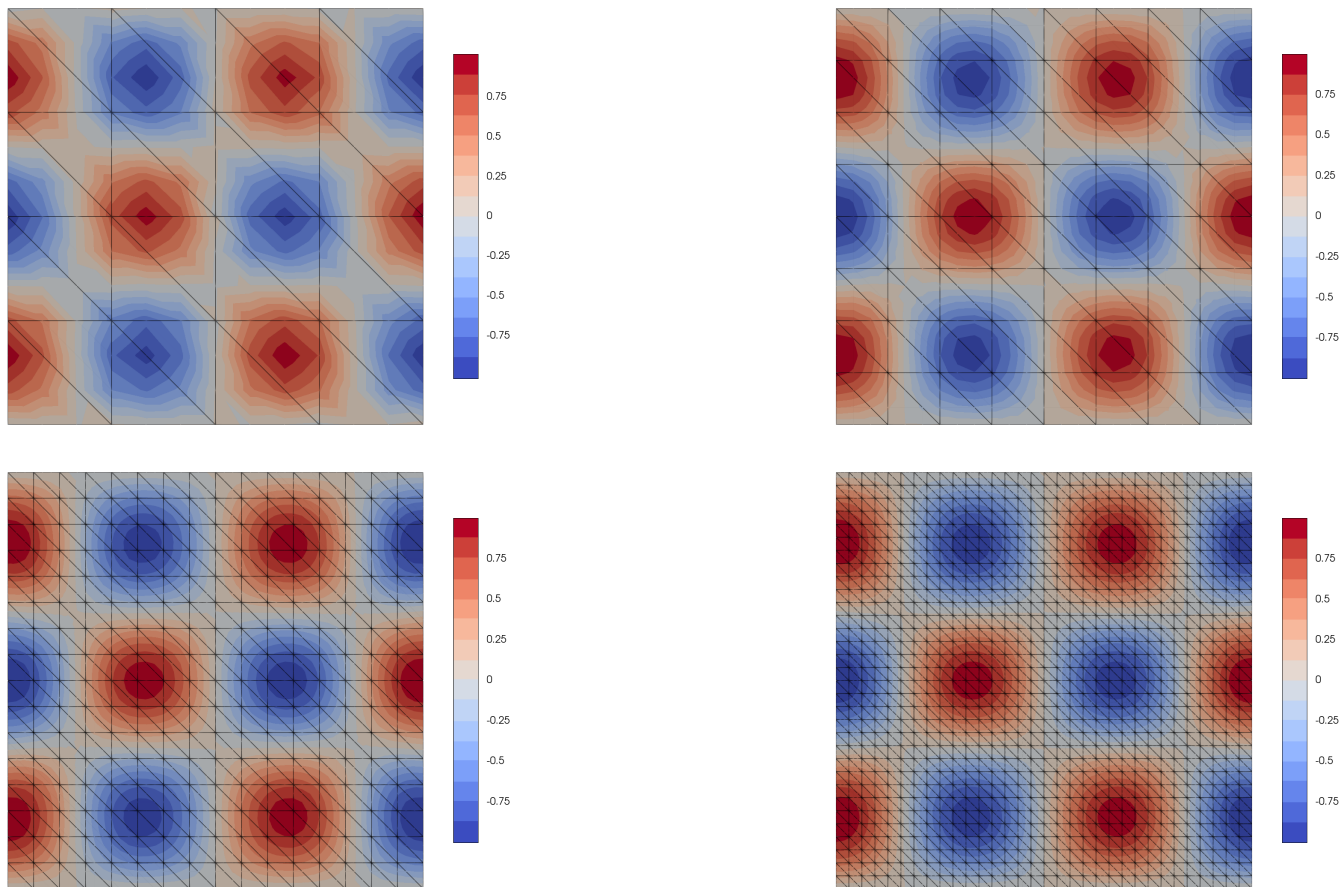


Figure 5.7: Solutions computed on the considered meshes for the u_2 solid displacement component evaluated at time $t = 20$ s.

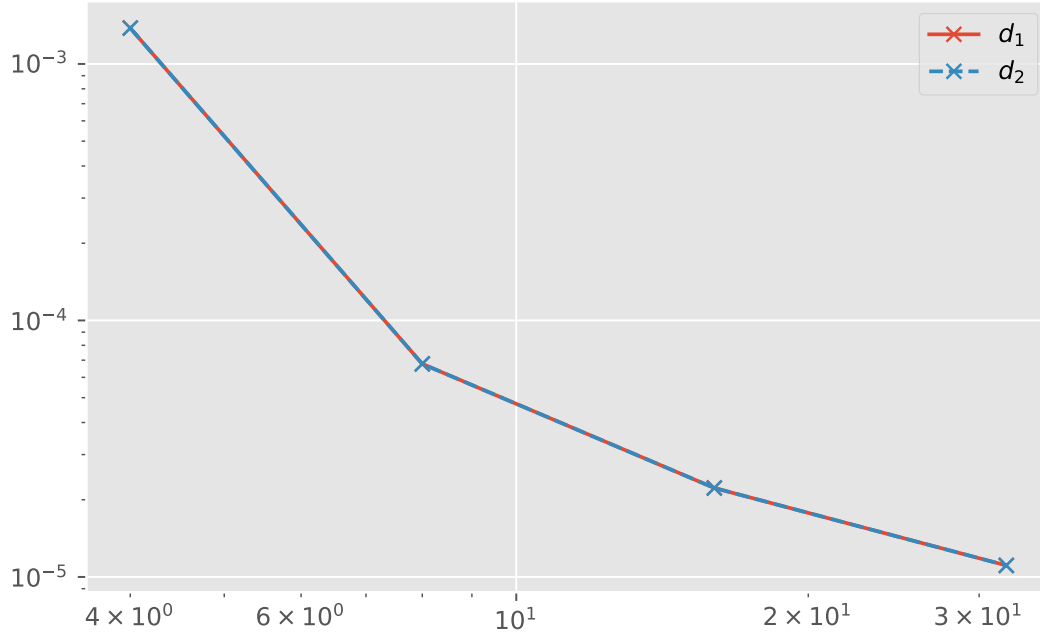


Figure 5.8: Convergence of the L^2 error between the manufactured and computed linear elasticity solutions for the considered meshes computed at time $t = 2.0$ s.

5.4 Thermal

Finally, we present a MMS verification of the heat equation. The configuration for the MMS simulation is presented in 5.4.

Table 5.4: Configuration for the heat equation MMS verification study.

Title	Symbol	Value
Domain height	H	3π m
Domain width	L	3π m
Polynomial order	k	3
Density	ρ	1 kg m^{-3}
Length scale	\bar{L}	1 m
Time step	Δt	1 s

The manufactured solution for the heat equation is given in (5.6). The manufactured solution is presented in Figure 5.9.

$$T = \sin x \cos y \sin t \quad (5.6)$$

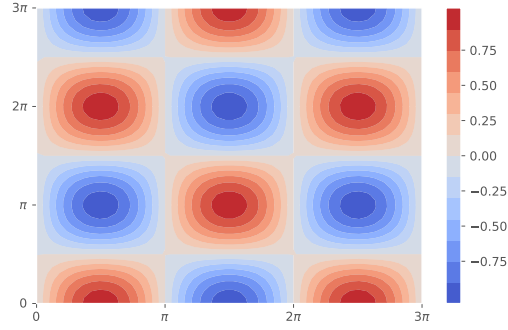


Figure 5.9: Manufactured solution for the temperature T evaluated at $t = 1.6$ s.

Inserting the manufactured solution into the heat equation produces the source term given in (5.7).

$$(2\tilde{\kappa} \sin t + \cos t) \sin x \cos y \quad (5.7)$$

The computed temperature field is presented in Figure 5.10. The convergence of the computed solution is presented in Figure 5.11. It is found the refining the meshes results in monotonic decrease of the error. Indeed, all single-physics problems were well resolved by the present hybridized DG solver. For this reason, the implementation is considered to be verified.

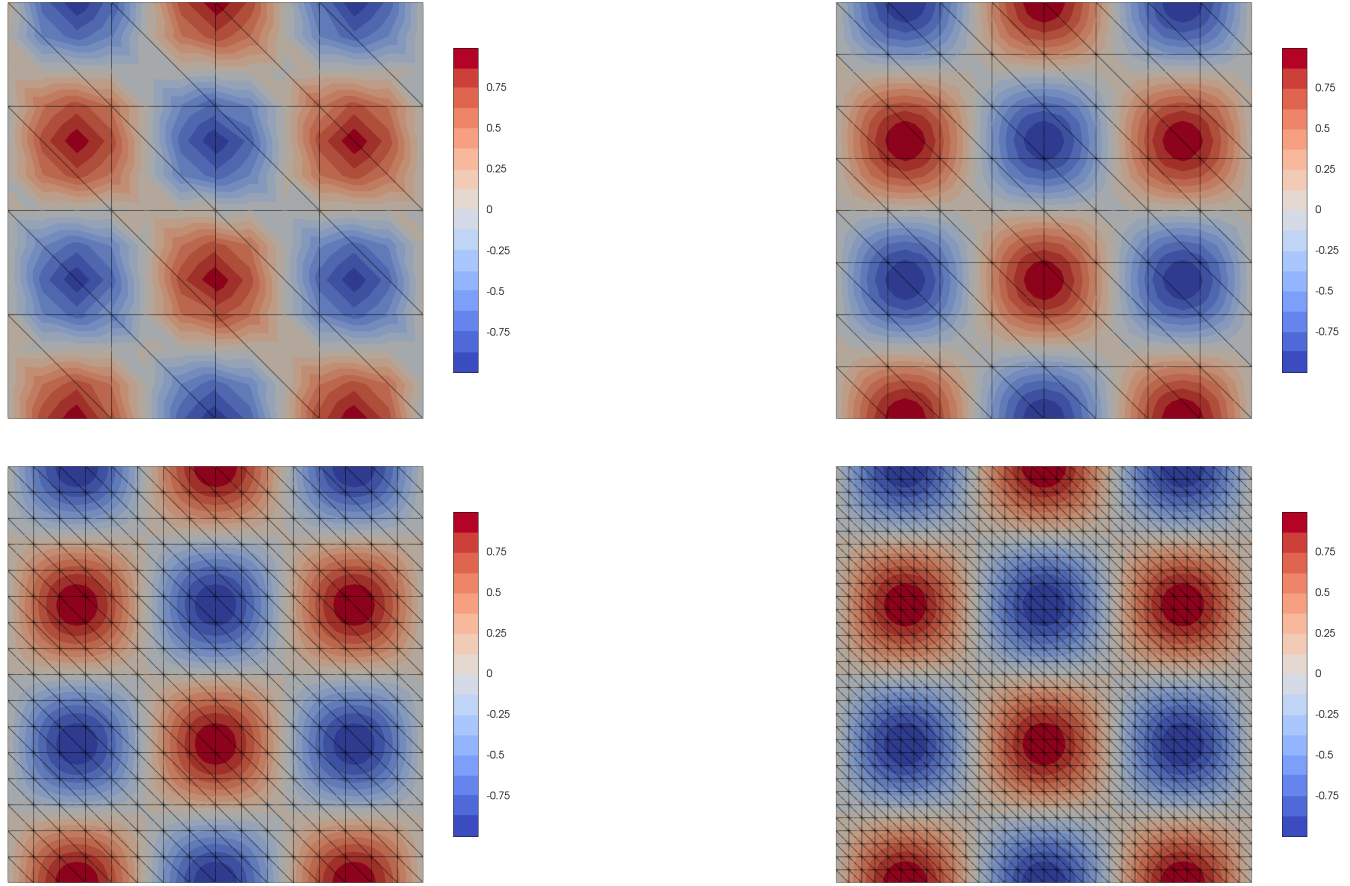


Figure 5.10: Solutions computed on the considered meshes for the temperature T evaluated at time $t = 1.6$ s.

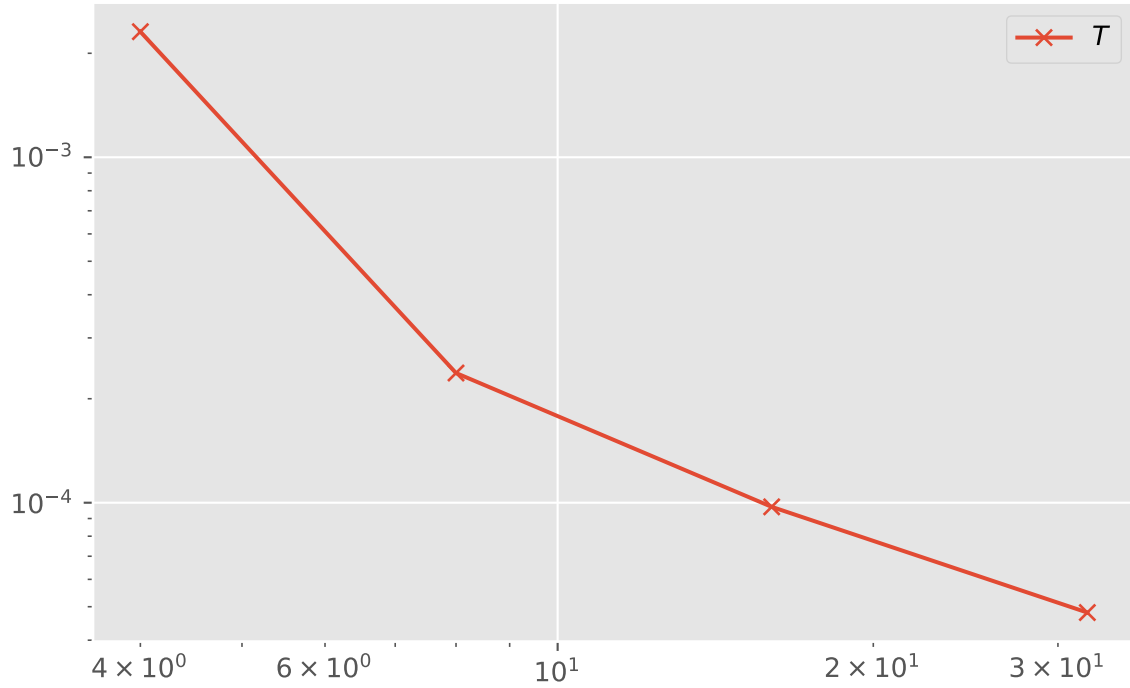


Figure 5.11: Convergence of the average L^2 error between the manufactured and computed heat equation solutions for the considered meshes computed at time $t = 1.6s$.

5.5 Hypersonic aerothermoelasticity on a cylinder

We consider simulation of hypersonic flow around a half cylinder. The blunt geometry poses a challenge to compressible flow solvers due to the formation of a bow shock in hypersonic flows, resulting in a complex velocity distribution downstream the shock. Flow near the cylinder slows to subsonic velocities and accelerates to supersonic velocities on the sides of the cylinder. Further, compression of the fluid on the surface of the cylinder causes a large increase in pressure and temperature. We note that while hypersonic flow around a half cylinder has been considered in

the literature [21], to the author's knowledge no aerothermoelasticity studies for this geometry has been performed.

5.5.1 Problem description

The half cylinder geometry is presented in Figure 5.13. The radius of the cylinder is 0.2 m. The maximum height and width of the domain are 8 radii and 3 radii, respectively. Note that a single mesh is employed to capture the fluid, thermal, and solid evolution. The domain was decomposed into a fluid domain Ω_f and solid domain Ω_s via a mesh consisting of 9,558 elements. The hypersonic aerothermoelasticity system is solved on this mesh using polynomial elements of order 0, resulting in 81,693 globally coupled unknowns. The mesh employed in this study is presented in Figure 5.12. Thermal evolution is performed within both domains. Thermal conduction and convection are performed by the energy equation in the compressible Navier-Stokes equations in the fluid domain, while thermal conduction is modeled by the heat equation in the solid domain. The temperature distribution within the solids adds additional deformation to the solid through an added thermal strain term.

PTC is employed to compute the steady-state hypersonic aerothermoelasticity solution. The initial condition for the fluid domain is set to the freestream condition. Zero displacements are initially prescribed for the fluid and solid domains. The solid is initially taken to be 1,000 K. A reference temperature of 300 K in the solid is prescribed for the generation of thermal strains. We note that since the PTC algorithm advances pseudo time, the initial condition does not have to be physical solution; it needs only to be "sufficiently close" to the steady-state solution. The fluid domain is modeled using the properties of air at an altitude of approximately 50km. The initial

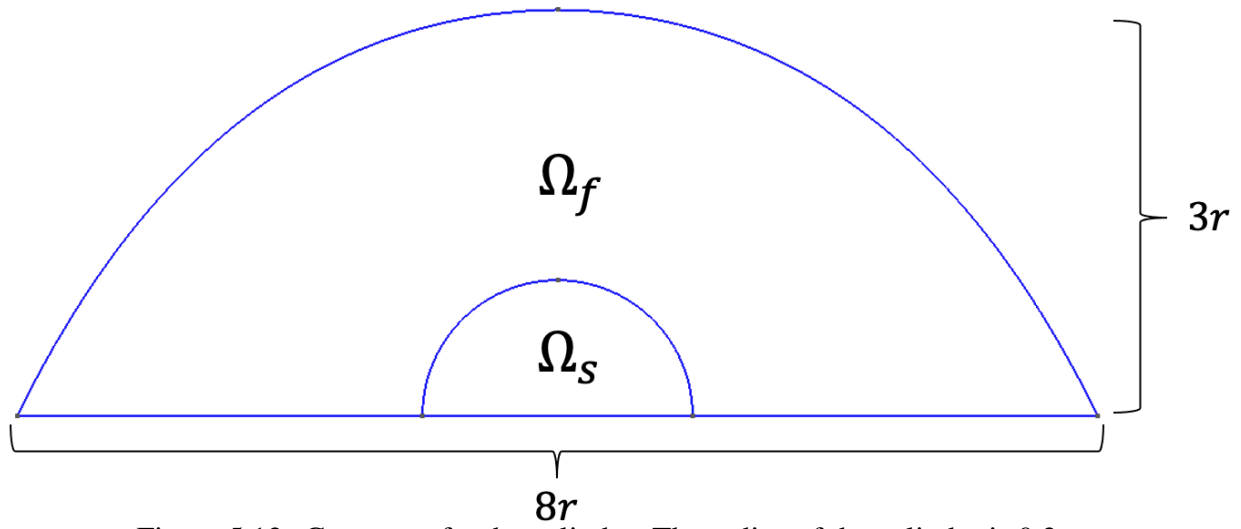


Figure 5.12: Geometry for the cylinder. The radius of the cylinder is 0.2 m.

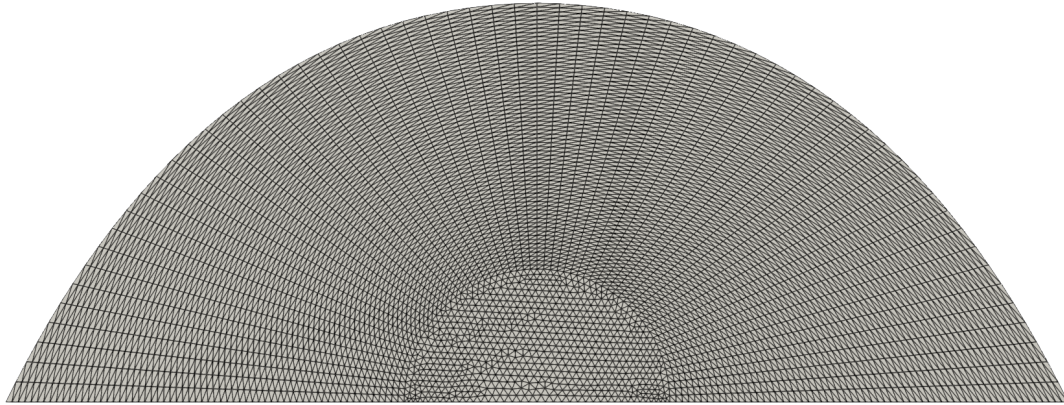


Figure 5.13: Mesh employed for the simulation of the cylinder.

condition and flow configuration for the fluid domain are presented in Table 5.5. The solid domain is modeled as an Inconel alloy. The configuration for the solid domain is given in Table 5.6.

A freestream boundary condition is prescribed on the inlet. The bottom flow outlet boundary simply extrapolates the flow state. The bottom solid boundary condition is set to a no-displacement condition. The velocity coupling condition on the fluid-structure boundary enforces a no-slip condition. The fluid displacements on the fluid-structure boundary are extrapolated from the displacements computed by the elastodynamics equations.

The compressible Navier-Stokes equations and the constraints imposed by the fluid on the coupling interface are written in terms of the ALE transformation. Displacement degrees of freedom are added to the fluid equation system and solved for using the elastostatics equation. The fluid mesh velocity is approximated using a first-order finite different scheme computed using the fluid displacements at the current and previous time steps. For the fluid mesh deformation, the non-dimensional Lamé parameters are both set to 0.1 which distributed the stress throughout the domain without resulting in the domain being overly elastic.

Table 5.5: Configuration of the fluid domain. Values with an ∞ superscript denote freestream values which are used as an initial condition for the PTC procedure.

Title	Symbol	Value
Density	ρ_f^∞	0.4 kg m ⁻³
Pressure	p_f^∞	2.5 kPa
Mach number	Ma^∞	5
Velocity	v_f^∞	1479.0 m s ⁻¹
Length scale	\bar{L}	0.2 m
Reynolds number	Re^∞	6.5e5
Temperature	T_f^∞	217.7 K
Specific heat	C_v	717.6 J kg ⁻¹ K ⁻¹
Specific heat ratio	γ	1.4
Dynamic viscosity	μ	1.8e-5 Pa s
Prandtl number	Pr	0.71

Table 5.6: Configuration for the solid domain. A value of 1,000 K was chosen for the initial temperature of the solid for the PTC procedure.

Title	Symbol	Value
Density	ρ_s	8.2e3 kg m ⁻³
Lamé's first parameter	λ_s	10.8 GPa
Lamé's second parameter	μ_s	7.8 GPa
Length scale	\bar{L}	0.2 m
Temperature	T_s	1e3 K
Reference temperature	$T_{0,s}$	3e2 K
Thermal conductivity	κ	11.6 W m ⁻¹ K ⁻¹
Specific heat	C_p	450 J kg ⁻¹ K ⁻¹
Fourier number	Fo	1

5.5.2 Solution procedure

The monolithic equation system is solved with a GMRES method. The Krylov vectors are restarted every 50 GMRES iterations. The GMRES algorithm is considered converged if the relative error in the linear residual is less than $1e - 8$, the absolute error is less than $1e - 12$, or if 350 GMRES iterations have elapsed. An ILU(2) preconditioner was employed with reasonable success. The initial time step was taken to be $1e - 3$ pseudo time units. The PTC method is considered to have converged if 500 units of pseudo time have elapsed. It was enforced that a pseudo time step of 100 pseudo time units could not be exceeded. The steady-state solution for fluid-only variables are presented in Figure 5.14 and the solution for variables shared by the fluid and solid domain are presented in Figure 5.15. Finally, values for pressure, temperature, and density at the stagnation point of the cylinder's surface are presented in Table 5.7

Certain complications with this simulation did arise. It was found that the PTC method is sensitive to the choice of the initial time step. A time step that was too large was unable to resolve the physics correctly, while a time step that was too small prevented satisfaction of the multiphysics

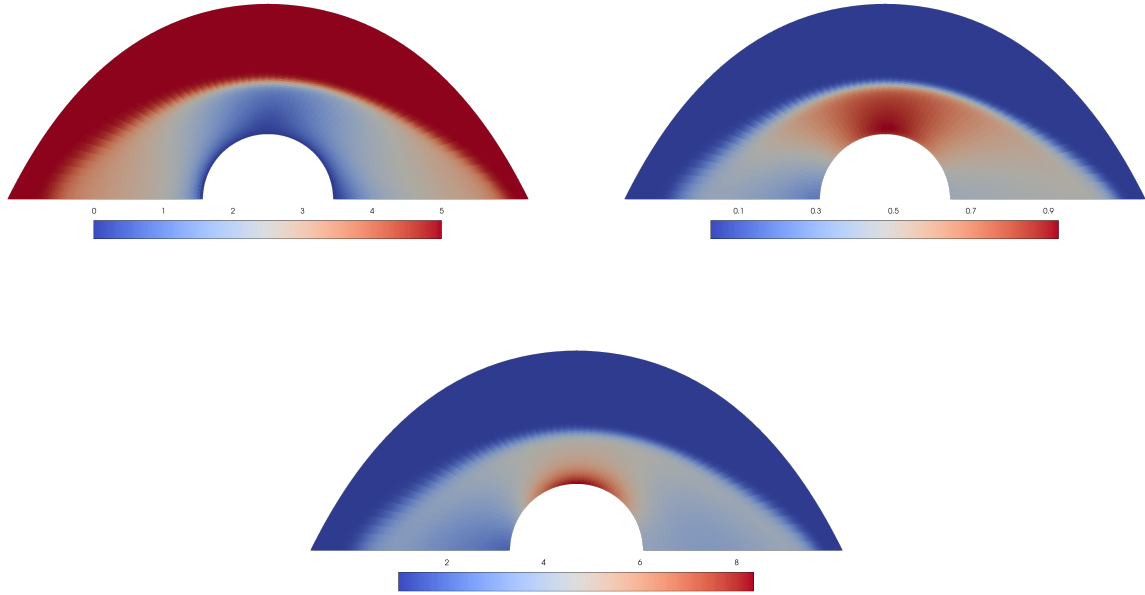


Figure 5.14: Steady-state solution for fluid-only variables on a cylinder subject to Mach 5 flow. The solution for Mach number (top left), non-dimensional pressure (top right) and non-dimensional density (bottom) are presented.

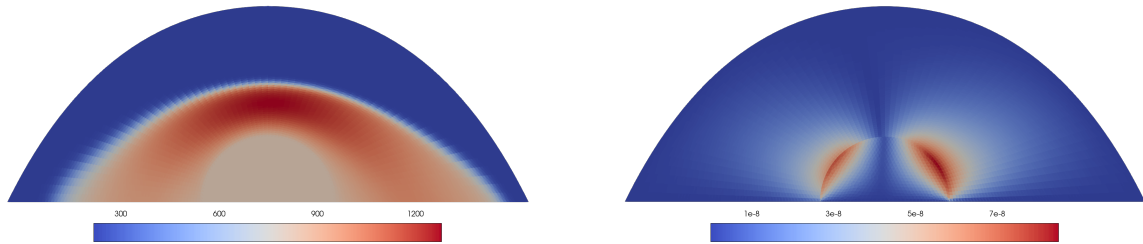


Figure 5.15: Steady-state solution for variables shared by the fluid and solid domain on a cylinder subject to Mach 5 flow. The solution for temperature (K) (left) and displacement (m) (right) are presented.

constraints within a “reasonable enough” time frame and lead to divergence. The solver at times struggled satisfy to the constraints on the coupling interface. The flux-based constraints were scaled to cause the solver to satisfy the flux-based constraints in early PTC iterations. The state-

Table 5.7: Values of density, pressure, and temperature at the stagnation point of the cylinder's surface.

Title	Non-dimensional value	Dimensional value
Pressure	0.887	7.761e4 Pa
Temperature	0.271	8.260e2 K
Density	8.32	3.328e-1 kg m ⁻³

based constraints posed less of a problem and were easily satisfied throughout the PTC iterations. Further, like with many mesh-based methods, a certain level of grid-dependent effects appear in the flow field, i.e. some solution asymmetry on the left and right sides of the domain appears due to the orientation of the triangular elements.

The constituent physics in hypersonic aerothermoelasticity evolve in vastly different time scales and pose a considerable challenge for the resolution of transient simulation for both monolithic and partitioned coupling approaches. It was found that the considerable difference in these time scales also posed a problem for steady state simulations achieved with PTC. Indeed, employing pseudo time integration via PTC was able to resolve the smallest time scale (i.e. the fluid time scale) well, but evolution of the remaining physics with larger time scales was delayed. For example, it was originally assumed that the solid was initially at the same temperature as the freestream fluid. In the early stages of the PTC algorithm, the thermal coupling condition would cause the fluid near the solid to remain at the freestream temperature, despite the surrounding flow correctly heating up. The conduction time scale in the solid was too slow to cause any notable change in temperature in the solid, resulting in the solid acting as cool Dirichlet condition for the fluid. To remedy this issue, it was necessary to artificially decrease the thermal time scale greatly by multiplying the thermal time scale by a sufficiently small value. We justify this decrease by noting that the steady state condition for the solid geometry considered is simply the solid heated to the final fluid temperature.

Finally, the presented approach successfully converged all physical variables defined on the fluid and solid domains and on the fluid-solid interface. Small displacements are expected for this test case due the solid domain being completely dense with material, lacking any holes. Despite the displacements being small for this test case, the effect of the coupling influenced the near-body solution. In particular, conduction in the cylinder greatly effected the temperature and, correspondingly, the energy distribution in the fluid near the surface. More pliable structures subject to greater temperatures and pressures are expected result in greater deformation. Despite the lack of a shock capturing scheme, the method was able to reasonably capture the bow shock.

5.6 Hypersonic aerothermoelasticity on a hollow cylinder

While the previous numerical experiment demonstrated the monolithic multiphysics coupling results on a geometry of interest, little deformation was actually observed. This is due largely to the cylinder being densely packed with material which prevented any significant displacement of the solid. To further demonstrate the capabilities of the presented method, the test case is repeated using a hollow cylinder.

5.6.1 Problem description

The same cylinder mesh is employed in this test with the exception that the cylinder most of the material inside the cylinder is removed. Indeed, the material within an inner cylinder with a radius taken to be 95% of the original cylinder's radius is removed. After the removal of the inner material, what remains is a thin circular arc structure. The mesh for this test case is presented in Figure 5.16. The configuration of the hollow cylinder test case is given Table 5.8 and 5.9. The fluid configuration seeks to model the fluid as air at sea level. Notably, the Mach number is taken to be

7 and the Lamé parameters are reduced. The same boundary conditions and linear solver settings as employed for the previous cylinder case are employed for the hollow cylinder with the exception that a displacement extrapolation condition is applied on the lower boundary of the hollow cylinder.

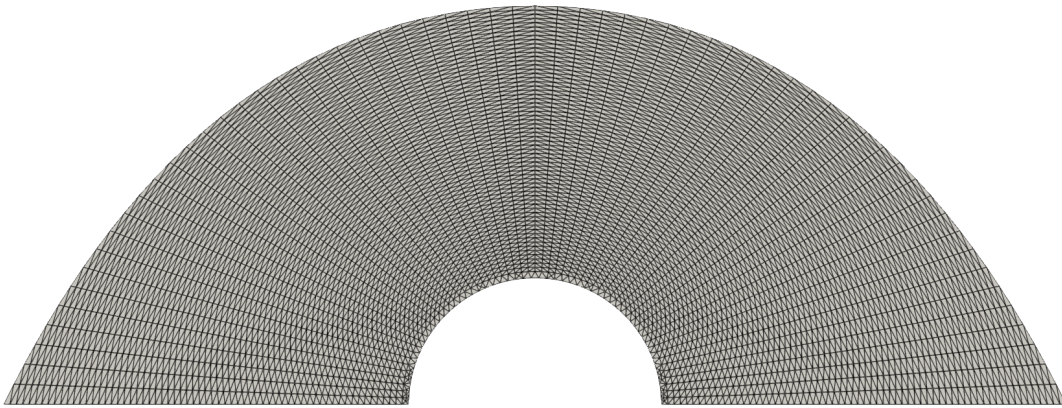


Figure 5.16: Mesh employed for the simulation of the hollow cylinder

The hollow cylinder was able to deform significantly more than the dense cylinder due to lack of inner material. The test case was performed in a higher Mach flow which causes a significant increase in fluid and solid temperature. Since the flow models air at sea level, the static pressure is found to be significantly greater. The combined effects of the fluid pressure and solid temperatures produce a large displacement within the structure. Decreasing the Lamé parameters is also found to greatly increase the deformation of the solid. A significant increase in temperature is also observed compared to the dense cylinder case, due largely to the elevated Mach number.

Table 5.8: Configuration of the fluid domain for the hollow cylinder case. Values with an ∞ superscript denote freestream values which are used as an initial condition for the PTC procedure.

Title	Symbol	Value
Density	ρ_f^∞	1.225 kg m ⁻³
Pressure	p_f^∞	101.325 kPa
Mach number	Ma^∞	7
Velocity	v_f^∞	2382.0 m s ⁻¹
Length scale	\bar{L}	0.2 m
Reynolds number	Re^∞	3.2e7
Temperature	T_f^∞	288.1 K
Specific heat	C_v	717.6 J kg ⁻¹ K ⁻¹
Specific heat ratio	γ	1.4
Dynamic viscosity	μ	1.8e-5 Pa s
Prandtl number	Pr	0.71

Table 5.9: Configuration for the solid domain for the hollow cylinder case. A value of 3,000 K was chosen for the initial temperature of the solid for the PTC procedure.

Title	Symbol	Value
Density	ρ_s	4e3 kg m ⁻³
Lamé's first parameter	λ_s	576.9 MPa
Lamé's second parameter	μ_s	384.6 MPa
Length scale	\bar{L}	0.2 m
Temperature	T_s	3e3 K
Reference temperature	$T_{0,s}$	3e2 K
Thermal conductivity	κ	11.6 W m ⁻¹ K ⁻¹
Specific heat	C_p	450 J kg ⁻¹ K ⁻¹
Fourier number	Fo	1

Despite the increase in complexity of the hollow cylinder test case, all physical equations were found to converge to the desired tolerance. However, certain concerns of validation of this solution remain. In the lack of availability of necessary truth data, it is impossible to validate the computed results. Still, both the dense and hollow cylinder results provide strong evidence that hybridized DG is well-suited to solve the hypersonic aerothermoelasticity system.

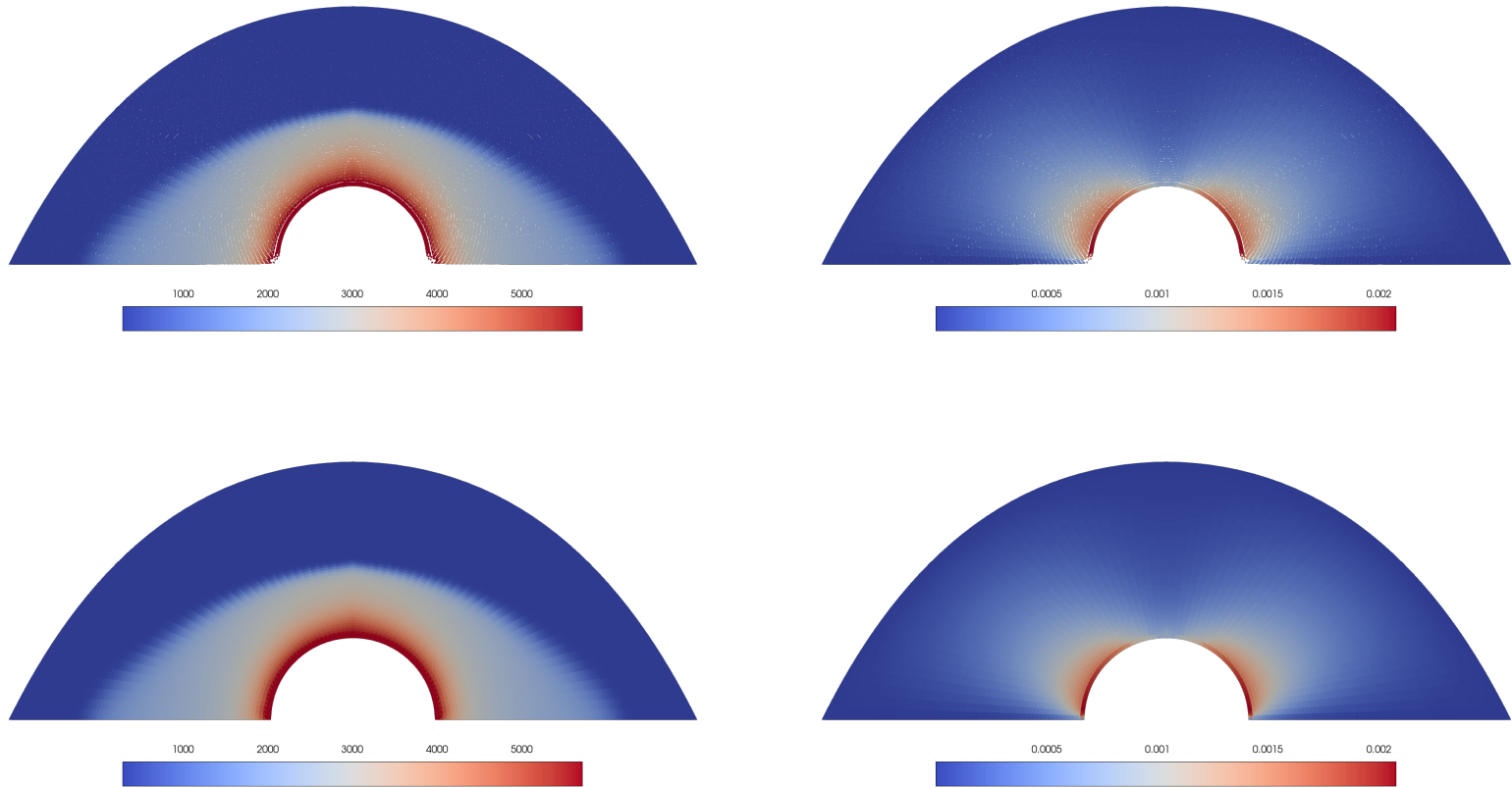


Figure 5.17: Steady-state solution for temperature (K) (left column) and displacement (m) (right column) variables for with displacements scaled by 10 (top row) and undeformed (bottom row). The discontinuous nature of the solution spaces are seen in the deformed visualizations.

CHAPTER VI

CONCLUSION

In this work we presented a monolithic coupling approach for the hypersonic aerothermoelasticity system. A hybridized DG code was developed for the simulation of arbitrary PDEs. The compressible Navier-Stokes, linear elasticity, and heat equation were implemented and coupled through constraints enforced on the interface. Non-dimensionalization of each physical system and the coupling constraints were implemented to ensure robustness of the implementation. The individual solvers were verified by the method of manufactured solutions and monotonic decreases in solution error were observed under mesh refinement. Coupled hypersonic aerothermoelasticity simulation was successfully performed on a cylinder and convergence of all physical variables was observed. It was found that hybridized DG methods are equipped to simulate multiphysics problems which transfer data across a coupling interface.

Monolithic multiphysics simulation using a hybridized DG method had been previously performed in a single study in [109] and later improved in [111] for simulation of low Mach FSI. While the present study takes inspiration from this previous work, this present study extends the multiphysics hypersonic and hybridized DG literature in several developments, namely 1) devising and implementation of three-field multiphysics coupling procedure, 2) resolution of coupled high speed flows domains which give rise to complex flow phenomena including flow compression and

shock formation, and 3) presentation of a general framework for monolithic multiphysics coupling for hybridized DG methods.

Several accomplishments with respect to the hypersonic aerothermoelasticity research are worth mentioning. First, the present study represents a novel approach for the accurate simulation of hypersonic aerothermoelasticity. The prevailing literature considers transient simulation of the hypersonic aerothermoelasticity system using empirical models and ROMs using staggered coupling approaches. This work improves on the current state-of-the-art by employing a high-fidelity simulation approaches for the constituent physics to ensure single-physics accuracy and a monolithic coupling scheme to prevent any spatio-temporal inaccuracies associated with the coupling scheme. While a steady state approach was pursued in this work, transient approaches can also be pursued with the presented approach.

Still, there remains many avenues for future work for multiphysics simulation of hypersonic aerothermoelasticity. With regards to the present implementation, it is necessary to consider three-dimensional flows to simulate the aerothermoelastic effects on more complex geometries. Implementation of other element geometries, e.g. quadrilaterals, are expected to further reduce the computational cost of the present solver since they typically require fewer unknowns and increase the accuracy for some simulations by removing a certain level of grid dependence. For the sake of simplicity, the present implementation does not consider thermally-varying material properties. Modeling material property degradation is a simple addition that is expected to result increased accuracy of the implementation. Extending the implementation to run on distributed architectures would enable simulations on more complex geometries. Finally, it was found that enforcing the interface constraints in the Newton solver was not always straightforward in practice. It may be

necessary in future work to reframe the current approach as a constrained optimization problem to be solved nonlinear optimization solver to more robustly ensure constraint satisfaction.

REFERENCES

- [1] “Rust Programming Language,” Rust Foundation.
- [2] A. C. Aitken, “XXV.—On Bernoulli’s Numerical Solution of Algebraic Equations,” *Proceedings of the Royal Society of Edinburgh*, vol. 46, 1927/ed, pp. 289–305.
- [3] R. Alexander, “Diagonally Implicit Runge–Kutta Methods for Stiff O.D.E.’s,” *SIAM Journal on Numerical Analysis*, vol. 14, no. 6, Dec. 1977, pp. 1006–1021.
- [4] R. W. Anderson, N. S. Elliott, and R. B. Pember, “An Arbitrary Lagrangian–Eulerian Method with Adaptive Mesh Refinement for the Solution of the Euler Equations,” *Journal of Computational Physics*, vol. 199, no. 2, Sept. 2004, pp. 598–617.
- [5] E. Aulisa, S. Bna, and G. Bornia, “A Monolithic ALE Newton–Krylov Solver with Multigrid–Richardson–Schwarz Preconditioning for Incompressible Fluid Structure Interaction,” Sept. 2017.
- [6] T. Bein, P. Friedmann, X. Zhong, and I. Nydick, “Hypersonic Flutter of a Curved Shallow Panel with Aerodynamic Heating,” *34th Structures, Structural Dynamics and Materials Conference*, La Jolla, CA, U.S.A., Apr. 1993, American Institute of Aeronautics and Astronautics.
- [7] M. Bergmann, S. Drapkina, G. Ashcroft, and C. Frey, “A COMPARISON OF VARIOUS NODAL DISCONTINUOUS GALERKIN METHODS FOR THE 3D EULER EQUATIONS,” *Proceedings of the VII European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS Congress 2016)*, Crete Island, Greece, 2016, pp. 7956–7966, Institute of Structural Analysis and Antiseismic Research School of Civil Engineering National Technical University of Athens (NTUA) Greece.
- [8] J. Betteridge, T. H. Gibson, I. G. Graham, and E. H. Müller, “Multigrid Preconditioners for the Hybridised Discontinuous Galerkin Discretisation of the Shallow Water Equations,” *Journal of Computational Physics*, vol. 426, Feb. 2021, p. 109948.
- [9] P. Birken, T. Gleim, D. Kuhl, and A. Meister, “Fast Solvers for Unsteady Thermal Fluid Structure Interaction,” *arXiv:1407.0893 [math]*, July 2014.
- [10] L. S. Blackford, R. Pozo, and E. Al, “An Updated Set of Basic Linear Algebra Subprograms (BLAS),” *NIST*, , no. 2, June 2002, pp. 135–151.

- [11] P. J. Blonigan, K. Carlberg, F. Rizzi, M. Howard, and J. A. Fike, “Model Reduction for Hypersonic Aerodynamics via Conservative LSPG Projection and Hyper-Reduction,” *AIAA Scitech 2020 Forum*, Orlando, FL, Jan. 2020, American Institute of Aeronautics and Astronautics.
- [12] P. N. Brown and Y. Saad, “Hybrid Krylov Methods for Nonlinear Systems of Equations,” *SIAM Journal on Scientific and Statistical Computing*, vol. 11, no. 3, May 1990, pp. 450–481.
- [13] J. M. Calligeros, *Similarity Laws Required for Experimental Aerothermoelastic Studies*, DTIC 0219760, May 1959.
- [14] R. L. Campbell and E. G. Paterson, “Fluid–Structure Interaction Analysis of Flexible Turbomachinery,” *Journal of Fluids and Structures*, vol. 27, no. 8, Nov. 2011, pp. 1376–1391.
- [15] K. Carlberg, M. Barone, and H. Antil, “Galerkin v. Least-Squares Petrov–Galerkin Projection in Nonlinear Model Reduction,” *Journal of Computational Physics*, vol. 330, Feb. 2017, pp. 693–734.
- [16] C. Carton de Wiart, L. T. Diosady, A. Garai, N. K. Burgess, P. J. Blonigan, D. Ekelschot, and S. M. Murman, “Design of a Modular Monolithic Implicit Solver for Multi-Physics Applications,” *2018 AIAA Aerospace Sciences Meeting*, Kissimmee, Florida, Jan. 2018, American Institute of Aeronautics and Astronautics.
- [17] J. Chan, “On Discretely Entropy Conservative and Entropy Stable Discontinuous Galerkin Methods,” *Journal of Computational Physics*, vol. 362, June 2018, pp. 346–374.
- [18] J. Chan, H. Ranocha, A. Rueda-Ramirez, G. Gassner, and T. Warburton, “On the Entropy Projection and the Robustness of High Order Entropy Stable Discontinuous Galerkin Schemes for Under-Resolved Flows,” *arXiv:2203.10238 [cs, math]*, Mar. 2022.
- [19] T. F. Chan and K. R. Jackson, “Nonlinearly Preconditioned Krylov Subspace Methods for Discrete Newton Algorithms,” *SIAM Journal on Scientific and Statistical Computing*, vol. 5, no. 3, Sept. 1984, pp. 533–542.
- [20] T. Chen and C.-W. Shu, “Review of Entropy Stable Discontinuous Galerkin Methods for Systems of Conservation Laws on Unstructured Simplex Meshes,” *CSIAM Transaction on Applied Mathematics*, vol. 1, no. 1, June 2020, pp. 1–52.
- [21] E. J. Ching, Y. Lv, P. Gnoffo, M. Barnhardt, and M. Ihme, “Shock Capturing for Discontinuous Galerkin Methods with Application to Predicting Heat Transfer in Hypersonic Flows,” *Journal of Computational Physics*, vol. 376, Jan. 2019, pp. 54–75.
- [22] C. Ciucă, P. Fernandez, A. Christophe, N. C. Nguyen, and J. Peraire, “Implicit Hybridized Discontinuous Galerkin Methods for Compressible Magnetohydrodynamics,” *Journal of Computational Physics: X*, vol. 5, Jan. 2020, p. 100042.

- [23] B. Cockburn, “Static Condensation, Hybridization, and the Devising of the HDG Methods,” *Building Bridges: Connections and Challenges in Modern Approaches to Numerical Partial Differential Equations*, G. R. Barrenechea, F. Brezzi, A. Cangiani, and E. H. Georgoulis, eds., vol. 114, Springer International Publishing, Cham, 2016, pp. 129–177.
- [24] B. Cockburn and G. Fu, “Devising Superconvergent HDG Methods with Symmetric Approximate Stresses for Linear Elasticity by M-decompositions,” *IMA Journal of Numerical Analysis*, vol. 38, no. 2, Apr. 2018, pp. 566–604.
- [25] B. Cockburn and J. Gopalakrishnan, “The Derivation of Hybridizable Discontinuous Galerkin Methods for Stokes Flow,” *SIAM Journal on Numerical Analysis*, vol. 47, no. 2, Jan. 2009, pp. 1092–1125.
- [26] B. Cockburn, J. Gopalakrishnan, and R. Lazarov, “Unified Hybridization of Discontinuous Galerkin, Mixed, and Continuous Galerkin Methods for Second Order Elliptic Problems,” *SIAM Journal on Numerical Analysis*, vol. 47, no. 2, Jan. 2009, pp. 1319–1365.
- [27] B. Cockburn, N. C. Nguyen, and J. Peraire, “HDG Methods for Hyperbolic Problems,” *Handbook of Numerical Analysis*, vol. 17, Elsevier, 2016, pp. 173–197.
- [28] B. Cockburn and J. Shen, “An Algorithm for Stabilizing Hybridizable Discontinuous Galerkin Methods for Nonlinear Elasticity,” *Results in Applied Mathematics*, vol. 1, June 2019, p. 100001.
- [29] B. Coulter, Z. Wang, D. Huang, and Y. Yao, “Hypersonic Trajectory Optimization with High-Fidelity Aerothermodynamic Models,” *AIAA Scitech 2021 Forum*, VIRTUAL EVENT, Jan. 2021, American Institute of Aeronautics and Astronautics.
- [30] A. R. Crowell and J. J. McNamara, “Model Reduction of Computational Aerothermodynamics for Hypersonic Aerothermoelasticity,” *AIAA Journal*, Aug. 2012.
- [31] A. Culler, A. Crowell, and J. McNamara, “Studies on Fluid-Structural Coupling for Aerothermoelasticity in Hypersonic Flow,” *50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Palm Springs, California, May 2009, American Institute of Aeronautics and Astronautics.
- [32] A. Culler and J. McNamara, “Fluid-Thermal-Structural Modeling and Analysis of Hypersonic Structures under Combined Loading,” *52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, Denver, Colorado, Apr. 2011, American Institute of Aeronautics and Astronautics.
- [33] A. J. Culler, J. J. McNamara, and A. R. Crowell, “Control-Oriented Aerothermoelastic Modeling Approaches for Hypersonic Vehicles,” *2009 American Control Conference*, St. Louis, MO, USA, 2009, pp. 2501–2506, IEEE.
- [34] C. F. Curtiss and J. O. Hirschfelder, “Integration of Stiff Equations*,” *Proceedings of the National Academy of Sciences*, vol. 38, no. 3, Mar. 1952, pp. 235–243.

- [35] J. Degroote, K.-J. Bathe, and J. Vierendeels, “Performance of a New Partitioned Procedure versus a Monolithic Procedure in Fluid–Structure Interaction,” *Computers & Structures*, vol. 87, no. 11, June 2009, pp. 793–801.
- [36] R. Developers, “Rayon,”.
- [37] J. Donea, A. Huerta, J.-P. Ponthot, and A. Rodríguez-Ferran, “Arbitrary Lagrangian–Eulerian Methods,” *Encyclopedia of Computational Mechanics*, John Wiley & Sons, Ltd, 2004, chapter 14.
- [38] H. L. Dryden and J. E. Duberg, *Aeroelastic Effects of Aerodynamic Heating*, Tech. Rep. AGARD20-P10, June 1995.
- [39] L. Failer and T. Wick, “Adaptive Time-Step Control for Nonlinear Fluid–Structure Interaction,” *Journal of Computational Physics*, vol. 366, Aug. 2018, pp. 448–477.
- [40] N. J. Falkiewicz and C. E. S. Cesnik, “Proper Orthogonal Decomposition for Reduced-Order Thermal Solution in Hypersonic Aerothermoelastic Simulations,” p. 35.
- [41] N. J. Falkiewicz, C. E. S. Cesnik, A. R. Crowell, and J. J. McNamara, “Reduced-Order Aerothermoelastic Framework for Hypersonic Vehicle Control Simulation,” p. 45.
- [42] C. Farhat, C. Degand, B. Koobus, and M. Lesoinne, “Torsional Springs for Two-Dimensional Dynamic Unstructured Fluid Meshes,” 1998.
- [43] C. Farhat, M. Lesoinne, and P. Le Tallec, “Load and Motion Transfer Algorithms for Fluid/Structure Interaction Problems with Non-Matching Discrete Interfaces: Momentum and Energy Conservation, Optimal Discretization and Application to Aeroelasticity,” *Computer Methods in Applied Mechanics and Engineering*, vol. 157, no. 1, Apr. 1998, pp. 95–114.
- [44] C. Farhat, K. G. van der Zee, and P. Geuzaine, “Provably Second-Order Time-Accurate Loosely-Coupled Solution Algorithms for Transient Nonlinear Computational Aeroelasticity,” *Computer Methods in Applied Mechanics and Engineering*, vol. 195, no. 17, Mar. 2006, pp. 1973–2001.
- [45] C. A. Felippa, K. C. Park, and C. Farhat, “Partitioned Analysis of Coupled Mechanical Systems,” *Computer Methods in Applied Mechanics and Engineering*, vol. 190, no. 24, Mar. 2001, pp. 3247–3270.
- [46] P. Fernandez, *Entropy-Stable Hybridized Discontinuous Galerkin Methods for Large-Eddy Simulation of Transitional and Turbulent Flows*, doctoral dissertation, Oct. 2018.
- [47] P. Fernandez, A. Christophe, S. Terrana, N. C. Nguyen, and J. Peraire, “Hybridized Discontinuous Galerkin Methods for Wave Propagation,” *Journal of Scientific Computing*, vol. 77, no. 3, Dec. 2018, pp. 1566–1604.

- [48] P. Fernandez, R. Moura, G. Mengaldo, and J. Peraire, “Non-Modal Analysis of Spectral Element Methods: Towards Accurate and Robust Large-Eddy Simulations,” *Computer Methods in Applied Mechanics and Engineering*, vol. 346, Apr. 2019, pp. 43–62.
- [49] P. Fernandez, N. C. Nguyen, and J. Peraire, “The Hybridized Discontinuous Galerkin Method for Implicit Large-Eddy Simulation of Transitional Turbulent Flows,” *Journal of Computational Physics*, vol. 336, May 2017, pp. 308–329.
- [50] P. Fernandez, N.-C. Nguyen, and J. Peraire, “Subgrid-Scale Modeling and Implicit Numerical Dissipation in DG-based Large-Eddy Simulation,” *23rd AIAA Computational Fluid Dynamics Conference*, 2017, p. 3951.
- [51] P. Fernandez, N.-C. Nguyen, and J. Peraire, “Entropy-Stable Hybridized Discontinuous Galerkin Methods for the Compressible Euler and Navier-Stokes Equations,” Aug. 2018.
- [52] P. Fernandez and Q. Wang, “Lyapunov Spectrum of Scale-Resolving Turbulent Simulations. Application to Chaotic Adjoints,” *Fernandez*, June 2017.
- [53] P. Fernandez del Campo, N. C. Nguyen, and J. Peraire, “A Physics-Based Shock Capturing Method for Unsteady Laminar and Turbulent Flows,” *Pablo Fernandez*, Jan. 2018.
- [54] C. Förster, W. A. Wall, and E. Ramm, “Artificial Added Mass Instabilities in Sequential Staggered Coupling of Nonlinear Structures and Incompressible Viscous Flows,” *Computer Methods in Applied Mechanics and Engineering*, vol. 196, no. 7, Jan. 2007, pp. 1278–1293.
- [55] B. Froehle and P.-O. Persson, “A High-Order Discontinuous Galerkin Method for Fluid–Structure Interaction with Efficient Implicit–Explicit Time Stepping,” *Journal of Computational Physics*, vol. 272, Sept. 2014, pp. 455–470.
- [56] C. W. Gear, “The Numerical Integration of Ordinary Differential Equations,” *Mathematics of Computation*, vol. 21, no. 98, 1967, pp. 146–156.
- [57] M. W. Gee, U. Küttler, and W. A. Wall, “Truly Monolithic Algebraic Multigrid for Fluid–Structure Interaction,” *International Journal for Numerical Methods in Engineering*, vol. 85, no. 8, 2011, pp. 987–1016.
- [58] G. Giorgiani, S. Fernández-Méndez, and A. Huerta, “Hybridizable Discontinuous Galerkin p -Adaptivity for Wave Propagation Problems: HDG P -ADAPTIVITY FOR WAVE PROPAGATION PROBLEMS,” *International Journal for Numerical Methods in Fluids*, vol. 72, no. 12, Aug. 2013, pp. 1244–1262.
- [59] G. Giorgiani, S. Fernández-Méndez, and A. Huerta, “Hybridizable Discontinuous Galerkin with Degree Adaptivity for the Incompressible Navier–Stokes Equations,” *Computers & Fluids*, vol. 98, July 2014, pp. 196–208.

- [60] V. Gravemeier, S. M. Civaner, and W. A. Wall, “A Partitioned-Monolithic Finite Element Method for Thermo-Fluid–Structure Interaction,” *Computer Methods in Applied Mechanics and Engineering*, vol. 401, Nov. 2022, p. 115596.
- [61] M. Haupt, R. Niesner, R. Unger, and P. Horst, “Coupling Techniques for Thermal and Mechanical Fluid-Structure-Interactions in Aeronautics,” *PAMM*, vol. 5, no. 1, 2005, pp. 19–22.
- [62] T. Houba, A. Dasgupta, S. Gopalakrishnan, R. Gosse, and S. Roy, “Supersonic Turbulent Flow Simulation Using a Scalable Parallel Modal Discontinuous Galerkin Numerical Method,” *Scientific Reports*, vol. 9, no. 1, Dec. 2019, p. 14442.
- [63] J. Hron and S. Turek, “A Monolithic FEM/Multigrid Solver for ALE Formulation of Fluid Structure Interaction with Application in Biomechanics,” 2006, p. 26.
- [64] D. Huang, *Development of a Hypersonic Aerothermoelastic Framework and Its Application to Flutter and Aerothermoelastic Scaling of Skin Panels*, doctoral dissertation, 2019.
- [65] D. Huang and P. P. Friedmann, “A MULTI-OBJECTIVE OPTIMIZATION FRAMEWORK FOR HYPERSONIC AEROTHERMOELASTIC SCALING LAWS AND ITS APPLICATION TO SKIN PANELS,” p. 44.
- [66] D. Huang, T. Rokita, and P. P. Friedmann, “An Aerothermoelastic Analysis Framework Enhanced by Model Order Reduction With Applications,” *58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Grapevine, Texas, Jan. 2017, American Institute of Aeronautics and Astronautics.
- [67] S. Kang, F. X. Giraldo, and T. Bui-Thanh, “IMEX HDG-DG: A Coupled Implicit Hybridized Discontinuous Galerkin and Explicit Discontinuous Galerkin Approach for Shallow Water Systems,” *Journal of Computational Physics*, vol. 401, Jan. 2020, p. 109010.
- [68] J. A. Kauffman, J. P. Sheldon, and S. T. Miller, “Overset Meshing Coupled with Hybridizable Discontinuous Galerkin Finite Elements: OVERSET MESHING COUPLED WITH HDG FINITE ELEMENTS,” *International Journal for Numerical Methods in Engineering*, vol. 112, no. 5, Nov. 2017, pp. 403–433.
- [69] C. T. Kelley and D. E. Keyes, “Convergence Analysis of Pseudo-Transient Continuation,” *SIAM Journal on Numerical Analysis*, vol. 35, no. 2, Apr. 1998, pp. 508–523.

- [70] D. E. Keyes, L. C. McInnes, C. Woodward, W. Gropp, E. Myra, M. Pernice, J. Bell, J. Brown, A. Clo, J. Connors, E. Constantinescu, D. Estep, K. Evans, C. Farhat, A. Hakim, G. Hammond, G. Hansen, J. Hill, T. Isaac, X. Jiao, K. Jordan, D. Kaushik, E. Kaxiras, A. Koniges, K. Lee, A. Lott, Q. Lu, J. Magerlein, R. Maxwell, M. McCourt, M. Mehl, R. Pawlowski, A. P. Randles, D. Reynolds, B. Rivière, U. Rüde, T. Scheibe, J. Shadid, B. Sheehan, M. Shephard, A. Siegel, B. Smith, X. Tang, C. Wilson, and B. Wohlmuth, “Multiphysics Simulations: Challenges and Opportunities,” *The International Journal of High Performance Computing Applications*, vol. 27, no. 1, Feb. 2013, pp. 4–83.
- [71] R. Klock and C. E. Cesnik, “Aerothermoelastic Simulation of Air-Breathing Hypersonic Vehicles,” *55th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, National Harbor, Maryland, Jan. 2014, American Institute of Aeronautics and Astronautics.
- [72] D. A. Knoll and D. E. Keyes, “Jacobian-Free Newton–Krylov Methods: A Survey of Approaches and Applications,” *Journal of Computational Physics*, vol. 193, no. 2, Jan. 2004, pp. 357–397.
- [73] U. Küttler and W. A. Wall, “Fixed-Point Fluid–Structure Interaction Solvers with Dynamic Relaxation,” *Computational Mechanics*, vol. 43, no. 1, Dec. 2008, pp. 61–72.
- [74] A. La Spina and J. Fish, “A Superconvergent Hybridizable Discontinuous Galerkin Method for Weakly Compressible Magnetohydrodynamics,” *Computer Methods in Applied Mechanics and Engineering*, vol. 388, Jan. 2022, p. 114278.
- [75] N. Lamorte and P. P. Friedmann, “Hypersonic Aeroelastic and Aerothermoelastic Studies Using Computational Fluid Dynamics,” *AIAA Journal*, vol. 52, no. 9, Sept. 2014, pp. 2062–2078.
- [76] N. Lamorte, P. P. Friedmann, B. Glaz, A. J. Culler, A. R. Crowell, and J. J. McNamara, “Uncertainty Propagation in Hypersonic Aerothermoelastic Analysis,” *Journal of Aircraft*, vol. 51, no. 1, 2014, pp. 192–203.
- [77] M. Lesoinne and C. Farhat, “Geometric Conservation Laws for Aeroelastic Computations Using Unstructured Dynamic Meshes,” *12th Computational Fluid Dynamics Conference*, American Institute of Aeronautics and Astronautics, 1995.
- [78] L. Li, S. Lanteri, and R. Perrussel, “A Hybridizable Discontinuous Galerkin Method Combined to a Schwarz Algorithm for the Solution of 3d Time-Harmonic Maxwell’s Equation,” *Journal of Computational Physics*, vol. 256, Jan. 2014, pp. 563–581.
- [79] R. Lind, “Linear Parameter-Varying Modeling and Control of Structural Dynamics with Aerothermoelastic Effects,” *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 4, 2002, pp. 733–739.

- [80] Y. Lv and M. Ihme, “Entropy-Bounded Discontinuous Galerkin Scheme for Euler Equations,” *Journal of Computational Physics*, vol. 295, Aug. 2015, pp. 715–739.
- [81] G. May, K. Devesse, A. Rangarajan, and T. Magin, “A Hybridized Discontinuous Galerkin Solver for High-Speed Compressible Flow,” *Aerospace*, vol. 8, no. 11, Nov. 2021, p. 322.
- [82] M. Mayr, M. H. Noll, and M. W. Gee, “A Hybrid Interface Preconditioner for Monolithic Fluid–Structure Interaction Solvers,” *Advanced Modeling and Simulation in Engineering Sciences*, vol. 7, no. 1, Apr. 2020, p. 15.
- [83] M. Mayr, W. Wall, and M. Gee, “Adaptive Time Stepping for Fluid-Structure Interaction Solvers,” *Finite Elements in Analysis and Design*, vol. 141, Mar. 2018, pp. 55–69.
- [84] J. McNamara, P. Friedmann, K. Powell, B. Thuruthimattam, and R. Bartels, “Three-Dimensional Aeroelastic and Aerothermoelastic Behavior in Hypersonic Flow,” *46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, American Institute of Aeronautics and Astronautics.
- [85] J. J. McNamara and P. P. Friedmann, “Aeroelastic and Aerothermoelastic Analysis in Hypersonic Flow: Past, Present, and Future,” *AIAA Journal*, vol. 49, no. 6, June 2011, pp. 1089–1122.
- [86] M. Mehl, B. Uekermann, H. Bijl, D. Blom, B. Gatzhammer, and A. van Zuijlen, “Parallel Coupling Numerics for Partitioned Fluid–Structure Interaction Simulations,” *Computers & Mathematics with Applications*, vol. 71, no. 4, Feb. 2016, pp. 869–891.
- [87] J. A. Meijerink and H. A. van der Vorst, “An Iterative Solution Method for Linear Systems of Which the Coefficient Matrix Is a Symmetric SM -Matrix,” *Mathematics of Computation*, vol. 31, no. 137, 1977, pp. 148–162.
- [88] J. G. Michopoulos, C. Farhat, and J. Fish, “Modeling and Simulation of Multiphysics Systems,” *Journal of Computing and Information Science in Engineering*, vol. 5, no. 3, Sept. 2005, pp. 198–213.
- [89] A. Monge, “Partitioned Methods for Time-Dependent Thermal Fluid-Structure Interaction,” *Numerical Analysis*, p. 197.
- [90] D. Moro-Ludafia, “A Hybridized Discontinuous Petrov-Galerkin Scheme for Compressible Flows,” p. 117.
- [91] R. C. Moura, P. Fernandez, G. Mengaldo, and S. J. Sherwin, “Viscous Diffusion Effects in the Eigenanalysis of (Hybridisable) DG Methods,” *Spectral and High Order Methods for Partial Differential Equations ICOSAHOM 2018*, Springer, Cham, 2020, pp. 371–382.
- [92] S. Muralikrishnan, *Fast and Scalable Solvers for High-Order Hybridized Discontinuous Galerkin Methods with Applications to Fluid Dynamics and Magnetohydrodynamics*, Thesis, Aug. 2019.

- [93] N. C. Nguyen and J. Peraire, “Hybridizable Discontinuous Galerkin Methods for Partial Differential Equations in Continuum Mechanics,” *Journal of Computational Physics*, vol. 231, no. 18, July 2012, pp. 5955–5988.
- [94] N. C. Nguyen, J. Peraire, and B. Cockburn, “An Implicit High-Order Hybridizable Discontinuous Galerkin Method for Linear Convection–Diffusion Equations,” *Journal of Computational Physics*, vol. 228, no. 9, May 2009, pp. 3232–3254.
- [95] N. C. Nguyen, J. Peraire, and B. Cockburn, “An Implicit High-Order Hybridizable Discontinuous Galerkin Method for Nonlinear Convection–Diffusion Equations,” *Journal of Computational Physics*, vol. 228, no. 23, Dec. 2009, pp. 8841–8855.
- [96] N. C. Nguyen, J. Peraire, and B. Cockburn, “High-Order Implicit Hybridizable Discontinuous Galerkin Methods for Acoustics and Elastodynamics,” *Journal of Computational Physics*, vol. 230, no. 10, May 2011, pp. 3695–3718.
- [97] N. C. Nguyen, J. Peraire, and B. Cockburn, “An Implicit High-Order Hybridizable Discontinuous Galerkin Method for the Incompressible Navier–Stokes Equations,” *Journal of Computational Physics*, vol. 230, no. 4, Feb. 2011, pp. 1147–1170.
- [98] N.-C. Nguyen, J. Peraire, M. Solano, and S. Terrana, “An HDG Method for Non-Matching Meshes,” *Preprint*, vol. 9, 2020.
- [99] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, “Automatic Differentiation in PyTorch,” Oct. 2017.
- [100] P. O. Persson, J. Bonet, and J. Peraire, “Discontinuous Galerkin Solution of the Navier–Stokes Equations on Deformable Domains,” *Computer Methods in Applied Mechanics and Engineering*, vol. 198, no. 17, Apr. 2009, pp. 1585–1595.
- [101] P.-O. Persson and J. Peraire, “Sub-Cell Shock Capturing for Discontinuous Galerkin Methods,” *44th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, Jan. 2006, American Institute of Aeronautics and Astronautics.
- [102] W. H. Reed and T. R. Hill, *Triangular Mesh Methods for the Neutron Transport Equation*, Tech. Rep., Los Alamos Scientific Lab., N. Mex.(USA), 1973.
- [103] Y. Saad and M. H. Schultz, “GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems,” *SIAM Journal on Scientific and Statistical Computing*, vol. 7, no. 3, July 1986, pp. 856–869.
- [104] A. Sadagopan, D. Huang, H. Xu, and X. I. Yang, “Numerical Investigation of Fluid-Thermal-Structural Interaction for a Control Surface in Hypersonic Flow,” *AIAA Scitech 2021 Forum*, VIRTUAL EVENT, Jan. 2021, American Institute of Aeronautics and Astronautics.

- [105] A. Samii, C. Michoski, and C. Dawson, “A Parallel and Adaptive Hybridized Discontinuous Galerkin Method for Anisotropic Nonhomogeneous Diffusion,” *Computer Methods in Applied Mechanics and Engineering*, vol. 304, June 2016, pp. 118–139.
- [106] J. Schöberl, C. Lehrenfeld, J. Schöberl, and C. Lehrenfeld, “C.: Domain Decomposition Preconditioning for High Order Hybrid Discontinuous Galerkin Methods on Tetrahedral Meshes. In: Advanced Finite Element Methods and Applications, Lect,” *Notes Appl. Comput. Mech*, 2013, pp. 27–56.
- [107] K. Sengupta, F. Mashayek, and G. Jacobs, “Large-Eddy Simulation Using a Discontinuous Galerkin Spectral Element Method,” *45th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, Jan. 2007, American Institute of Aeronautics and Astronautics.
- [108] J. P. Sheldon, S. T. Miller, and J. S. Pitt, “Methodology for Comparing Coupling Algorithms for Fluid-Structure Interaction Problems,” *World Journal of Mechanics*, vol. 04, no. 02, Feb. 2014, p. 54.
- [109] J. P. Sheldon, S. T. Miller, and J. S. Pitt, “A Hybridizable Discontinuous Galerkin Method for Modeling Fluid–Structure Interaction,” *Journal of Computational Physics*, vol. 326, Dec. 2016, pp. 91–114.
- [110] J. P. Sheldon, S. T. Miller, and J. S. Pitt, “A Hybridizable Discontinuous Galerkin Method for Modeling Fluid–Structure Interaction,” *Journal of Computational Physics*, vol. 326, Dec. 2016, pp. 91–114.
- [111] J. P. Sheldon, S. T. Miller, and J. S. Pitt, “An Improved Formulation for Hybridizable Discontinuous Galerkin Fluid-Structure Interaction Modeling with Reduced Computational Expense,” *Communications in Computational Physics*, vol. 24, no. 5, 2018.
- [112] S. J. Sherwin, R. M. Kirby, J. Peiró, R. L. Taylor, and O. C. Zienkiewicz, “On 2D Elliptic Discontinuous Galerkin Methods,” *International Journal for Numerical Methods in Engineering*, vol. 65, no. 5, Jan. 2006, pp. 752–784.
- [113] L. J. Smith, L. J. Halim, G. Kennedy, and M. J. Smith, “A High-Fidelity Coupling Framework for Aerothermoelastic Analysis and Adjoint-Based Gradient Evaluation,” *AIAA Scitech 2021 Forum*, VIRTUAL EVENT, Jan. 2021, American Institute of Aeronautics and Astronautics.
- [114] M. Solano, S. Terrana, N.-C. Nguyen, and J. Peraire, “An HDG Method for Dissimilar Meshes,” *IMA Journal of Numerical Analysis*, vol. 42, no. 2, Apr. 2022, pp. 1665–1699.
- [115] S. Terrana, N. Nguyen, J. Bonet, and J. Peraire, “A Hybridizable Discontinuous Galerkin Method for Both Thin and 3D Nonlinear Elastic Structures,” *Computer Methods in Applied Mechanics and Engineering*, vol. 352, Aug. 2019, pp. 561–585.
- [116] E. A. Thornton and P. Dechaumphai, “Coupled Flow, Thermal, and Structural Analysis of Aerodynamically Heated Panels,” *Journal of Aircraft*, vol. 25, no. 11, Nov. 1988, pp. 1052–1059.

- [117] A. Toselli and O. B. Widlund, *Domain Decomposition Methods — Algorithms and Theory*, vol. 34 of *Springer Series in Computational Mathematics*, Springer, Berlin, Heidelberg, 2005.
- [118] H. Tran and C. Farhat, “An Integrated Platform for the Simulation of Fluid-Structure-Thermal Interaction Problems,” *43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Denver, Colorado, Apr. 2002, American Institute of Aeronautics and Astronautics.
- [119] S. Turek and J. Hron, “Proposal for Numerical Benchmarking of Fluid-Structure Interaction between an Elastic Object and Laminar Incompressible Flow,” *Fluid-Structure Interaction*, H.-J. Bungartz and M. Schäfer, eds., Berlin, Heidelberg, 2006, Lecture Notes in Computational Science and Engineering, pp. 371–385, Springer.
- [120] E. H. van Brummelen, “Added Mass Effects of Compressible and Incompressible Flows in Fluid-Structure Interaction,” *Journal of Applied Mechanics*, vol. 76, no. 2, Mar. 2009, p. 021206.
- [121] E. H. van Brummelen, “Partitioned Iterative Solution Methods for Fluid–Structure Interaction,” *International Journal for Numerical Methods in Fluids*, vol. 65, no. 1-3, 2011, pp. 3–27.
- [122] C. A. Vargas Venegas and D. Huang, “Expedient Hypersonic Aerothermal Prediction for Aerothermoelastic Analysis Via Field Inversion and Machine Learning,” *AIAA Scitech 2021 Forum*, VIRTUAL EVENT, Jan. 2021, American Institute of Aeronautics and Astronautics.
- [123] Z. D. Wilcox, W. MacKunis, S. Bhat, R. Lind, and W. E. Dixon, “Lyapunov-Based Exponential Tracking Control of a Hypersonic Aircraft with Aerothermoelastic Effects,” *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 4, July 2010, pp. 1213–1224.
- [124] M. Woopen, A. Balan, and G. May, “A Hybridized Discontinuous Galerkin Method for Three-Dimensional Compressible Flow Problems,” *52nd Aerospace Sciences Meeting*, National Harbor, Maryland, Jan. 2014, American Institute of Aeronautics and Astronautics.
- [125] M. Woopen, G. May, and J. Schütz, “Adjoint-Based Error Estimation and Mesh Adaptation for Hybridized Discontinuous Galerkin Methods,” *International Journal for Numerical Methods in Fluids*, vol. 76, no. 11, Dec. 2014, pp. 811–834.
- [126] K. Ye, Z. Ye, Z. Feng, Y. Pan, and G. Wang, “Numerical Investigation on the Aerothermoelastic Deformation of the Hypersonic Wing,” *Acta Astronautica*, vol. 160, July 2019, pp. 76–89.
- [127] Y. Zhu, W. Peng, R. Xu, and P. Jiang, “Review on Active Thermal Protection and Its Heat Transfer for Airbreathing Hypersonic Vehicles,” *Chinese Journal of Aeronautics*, vol. 31, no. 10, Oct. 2018, pp. 1929–1953.