Modeling, Discontinuous Galerkin Approximation and Simulation of the 1-D Compressible Navier Stokes Equations

Zachary G. Grigorian

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Masters in Mathematics
in
Mathematics

John Burns, Chair
Jeff Borggaard
Eugene Cliff
Shu-Ming Sun

May 10, 2019
Blacksburg, Virginia

Keywords: Navier Stokes, Derivation, DG Method, Simulation, Pseudo-Transient
Copyright 2019, Zachary G. Grigorian
Modeling, Discontinuous Galerkin Approximation and Simulation of the 1-D Compressible Navier Stokes Equations

Zachary G. Grigorian

(ABSTRACT)

In this thesis we derive time dependent equations that govern the physics of a thermal fluid flowing through a one dimensional pipe. We begin with the conservation laws described by the 3D compressible Navier Stokes equations. This model includes all residual terms resulting from the 1D flow approximations. The final model assumes that all the residual terms are negligible which is a standard assumption in industry. Steady state equations are obtained by assuming the temporal derivatives are zero. We develop a semi-discrete model by applying a linear discontinuous Galerkin method in the spatial dimension. The resulting finite dimensional model is a differential algebraic equation (DAE) which is solved using standard integrators. We investigate two methods for solving the corresponding steady state equations. The first method requires making an initial guess and employs a Newton based solver. The second method is based on a pseudo-transient continuation method. In this method one initializes the dynamic model and integrates forward for a fixed time period to obtain a profile that initializes a Newton solver. We observe that non-uniform meshing can significantly reduce model size while retaining accuracy. For comparison, we employ the same initialization for the pseudo-transient algorithm and the Newton solver. We demonstrate that for the systems considered here, the pseudo-transient initialization algorithm produces initializations that reduce iteration counts and function evaluations when compared to the Newton solver. Several numerical experiments were conducted to illustrate the ideas. Finally, we close with suggestions for future research.
In this thesis we derive time dependent equations that govern the physics of a fluid flowing through a one dimensional pipe. This model includes all error terms that result from 1D modeling approximations. The final model assumes that all of these error terms are negligible which is a standard assumption in industry. Steady state equations result when all time dependence is removed from the 1D equations. We approximate the true solution by a discontinuous piece-wise linear function. Standard techniques are used to solve for this approximate solution. We investigate two methods for solving the steady state equations. In the first method, one makes an educated guess about the solution profile and uses Newton’s method to solve for the true solution. The second method, pseudo-transient initialization, attempts to improve this initial guess through dynamic simulation. In this method, an initial guess is treated as the initial conditions for dynamic simulation. The dynamic simulation is then run for a fixed amount of time. The solution at the end of the simulation is the improved initial guess for Newton’s method and is used to solve for the steady state profile. To test the pseudo-transient initialization, we determine the number of function evaluations required to obtain the steady state solution for an initial guess with and without performing pseudo-transient initialization on it. We demonstrate that for the systems considered here, the pseudo-transient initialization algorithm reduced overall computational costs. Also, we observe that non-uniform meshing can significantly reduce model size while retaining accuracy. Several numerical experiments were conducted to illustrate these ideas. Finally, we close with suggestions for future research.
Dedication

This thesis is dedicated to my parents.
Acknowledgments

I am grateful to the members of my committee for your help in writing and critiquing the many drafts of this thesis. I am especially grateful to Dr. Burns for his guidance and advising me for these two years. I would like to thank CCS and the Heat Transfer Group at Carrier for funding my research and opening many doors for my future career. Additionally, I would like to thank Dr. Borggaard and Dr. Zietsman specifically for their help and encouragement in starting my masters program. Without them, I would not have even started on this journey.
Contents

List of Figures ix

List of Tables xi

Nomenclature xii

1 Introduction 1

2 Background 9

2.1 The Finite Volume Method 9

2.2 The Discontinuous Galerkin Method 12

2.3 The Linear Basis Functions 17

2.4 Psuedo-Transient Initialization 19

3 Derivation of the 1D Navier Stokes Equations 25

3.1 Continuity Equation 25

3.2 Momentum Equation 27

3.3 The Energy Equation 31

3.4 The Enthalpy Equation 34

3.5 Summary of Equations and Model Closure 37
8 Summary and Conclusions

8.1 Final Differential Algebraic Equations .................................. 105
8.2 Dynamic System Semi-Discretization ................................. 105
8.3 Steady State Discretization ................................................ 109
8.4 Numerical Results .......................................................... 110
8.5 Future Work ............................................................... 111

Bibliography .......................... 114
## List of Figures

2.1 Example of a finite volume method solution profile ............................................. 10

2.2 Example of a linear discontinuous Galerkin method solution profile .................. 12

2.3 Example of a piece-wise linear basis functions ..................................................... 18

2.4 Illustration of notation used in describing the DG linear basis functions and a local approximate solution for arbitrary quantity \( \psi(t, x) \) .................................................. 19

2.5 An example illustrating the A-stability of backwards Euler and the A-instability of Adams Bashforth. ................................................................. 21

2.6 Illustration of a basin of attraction for compressible Navier Stokes equations. ....... 22

2.7 Illustration behind idea of pseudo-transient initialization ........................................ 23

4.1 Illustration of example mesh and associated notation. ........................................... 42

4.2 Illustration of how a global approximate solution is constructed from a linear basis functions. ......................................................................................... 43

6.1 The steady state enthalpy profile using a uniform mesh. ...................................... 90

6.2 The steady state velocity profile using a uniform mesh. ...................................... 91

6.3 Plot of approximate steady state enthalpy profile using 10 non-uniformly clustered elements and 64 uniform elements. ................................. 92

6.4 Plot of approximate steady state velocity profile using 10 non-uniformly clustered elements and 64 uniform elements. ........................................ 93
6.5 Steady state enthalpy profile at various mass flow rates. .......................... 94

6.6 Nearly adiabatic (U=0.5) steady state enthalpy profile at varying mass flow
rates. ............................................................................................................. 95

7.1 The dynamic enthalpy profile over an integration period of one second. Note:
the pipe coordinate axis is flipped to see surface plot more clearly. ............... 98

7.2 The dynamic velocity profile over an integration period of one second. Note:
the pipe coordinate axis is flipped to see surface plot more clearly. ............... 99

7.3 The dynamic density profile over an integration period of one second. Note:
the pipe coordinate axis is flipped to see surface plot more clearly. ............... 99

7.4 Final enthalpy profile of the dynamic solver compared to steady state solution. 100

7.5 Final velocity profile of the dynamic solver compared to steady state solution. 100
List of Tables

4.1 Table labeling each term in the enthalpy equation. ........................................ 49

4.2 Table labeling each term in the enthalpy equation. ........................................ 72

6.1 Table of test parameters used for determining effect of non-uniform meshing
on solution accuracy .................................................................................................. 89

7.1 Table of test parameters used for testing dynamic compressible flow solver .... 97

7.2 Table of number of right hand side function evaluations needed to solve for
steady state using the pseudo-transient initialization scheme employing dif-
ferent Newton algorithms. ......................................................................................... 102

7.3 Table of right hand side function evaluations to attain steady state varying
the enthalpy boundary condition .............................................................................. 103

8.1 Table labeling each term in the enthalpy equation. ........................................ 106
**Nomenclature**

\( \sigma \)  Stress Tensor on a fluid

\( \rho \)  Density of a substance

\( A \)  Cross-sectional area of a pipe

\( P \)  Perimeter of a pipe

\( p \)  Pressure at a specific point in a pipe

\( \mathbf{u} \)  Velocity vector a fluid

\( u \)  \( x \) component of the velocity vector \( \mathbf{u} \)

\( v \)  \( y \) component of the velocity vector \( \mathbf{u} \)

\( w \)  \( z \) component of the velocity vector \( \mathbf{u} \)

\( \tau \)  Shear Stress Tensor on a fluid

\( e \)  Specific internal energy of a fluid

\( q \)  Thermal Conductivity of a fluid

\( f \)  Body forces on a fluid

\( U \)  Heat transfer coefficient of the pipe

\( \tilde{U} \)  Overall heat transfer coefficient of the pipe

\( \kappa \)  Thermal conductivity coefficient of a fluid
\( \mu \) Dynamic viscosity of a fluid

\( F_F \) Frictional force acting on a fluid

\( h \) Specific enthalpy of a fluid

\( \Phi \) Viscous Dissipation forces on a fluid

\( \dot{Q} \) Heat Generation/External Heat Transfer

\( \Phi \) Viscous Dissipation

\( T \) Temperature of the fluid

\( E \) Overall internal energy of the fluid

\( f \) Body forces acting on the fluid

\( \tau \) shear stress tensor

\( \tau^r \) Velocity squared residual term

\( \Pi_{diff} \) Shear stress residual term

\( \Phi_{res} \) Viscous Dissipation residual term

\( X \) Gravitational potential energy

\( \phi \) Linear basis function (unless stated otherwise)
Chapter 1

Introduction

Building physical prototypes of engineered products can be expensive and time consuming. For example, a prototype car built for crash testing can cost between $250,000 and $1,000,000 [19]. Furthermore, societal and governmental pressures are constantly forcing companies to increase the energy efficiency of their products. The European Union has issued a directive to reduce overall annual energy consumption by 30% by 2030 from 2014 levels [10]. These types of regulations are forcing companies to redesign many products from the ground up more rapidly than ever before. However, building physical prototypes for all of these new products is economically infeasible. Instead, companies such as the Climate, Controls and Security (CCS) Division of the United Technologies Corporation and various automotive companies are using mathematical models to improve efficiency and to develop new products.

Constructing appropriate mathematical models, discretization of these models and developing the corresponding software to implement the discretized models can be significantly faster and more cost effective than using physical prototypes to design and build new products. Mathematical models allow engineers to evaluate hundreds of potential designs before deciding on a final build for physical testing. As modeling and simulation become more mature, computational models can produce sufficiently accurate numerical data to replace test data collected during physical testing sessions [14]. These benefits are pushing industry to move from expensive physical tests to computational modeling.

For several years now, many industries have been moving towards the use of mathematical
models and computational science to understand, design, control and optimize complex engineering systems. This process has multiple labels (Model Based Design, Model Based Development, Model Based Engineering), but the basic idea is to use mathematical modeling and numerical computation to solve specific engineering problems. These problems include analysis, control, optimization and design. The methods and computational tools used to solve these problems must be applicable to a broad range of technologies. In this thesis, we use the term Model Based Development (MBD) to describe the process of combining mathematical modeling with computational science to address such problems. In addition, we focus on physics based modeling where we emphasize that knowing how the model is to be used (simulation, optimization, control, etc.) places requirements on the model properties. We call this modeling for “X” where “X” denotes the problem to be solved (i.e., simulation, control, optimization, or design).

First and foremost, a physics based model should provide the ability to simulate the important physics present in the system of interest. However, if the goal of modeling is more than simulation alone, then the model needs to have additional properties that make it useful for the intended goals. The way one realizes a physical system as a mathematical model greatly impacts the model’s smoothness and other model properties. This can be a limiting factor on how the model can be used in a typical engineering workflow. This is especially true when the model is to be used in optimization based design. For example, as noted by Polak in [21], in addition to smoothness, many optimization algorithms require consistent approximations of the dual or adjoint systems to ensure convergence of the optimizer.

Regardless of the problem to be solved, MBD has the following minimum requirements:

1. **Mathematical Model(s):** This is a set of mathematical equations that describe the science of the engineering system at a level sufficient for use on a particular problem. Since the MBD engineering workflow involves solving multiple problems, one
often needs to develop models of varying fidelity and with different properties. Often, a model suitable for detailed simulation may not be useful in controller design and optimization.

2. **Simplifications and Discretization Methods:** Approximations are required to reduce the fundamental equations (e.g. conservation laws) to finite dimensional systems of equations suitable for implementation on a digital computer. Finite difference, finite volume and finite elements (or combinations) are typical methods used in thermal fluid systems.

3. **Numerical Algorithms:** Numerical algorithms enable the computation of the desired solutions and vary from problem to problem. However, there is often a cross fertilization in the sense that a numerical method designed for one task can prove useful for another. Indeed, some of the best steady state solvers are optimization based and can be improved by taking advantage of suitable dynamic models.

4. **Software and Software Tools:** By software we mean the implementation of numerical algorithms into a particular scientific language and other high level tools that enables the engineering workflow such as system composition and interfacing with external solvers.

Without suitable mathematical models and appropriate numerical algorithms, MBD cannot be realized. The focus here is on physics based modeling which begins with continuum modeling of pipe flow. We use this problem to illustrate the process of model development, problem formulation and numerical solution. In this thesis we consider a thermal fluid in a pipe flowing in one direction and although the focus here is on steady state solutions, we derive the full dynamic model as a starting point. Specifically:

1. We derive a dynamically consistent full-flux model by starting with the compressible
Navier-Stokes equations and reduce the equations to a set of partial differential equations (PDEs) in one spatial dimension. This 1D system includes thermal diffusion, shear stress and viscous dissipation. Starting with the fundamental conservation laws for model development allows one to develop hierarchical models for different applications. In addition, retaining the dynamics, thermal diffusion, shear stress and viscous dissipation in the model generates a numerical model that is more suitable for a variety of operating conditions (e.g., low flow).

2. We apply a Discontinuous Galerkin (DG) spatial discretization to the 1D PDE system and generate a set of finite dimensional Differential Algebraic Equations (DAEs). This system of DAEs provides the basic numerical model used for simulation. It is important to note that the conservation laws have a mathematical structure that can be leveraged to produce discretizations that retain this structure and hence capture the important physics at every level of model fidelity. Modern Discontinuous Galerkin (DG) methods (see [11, 12]), high order Finite Element (FE) methods and combined finite element - finite volume (FE-FV) (see [7] and [8]) now exist that are valid for an entire range of operating conditions without having to resort to special numerical stabilization techniques. In addition, these methods are simple to implement for 1D problems.

3. We employ numerical methods based on Newton type methods to solve the steady state system. It is important to note that most of the modern solvers are optimization based meaning a certain amount of model smoothness may be required. In general, optimization tools developed in the past ten years (now readily available open source software [1]) can deal with millions of design variables if the models are developed with optimization in mind.

4. We make use of the dynamically consistent model to generate good initialization guesses for these Newton solvers. In particular, we demonstrate that Pseudo-Transient
Continuation (PTC) initialization (see [6, 17]) can greatly improve both speed and robustness when combined with Newton solvers.

It is difficult to model and optimize an entire product or system all at once. When dealing with large complex systems, it is first broken up into many simpler sub-problems. Each of these sub-problems should be substantial enough that they warrant the time dedicated to solving them but small enough that it can be feasibly solved. One of these sub-problems is modeling thermal fluid flow through a pipe. This sub-problem appears in everything from air conditioners to car engines and is the focus of this thesis.

Fluid is a general term for a substance that easily deforms under stress [29]. Although all fluids can be somewhat compressed, for modeling purposes, liquids (such as water) are considered in-compressible while gasses (such as air) are considered compressible. This distinction greatly impacts the complexity of a mathematical model. In general, in-compressible fluids are much easier to model than compressible fluids. When modeling a fluid, there are five main parameters to keep track of: velocity ($u$), internal energy ($e$), temperature ($T$), pressure ($p$), and density ($\rho$). These five quantities are related through the compressible Navier Stokes equations [25, 28]. In vector form, these equations are:

Continuity Equation: \[ \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho u) = 0 \]

Momentum Equation: \[ \frac{\partial u}{\partial t} + \nabla \cdot (\rho uu) = \rho f + \nabla \cdot \sigma \quad (1.1) \]

Energy Equation: \[ \frac{\partial e}{\partial t} + \nabla \cdot (\rho ue) = -\nabla \cdot (\rho u) + \nabla \cdot q + \nabla \cdot (\tau \cdot u) \]

In addition to the compressible Navier Stokes equations, additional equations of state are necessary to close the model. These equations of state should relate temperature and density to the pressure and enthalpy of the system. The in-compressible Navier Stokes equations can be obtained by assuming that the density of the fluid is constant [29]. This thesis will consider the compressible Navier Stokes equations.
Transport by fluid flow can be thought of as having two different components: a convective and diffusive part. The convective part of fluid flow describes how the bulk movement of the fluid transports fluid properties through the pipe. The diffusive part describes how the subatomic motion of molecules transports fluid properties \[29\]. The relative scale of each of these is described by a Peclet number which is defined as the rate of transport by convection divided by the rate of transport by conduction. Most flow regimes that people encounter in everyday life are almost entirely dominated by the convective component of flow. For example, a breeze can have a peclet number in the hundreds meaning that the convective transport is hundreds of times larger than the diffusive transport.

When faced with differences in magnitude of this order, it is common to neglect the diffusive component entirely. This assumption is valid in many real life applications. The version of the Compressible Navier Stokes equations without diffusion is known as the Euler equations \[2\]. This assumption is often made while modeling pipe flow due to the high volume of fluid that moves through pipes.

Even though diffusion can be a small effect, it is necessary for many pipe flow models. For example, consider the problem of optimizing the efficiency of home air conditioning units. Most home air conditioning units force their coolant at maximum velocity until the desired temperature is reached and then turn off. The coolant is either completely at rest or traveling relatively quickly. Recently, many companies have been developing partial flow air conditioners where the coolant isn’t always forced its maximum velocity. Instead, the velocity of the coolant is dictated by the required temperature change. If a small temperature change is required, then the coolant may be traveling slow enough where convection no longer dominates diffusion. If a model does not include diffusion, then simulations for low flow will be incorrect.

Solving the compressible Navier Stokes equations is a difficult task. Instead of solving
the differential equations exactly, a system of algebraic equations is created whose solution approximates the true solution. This method of approximating the original differential equations by a system of algebraic equations is known as discretization. A popular discretization method in fluid mechanics is the finite volume approach \[18\]. In the finite volume approach, the domain of interest is broken up into a mesh of volumes or elements and each quantity of interest is averaged across the element. The finite volume approach is well suited for the fluid dynamics because it is a conservative method. Physical quantities are conserved between elements. One flaw is that the finite volume method has difficulty handling diffusive terms. This stems from the fact that the finite volume method approximates the solution by piece-wise constant functions. Any approximation of a second derivative requires interpolation and is not natural to the method itself \[26\].

One way to resolve this issue is to approximate the solution by using higher order polynomials such as lines or quadratics instead of constants. This is the idea behind discontinuous Galerkin methods (DG Methods). DG Methods retain the conservation and stability properties of basic Finite Volume Methods but with the ability to better approximate second order spatial derivatives \[12\].

For many applications computing steady state solutions is the primary goal. This is especially true when a pipe flow reaches equilibrium relatively quickly. In this case, the system will spend a majority of its operation at steady state conditions. We will briefly describe the three methods of computing the steady state solution. The first method begins by assuming all temporal derivatives are zero and then solving the resulting system of ordinary differential equations. These equations are then appropriately discretized to create a system of non-linear algebraic equations. Solving these non-linear algebraic equations requires guessing an initial solution and then using this guess to initialize a Newton solver. Guessing a good initialization can be difficult. Furthermore, even if the Newton method converges there is no
Chapter 1. Introduction

guarantee that the solution attained through this method is physically realistic (e.g., may not be stable).

The second approach is to select an initial value for the dynamic model and let the time dependent dynamics run for a specified time $0 << t_f$. We know a steady state solution has been reached when the solution profile no longer changes by increasing $t_f$. This method requires that the numerical time integration scheme is stable for long times and is computationally efficient. Finding schemes with both of these properties can be difficult. Additionally, note that this method assumes that the system has a unique stable equilibrium point.

The third method combines the methods described above. Instead of letting the dynamic system run for a long time $0 << t_f$, we run the dynamics for a relatively short time (e.g, $t_f = 1$ time period). We then use the dynamic solution at time $t_f$ to initialize a Newton solver derived from the first method. Again this approach only works when the system has a unique stable equilibrium state. This is one version of so-called Pseudo-Transient Continuation (PTC) initialization methods and have been fine tuned for fluid flow equations (see [17]).

Chapter 2 provides relevant background information on the techniques and notation used in this thesis. In Chapter 3 we derive the 1D version for the compressible Navier Stokes equations. Additionally, we carefully document where 1D approximation errors occur so that future work can be performed with these errors in mind. In Chapter 4, we implement a piecewise linear DG method to discretize the system using both uniform and non-uniform meshes. Chapter 5 discusses the solution and implementation assumptions made while solving the system of differential algebraic equations derived in Chapter 4. Chapter 6 contains a brief review of steady state Newton solver results. In Chapter 7 we presents the numerical results for the dynamic solver and pseudo-transient initialization scheme. Finally, Chapter 8 summarizes the main results of this thesis and potential future work.
Chapter 2

Background

This chapter will provide some necessary background to understand the rest of this thesis. We will briefly describe the formulation of the finite volume method, the discontinuous Galerkin method, the choice of basis functions and notation, and an introduction to pseudo-transient initialization schemes.

2.1 The Finite Volume Method

In this section we will describe the basic procedure behind the Finite Volume method by discretization the advection equation. This discretization and all relevant information is based on the works of *Finite-Volume Methods for Hyperbolic Problems* by Leveque [16] and *Computational Fluid Dynamics: an Introduction* by John F. Wendt [27].

In its most general form, the finite volume method (FVM) is an integral method for solving ordinary differential equations (ODEs) and partial differential equations (PDEs). The basic idea behind FVM is to approximate the solution to one of these equations by using piece-wise constant functions. A typical FVM solution to a differential equation will look similar to Figure 2.1. In this figure, the red lines indicate the finite volume approximation $f_{FV}(x)$ to the true solution $f(x)$. The blue dashed lines indicate where element or volume boundaries are located. The finite volume method is best understood from the advection equation
Chapter 2. Background

Figure 2.1: Example of a finite volume method solution profile

Describing the bulk movement of a conserved property $\phi(t, x)$.

$$\frac{\partial \phi(t, x)}{\partial t} = - \frac{\partial \phi(t, x)}{\partial x} \quad x \in [0, L] \quad t \in [0, \infty] \quad (2.1)$$

$$\phi(0, x) = \alpha(x); \quad \phi(t, 0) = \beta(t)$$

The first step is to break the spatial domain into $N$ distinct subsections known as volumes or elements. For simplicity, we first assume uniformly sized elements of length $\frac{L}{N} = \Delta x$. However, this does not need to be the case. Next, we integrate the advection equation across an element $e_i = [x_{i-1}, x_i]$ where $i = 1, 2, \ldots, N$,

$$\int_{x_{i-1}}^{x_i} \frac{\partial \phi(t, x)}{\partial t} \, dx = - \int_{x_{i-1}}^{x_i} \frac{\partial \phi(t, x)}{\partial x} \, dx \quad (2.2)$$

We can now move the time derivative outside of the integral and use the divergence theorem to obtain:

$$\frac{\partial}{\partial t} \int_{x_{i-1}}^{x_i} \phi(t, x) \, dx = -\phi(t, x_i) + \phi(t, x_{i-1}) \quad (2.3)$$

The divergence theorem requires that we evaluate $\phi(t, x)$ at the element boundaries. Examining Figure 2.1, one can see that the finite volume approximation has two potential values at each node. One value when approaching from the left and another when approaching from the right. A method to resolve this issue is to assign a nodal value based on the function values in the left and right elements. The way that we assign this nodal value is known as
2.1. The Finite Volume Method

the numerical flux of the system and is denoted by $F(\cdot)$. We now use the approximation where the solution $\phi(t, x)$ is assumed to be constant on each control volume. This allows us to use the average value of $\phi(t, x)$ in $e_i$ where $i = 1, 2, ..., N$. The constant approximation is defined as:

$$
\overline{\phi}(t, x_{i - \frac{1}{2}}) = \frac{1}{\Delta x} \int_{x_{i-1}}^{x_i} \phi(t, x) \, dx \quad \Rightarrow \quad \overline{\phi}(t, x_{i - \frac{1}{2}}) \Delta x = \int_{x_{i-1}}^{x_i} \phi(t, x) \, dx
$$

(2.4)

For notation purposes: $\overline{\phi}(t, x_{i + \frac{1}{2}}) = \overline{\phi}_{i + \frac{1}{2}}(t)$. Plugging this approximation into the previous equation we obtain:

$$
\Delta x \frac{\partial}{\partial t} (\overline{\phi}_{i - \frac{1}{2}}(t)) = -F(\phi(t, x_i)) + F(\phi(t, x_{i-1}))
$$

(2.5)

All that is left now is to choose how to evaluate the numerical fluxes and to discretize the temporal derivative. In this example, we will use upwinding for the numerical flux. Upwinding is defined as:

$$
F(\phi(t, x_i)) = \overline{\phi}_{i - \frac{1}{2}}(t)
$$

(2.6)

We can apply upwinding to this problem because flow is in only one direction. However, many schemes exist to deal with numerical flux such as central difference, downwinding, and Lax Friedrichs.

Now we discretize the temporal derivative. We choose to use a Forward Euler scheme with uniform time step $\Delta t$. Notationally, we say: $\overline{\phi}_{i + \frac{1}{2}}(n\Delta t) = \overline{\phi}^n_{i + \frac{1}{2}}$ where $n = 0, 1, ...$ is the current time step. The fully discretized system becomes:

$$
\Delta x \frac{\overline{\phi}^{n+1}_{i - \frac{1}{2}} - \overline{\phi}^n_{i - \frac{1}{2}}}{\Delta t} = -\overline{\phi}^n_{i - \frac{1}{2}} + \overline{\phi}^n_{i - \frac{3}{2}}
$$

(2.7)

Rearranging for terms next time step we have:

$$
\overline{\phi}^{n+1}_{i - \frac{1}{2}} = \overline{\phi}^n_{i - \frac{1}{2}} + \frac{\Delta t}{\Delta x} (\overline{\phi}^n_{i - \frac{3}{2}} - \overline{\phi}^n_{i - \frac{1}{2}})
$$

(2.8)

In the next section, we will describe an extension of the finite volume method, the discontinuous Galerkin method.
2.2 The Discontinuous Galerkin Method

In this section we describe the discontinuous Galerkin method by discretizing the 1D advection equation using the discontinuous Galerkin Method. We then illustrate the basic method for discretizing second order terms. This discretization is described in the book *Nodal Discontinuous Galerkin Methods: algorithms, analysis, and applications* by Hesthave and Warburton [12] and *Numerical Methods for Conservation laws: From Analysis to Algorithms* by Hesthaven [11].

The discontinuous Galerkin method (DGM) can be thought of as a high order finite volume method. Instead of approximating the solution through a piece-wise continuous constant function, we instead approximate the solution by a piece-wise continuous polynomial. Figure 2.2 illustrates a piece-wise linear discontinuous Galerkin solution profile \( f_{DG}(x) \) in red. The true solution \( f(x) \) is in green and the element boundaries are the blue dashed lines.

![Figure 2.2: Example of a linear discontinuous Galerkin method solution profile](image)

Similar to the finite volume method, the discontinuous Galerkin method is easily understood from the advection equation given below:

\[
\frac{\partial \phi(t, x)}{\partial t} + \frac{\partial \phi(t, x)}{\partial x} = 0 \quad x \in [0, L] \quad t \in [0, \infty]
\]

\[
\phi(0, x) = \alpha(x); \quad \phi(t, 0) = \beta(t)
\]
2.2. The Discontinuous Galerkin Method

As in the FVM, we break the spatial domain into $N$ distinct subsections known as elements. Again for simplicity we assume uniformly sized elements of length $\frac{L}{N} = \Delta x$. Next, we multiply the entire system by a test function $\nu(x) \in H^1 = \{\nu(x) \text{ and } \nu'(x) \text{ are square integrable from } 0 \text{ to } L\}$

\[
\frac{\partial \phi(t, x)}{\partial t} \nu(x) + \frac{\partial \phi(t, x)}{\partial x} \nu(x) = 0 \tag{2.10}
\]

We now integrate across an element $e_i = [x_{i-1}, x_i]$ where $i = 1, 2, ... N$:

\[
\int_{x_{i-1}}^{x_i} \frac{\partial \phi(t, x)}{\partial t} \nu(x) \, dx + \int_{x_{i-1}}^{x_i} \frac{\partial \phi(t, x)}{\partial x} \nu(x) \, dx = 0 \tag{2.11}
\]

Integrating by parts allows us to move the derivative from $\phi(t, x)$ to the test function $\nu(x)$ and introduce flux into the system.

\[
\int_{x_{i-1}}^{x_i} \frac{\partial \phi(t, x)}{\partial t} \nu(x) \, dx + [\phi(t, x) \nu(x)]_x^{x_i} - \int_{x_{i-1}}^{x_i} \phi(t, x) \frac{\partial \nu(x)}{\partial x} = 0 \tag{2.12}
\]

At this point, we must now choose how we would like to approximate $\phi(t, x)$ both globally (across the entire mesh), and locally (each individual element). We will denote the approximate solution with an “h” subscript. For example, $\phi_h(t, x)$ is the approximate solution to $\phi(t, x)$. A common local approximation is $\phi_h(t, x)|_{e_i} \in \Pi_k = \{\text{Continuous polynomials up to degree } k\}$. Additionally, as implied by the name, we would also like to let the global solution be discontinuous. Thus, we say that the global approximate solution $\phi_h(t, x) \in V_k = \{V(\cdot) \in L^2_0(0, L) : V(\cdot)|_{e_i} \text{ is a polynomial of at most degree } k\}$. In other words, $V_k$ is the space of all piecewise continuous polynomials of at most degree $k$ defined on the interval $[0, L]$.

Once we have chosen the appropriate polynomial degree $k$, we can now represent any polynomial on element $e_i$ where $i = 1, 2, ... N$ by the linear combination of $k + 1$ basis functions denoted $\xi_j(x)$ $j = 0, 1, ..., k$. We will say that $\xi_j(x)$ is only non-zero on $e_i$ and zero everywhere else in the mesh. Thus, the approximate global solution $\phi_h(t, x)$ restricted to $x \in [x_{i-1}, x_i]$...
Chapter 2. Background

is given by

\[ \phi_h(t, x) = \sum_{j=0}^{k} c_j(t) \xi_j(x) \quad x \in [x_{i-1}, x_i] \]  

(2.13)

We can repeat this procedure over each element in the mesh to obtain a global representation of \( \phi_h(t, x) \). Notionally, we say \( \xi_{i,j}(x) \) is the j'th non-zero piece-wise continuous basis function defined on the i'th element of the mesh where \( i = 1, 2, ..., N \) and \( j = 0, 1, ..., k \). The same subscript notation applies the the time dependent coefficients. Thus, we have

\[ \phi_h(t, x) = \bigoplus_{r=1}^{N} \sum_{s=0}^{k} c_{r,s}(t) \xi_{r,s}(x) \quad x \in [0, L] \]  

(2.14)

The symbol \( \bigoplus \) means that we sum the local solutions everywhere except at the node boundaries. There, let the solution in both the left and right elements exist simultaneously. Now that we have a piece-wise continuous approximation of the solution, we can plug this into the advection equation:

\[ \int_{x_{i-1}}^{x_i} \frac{\partial \phi_h(t, x)}{\partial t} \nu(x) \, dx + [\phi_h(t, x) \nu(x)]_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} \phi_h(t, x) \frac{\partial \nu(x)}{\partial x} = 0 \]  

(2.15)

Expanding the approximate solution in terms of its piece-wise continuous approximation we obtain

\[ \int_{x_{i-1}}^{x_i} \frac{\partial}{\partial t} \bigoplus_{r=1}^{N} \sum_{s=0}^{k} c_{r,s}(t) \xi_{r,s}(x) \nu(x) \, dx + [\phi_h(t, x) \nu(x)]_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} \bigoplus_{r=1}^{N} \sum_{s=0}^{k} c_{r,s}(t) \xi_{r,s}(x) \frac{\partial \nu(x)}{\partial x} = 0 \]  

(2.16)

There are many choices to evaluate the flux term including upwinding, central, and downwinding. As such, we will not make a choice for this example and leave the flux term in its exact form. Next, we note that if \( r \neq i \) then \( \phi_{r,s} = 0 \). We can then simplify to

\[ \int_{x_{i-1}}^{x_i} \frac{\partial}{\partial t} \sum_{s=0}^{k} c_{i,s}(t) \xi_{i,s}(x) \nu(x) \, dx + [\phi_h(t, x) \nu(x)]_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} \sum_{s=0}^{k} c_{i,s}(t) \xi_{i,s}(x) \frac{\partial \nu(x)}{\partial x} = 0 \]  

(2.17)

Now, we must find \( \phi_h(t, x) \in V_k \) such that the above equation is satisfied for all \( \nu(x) \in H^1_0 = \)
2.2. The Discontinuous Galerkin Method

\{p(\cdot) \text{ and } p'(\cdot) \in L^2_0(x_{i-1}, x_i)\}. Since \(\nu(x)\) is arbitrary, we can choose it to be any function that we want, including one of the basis functions used to represent \(\phi_h(t, x)\). Thus, we let \(\nu(x) = \xi_{i,j}(x)\) for \(i = 1, 2, \ldots, N\) and \(j = 0, 1, \ldots k\). This lets us rewrite the previous equation as

\[
\int_{x_{i-1}}^{x_i} \partial_t \left[ \sum_{s=0}^{k} c_{i,s}(t) \xi_{i,s}(x) \right] \xi_{i,j}(x) \, dx + [\phi_h(t, x) \xi_{i,j}(x)]_{x_{i-1}}^{x_i} - \\
\int_{x_{i-1}}^{x_i} \sum_{s=0}^{k} c_{i,s}(t) \xi_{i,s}(x) \frac{\partial \xi_{i,j}(x)}{\partial x} = 0 \tag{2.18}
\]

We now evaluate the temporal derivative to obtain an ordinary differential equation given by

\[
\sum_{s=0}^{k} \dot{c}_{i,s}(t) \int_{x_{i-1}}^{x_i} \xi_{i,s}(x) \xi_{i,j}(x) \, dx + [\phi_h(t, x) \xi_{i,j}(x)]_{x_{i-1}}^{x_i} - \\
\sum_{s=0}^{k} c_{i,s}(t) \int_{x_{i-1}}^{x_i} \xi_{i,s}(x) \frac{\partial \xi_{i,j}(x)}{\partial x} = 0 \tag{2.19}
\]

Repeating these steps for all elements and all basis elements in the mesh will result in a system of ordinary differential equations. By solving these ordinary differential equations for all \(c_{i,j}(t)\) \(i = 1, 2, \ldots, N\) and \(j = 0, 1, \ldots k\) we can attain the approximate solution \(\phi_h(t, x)\) to the advection equation.

The DG Method For Diffusive Terms

We now consider how the DG method handles diffusive terms. It is tempting to perform integration by parts and then simply compute the derivative of the approximate solution. This idea is attractive because it seems natural since for \(k > 0\) the local approximate solution on \([x_{i-1}, x_i]\) will have a non-zero derivative. It is well known that this approach is weakly unstable \[30\]. Although this naive approach does converge, in some cases it converges to the incorrect solution \[30\]. Instead, we must re-write diffusive term as a system of first order differential equations. The following section follows the stable formulation presented in \[30\].
We begin by attempting to discretize the second order term $\frac{\partial^2 \phi(t,x)}{\partial x^2}$. As in the advective case, we multiply by a test function $\nu \in H^1$ and integrate across an element $e_i = [x_{i-1}, x_i]$ for $i = 1, 2, \ldots, N$ and multiply by a test function $\nu(x) \in H^1_0$.

$$\int_{x_{i-1}}^{x_i} \frac{\partial^2 \phi(t,x)}{\partial x^2} \nu(x) \, dx$$  \hspace{1cm} (2.20)

We assume that the approximate solution $\phi_h(t,x) \in V_k$ = can be written as the linear combination of $k + 1$ linearly independent basis functions $\xi_{i,j}(x)$ on element $e_i i = 1, 2, \ldots, N$ and $j = 0, 1, \ldots, k$. That is:

$$\phi_h(t,x) = \bigoplus_{r=1}^{N} \sum_{s=0}^{k} \phi_{r,s}(t) \xi_{r,s}(x)$$  \hspace{1cm} (2.21)

We now replace the test function, $\nu(x)$, with a basis function, $\xi_{i,j}(x)$ where $i = 1, 2, \ldots, N$ and $j = 0, 1, \ldots, k$, integrate by parts on each subinterval, and replace $\phi(t,x)$ with $\phi_h(t,x)$ to obtain:

$$\left. \left[ \frac{\partial \phi_h(t,x)}{\partial x} \xi_{i,j}(x) \right] \right|_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} \frac{\partial \phi_h(t,x)}{\partial x} \frac{\partial \xi_{i,j}(x)}{\partial x} \, dx$$  \hspace{1cm} (2.22)

The flux variable $\psi(t,x) = \frac{\partial \phi_h(t,x)}{\partial x}$ is assumed to have DG representation similar to $\phi(t,x)$.

This results in the following system of equations

$$\left[ \psi(t,x) \xi_{i,j}(x) \right]_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} \psi(t,x) \frac{\partial \xi_{i,j}(x)}{\partial x} \, dx$$

$$\psi(t,x) - \frac{\partial \phi_h(t,x)}{\partial x} = 0$$  \hspace{1cm} (2.23)

The same steps as before are performed on this second equation to obtain

$$\int_{x_{i-1}}^{x_i} \psi(t,x) \xi_{i,j}(x) \, dx + [\phi_h(t,x) \xi_{i,j}]_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} \phi_h(t,x) \frac{\partial \xi_{i,j}(x)}{\partial x} \, dx = 0$$  \hspace{1cm} (2.24)
2.3. The Linear Basis Functions

We now expand $\psi(t, x)$ and $\phi_h(t, x)$ by their DG piece-wise polynomial representation.

$$
\left[\psi(t, x)\xi_{i,j}(x)\right]_{x_i}^{x_{i+1}} - \int_{x_{i-1}}^{x_i} \sum_{s=0}^{k} \psi_{i,s}(t)\xi_{i,s}(x)\frac{\partial \xi_{i,j}(x)}{\partial x} \, dx
$$

$$
\int_{x_{i-1}}^{x_i} \sum_{s=0}^{k} \psi_{i,s}(t)\xi_{i,s}(x)\xi_{i,j}(x) \, dx + [\phi_h(t, x)\xi_{i,j}(x)]_{x_i}^{x_{i+1}} - \int_{x_{i-1}}^{x_i} \sum_{s=0}^{k} \phi_{i,s}(t)\xi_{i,s}(x)\frac{\partial \xi_{i,j}(x)}{\partial x} \, dx = 0
$$

Finally, we deal with the numerical flux terms for each equation. As noted in [30], the most stable flux to choose is an upwind flux for the diffusive term and a downwind flux for the flux variable correlation. It is important to note that $\psi(t, x)$ can be directly solved for in terms of $\phi_h(t, x)$. One could represent $\psi(t, x)$ entirely in terms of $\phi_h(t, x)$ and eliminate $\psi(t, x)$ from the formulation completely. In the next section, we describe in detail the basis functions we use for the discretization of the compressible Navier Stokes Equations.

2.3 The Linear Basis Functions

In this thesis we use Lagrangian piece-wise linear functions as the basis functions for the approximate solution. That is, each basis function is equal to one at a single node and zero at all nodes in the mesh. Additionally, each basis function is only non-zero on a single element or volume. The solution on each element $i = 1, 2, ... N$ is described by two basis functions $\phi_{i,0}(x)$ and $\phi_{i,1}(x)$. We define these basis functions as follows:

$$
\phi_{i,0}(x) = \begin{cases} 
\frac{x_{i+1}-x}{x_{i+1}-x_i} & x \in [x_i, x_{i+1}] \\
0 & \text{otherwise}
\end{cases}
$$

$$
\phi_{i,1}(x) = \begin{cases} 
\frac{x-x_i}{x_{i+1}-x_i} & x \in [x_i, x_{i+1}] \\
0 & \text{otherwise}
\end{cases}
$$

17
Chapter 2. Background

Figure 2.3 illustrates locally linear basis functions. Each basis function is colored differently to demonstrate that each basis function is independent from all other basis functions. To put it simply, basis functions do not have to match at element boundaries. The only reason they have the same value on the boundary in 2.3 is because all basis functions are scaled to one at their respective node.

![Figure 2.3: Example of a piece-wise linear basis functions](image)

We refer to the basis function that is one at the left boundary of an element (j=0) as the left basis function. Similarly, we refer to the basis function that is one on the right boundary of an element (j=1) as the right basis function. In Figure 2.3 the green basis function ($\phi_{i-1,0}$) is the right basis function on Volume i-1 while the blue basis function ($\phi_{i,0}$) is the right basis function on Volume i. We can now represent every quantity as a linear combination of these basis functions. For the arbitrary quantity $\psi_h(t, x)$

$$\psi_h(t, x) = \bigoplus_{i=1}^{N} \psi_{i,0}(t)\phi_{i,0}(x) + \psi_{i,1}(t)\phi_{i,1}(x), \quad x \in [0, L]$$  \hspace{1cm} (2.26)

Below we list the piece-wise linear DG approximation for all variables in this thesis:

$$\rho(t, x) \approx \rho_h(t, x) = \bigoplus_{i=1}^{N} \rho_{i,0}(t)\phi_{i,0}(x) + \rho_{i,1}(t)\phi_{i,1}(x), \quad x \in [0, L]$$  \hspace{1cm} (2.27)

$$h(t, x) \approx h_h(t, x) = \bigoplus_{i=1}^{N} h_{i,0}(t)\phi_{i,0}(x) + h_{i,1}(t)\phi_{i,1}(x), \quad x \in [0, L]$$  \hspace{1cm} (2.28)
2.4. Psuedo-Transient Initialization

\[ u(t, x) \approx u_h(t, x) = \bigoplus_{i=1}^{N} u_{i,0}(t) \phi_{i,0}(x) + u_{i,1}(t) \phi_{i,1}(x), \quad x \in [0, L] \]  
(2.29)

\[ p(t, x) \approx p_h(t, x) = \bigoplus_{i=1}^{N} p_{i,0}(t) \phi_{i,0}(x) + p_{i,1}(t) \phi_{i,1}(x), \quad x \in [0, L] \]  
(2.30)

\[ T(t, x) \approx T_h(t, x) = \bigoplus_{i=1}^{N} T_{i,0}(t) \phi_{i,0}(x) + T_{i,1}(t) \phi_{i,1}(x), \quad x \in [0, L] \]  
(2.31)

Figure 2.4 illustrates how these continuous local linear basis functions are added together to form an approximate solution.

Figure 2.4: Illustration of notation used in describing the DG linear basis functions and a local approximate solution for arbitrary quantity \( \psi(t, x) \)

2.4 Psuedo-Transient Initialization

One objective of this thesis is to compute numerical solutions to for obtaining the steady state version of the compressible Navier Stokes equations. A steady state solution to a system of differential equations is defined a state when the solution profile no longer changes with time. One potential method to attain a steady state solution is to solve the dynamical...
system and then compute the limit at \( t \to +\infty \). Assuming a correct dynamical model and a stable time integrator, this method of attaining the steady state solution is guaranteed to give a physical solution for any physically consistent initial condition. However, to find the true steady state profile we would need to run the dynamical system for an infinite amount of time; stopping the integration at any finite \( t \) is only an approximation of the true steady state solution.

Furthermore, simply integrating for a long period of time has problems. If the solution varies too rapidly, then explicit time integrator requires increasingly small step sizes to remain numerically stable \([24]\). One way to classify the stability of a time integration scheme is through A-stability. If a time integration method is A-stable, then when applied to \( y'(t) = qy(t) \) where \( q > 0 \), the approximate solution will always tend towards zero even with very large time steps.

Figure 2.5 shows the Adam’s Bashforth and backwards Euler time integration schemes applied to \( y'(t) = -11y(t) \). The A-instability Adam’s Bashforth can be seen. The approximate solution grows instead of decaying to zero as time marches forward. On the other hand, backwards Euler is A-stable. Even though it has a much larger time step, the approximate solution decays to zero. This is true no matter how long the system is integrated or how large the time step. Although this is a simple example, no multi-step explicit method is A-stable \([5]\). All multi-step explicit methods will exhibit this same behavior; the problem simply needs to decay fast enough. This fact means that for some systems step sizes need to be so small that explicit integrators are no longer feasible. This may force one to use an implicit integrator which can be computationally expensive \([24]\).
2.4. Pseudo-Transient Initialization

Figure 2.5: An example illustrating the A-stability of backwards Euler and the A-instability of Adams Bashforth.

Another method to obtain this steady state solution is to simply set the temporal derivative equal to zero and solve the resulting system of algebraic and differential equations. The resulting discretized differential and algebraic equations is non-linear. Thus, we need an algorithm to solve a system of non-linear equations. One of the most popular algorithms, and the one utilized in this thesis, is Newton’s Method. The first step in applying Newton’s method is to provide an initial guess for the solution.

This poses a problem since Newton’s Method is not guaranteed to converge for all initial guesses. Under suitable conditions one can show that there is a basin of attraction around a solution such that for an initial guess inside that basin the Newton iterations converge...
Each solution to the system of non-linear equations is associated with a basin of attraction which is the set of initial guesses that are guaranteed to converge to a particular solution. If the initial guess is outside of the basin of attraction, then Newton’s Method may not converge. Figure 2.6 gives a pictorial representation of what a basin of attraction might look like. Furthermore, basins of attraction do not guarantee anything about the rate of convergence. An initial guess in the basin of attraction may take a large number of iterations to converge. That said, Newton’s method does guarantee a region of quadratic convergence for all solutions [15]. In the region of quadratic convergence Newton’s method is guaranteed to converge quadratically. In simple terms, this means that the error at the next iteration is a proportion of the square of the error at the previous step [23].

Even if Newton’s method converges, it may not converge to physically meaningful solution. The non-linear system of equations may have many solutions; however, not all of them are physically realistic. Moreover, the steady state solution may not be dynamically stable even
2.4. Psuedo-Transient Initialization

if it is known that the physically realistic solution is stable. Thus, one needs to find an initial guess that belongs to the basin of attraction for a physically meaningful solution.

The pseudo-transient initialization scheme is a method that attempts to generate “better” initializations for the case when one seeks a stable solution. The basic idea is to use a “time-marching scheme” to generate a sequence of iterates that eventually fall into the basin of attraction for the desired stable solution. Newton’s method. Ideally, the iterates lands in the region of quadratic convergence. Figure 2.7 illustrates the idea behind a pseudo-transient initialization approach.

![Figure 2.7: Illustration behind idea of pseudo-transient initialization](image)

In a pseudo-transient initialization approach, we first initialize the dynamical system with a physically consistent initial condition. This initial condition can theoretically be anywhere as long as it is physically consistent. We then use the dynamics to propagate this initial condition forward. After propagating the initial condition forward for a “small amount of time”, we use the resulting profile as an initial guess for the steady state Newton’s solver. If
done correctly, the initial guess obtained from the dynamics should be in the Newton solver’s region of quadratic convergence. This guarantees that the Newton Solver will converge quadratically to a physically meaningful solution.

We have just described the simplest way of implementing a pseudo-transient initialization scheme. A significant amount of work has been performed to optimize this scheme, especially with regards to fluid flow equations. Kelley and Keyes [15] have outlined methods to adjust the time step size corresponding to how close the current iterate is from the true steady state solution for the compressible Euler equations. They suggest using small time steps in the beginning of the initialization scheme and then slowly increasing the length of the time step as the integration progresses yields accurate and efficient results. Ceze and Fidkowski [3] analyzed different time stepping methods. Each scheme has its own level of robustness and efficiency. Hovland and McInnes [13] have parallelized implicit Euler time stepping for the full 3D Euler equations using a Newton-Krylov-Schwarz technique for solving the system of non-linear equations at each time step. This is important for exceptionally large problems (millions of unknowns). These results are just a few examples of how pseudo-transient initialization has been optimized over the past thirty years. Other approaches include the use of residual smoothing to improve the Newton solver convergence rate and implementing the scheme under various equality/inequality constraints [4, 17].

In the next chapter we will derive the 1D Navier Stokes equations from the full 3D Navier Stokes equations.
Chapter 3

Derivation of the 1D Navier Stokes Equations

In this chapter we derive a 1-D representation of the Navier Stokes equations for pipe flow. We begin with the vector form of the 3-D Navier Stokes equations given by

Continuity Equation:
\[ \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \]

Momentum Equation:
\[ \frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = \rho \mathbf{f} + \nabla \cdot \sigma \]  

Energy Equation:
\[ \frac{\partial p_e}{\partial t} + \nabla \cdot (\rho e \mathbf{u}) = \nabla \cdot (p \mathbf{u}) + \nabla \cdot q + \nabla \cdot (\tau \cdot \mathbf{u}) \]  

The meanings of all variables can be found in the nomenclature section of this thesis. We will first derive the 1D continuity equation, then the momentum equation, and finally the enthalpy equation. After deriving all three equations, we will discuss closing the model using temperature correlations and equations of state.

3.1 Continuity Equation

The 3D vector form of the continuity equation is

\[ \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \]  

First, we make the 1-D flow assumption: we assume that the fluid only moves in the x-direction, i.e v=0 and w=0. This allows us to re-write the continuity equation as:

\[ \frac{\partial \rho}{\partial t} + \frac{\partial (\rho u)}{\partial x} = 0 \]  

(3.3)
Next, we integrate the continuity equation on the y-z cross-sectional surface of the pipe to obtain:

\[ \int \int_A \frac{\partial \rho}{\partial t} \, dy \, dz + \int \int_A \frac{\partial (\rho u)}{\partial x} \, dy \, dz = 0 \quad (3.4) \]

It is assumed that all variables are continuously differential of order two. Moving the derivatives outside of the integrals yields:

\[ \frac{\partial}{\partial t} \int \int_A \rho \, dy \, dz + \frac{\partial}{\partial x} \int \int_A (\rho u) \, dy \, dz = 0 \quad (3.5) \]

We now define the area average of \( \rho \) and \( \rho u \) as

\[ \bar{\rho} = \frac{1}{A} \int \int_A \rho \, dy \, dz \iff A\bar{\rho} = \int \int_A \rho \, dy \, dz, \quad (3.6) \]

and

\[ \bar{\rho} \bar{u} = \frac{1}{A} \int \int_A \rho u \, dy \, dz \iff A\bar{\rho} \bar{u} = \int \int_A \rho u \, dy \, dz, \quad (3.7) \]

respectively. Thus, the area-averaged continuity equation becomes

\[ \frac{\partial A\bar{\rho}}{\partial t} + \frac{\partial A\bar{\rho} \bar{u}}{\partial x} = 0 \quad (3.8) \]

It is readily apparent that \( \bar{\rho} \bar{u} \) is not easily computable. We generate computable variables by introducing a mass weighted average, known as Favre-Filtering defined by

\[ \hat{u} = \frac{\bar{\rho} \bar{u}}{\bar{\rho}} \iff \bar{\rho} \hat{u} = \bar{\rho} \bar{u}. \quad (3.9) \]

The final 1D continuity equation is:

\[ \frac{\partial A\bar{\rho}}{\partial t} + \frac{\partial A\bar{\rho} \hat{u}}{\partial x} = 0. \quad (3.10) \]

In the next section we derive the 1D momentum equation.
3.2 Momentum Equation

We begin with the 3D momentum equation

\[
\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = \rho \mathbf{f} + \nabla \cdot \sigma \tag{3.11}
\]

The relationship between the stress tensor \(\sigma\) and the shear stress tensor \(\tau\) is

\[
\sigma = -p\mathbf{I} + \tau - \left( \frac{2}{3}\mu \nabla \cdot \mathbf{u} \right) \mathbf{I} \tag{3.12}
\]

Substituting this relationship into the 3D momentum equation produces the following equations

\[
\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = \rho \mathbf{f} - \nabla \cdot p + \nabla \cdot \mathbf{\tau} - \nabla \cdot \left( \frac{2}{3}\mu \nabla \cdot \mathbf{u} \right) \mathbf{I} \tag{3.13}
\]

We now assume that the fluid we are working with is a compressible, Newtonian fluid. Additionally, we assume that the stress tensor is a linear function of strain rates, the fluid isotropic, and that when the fluid is at rest \(\nabla \cdot \tau = 0\). These assumptions allow us to write the shear stress tensor as

\[
\tau_{ij} = \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \tag{3.14}
\]

Assuming that pressure is constant across the cross-sectional area of the pipe, we expand the resulting momentum equations in the \(x\), \(y\) and \(z\) directions. These are given by

\[
\frac{\partial (\rho u)}{\partial t} + \frac{\partial (\rho u u)}{\partial x} + \frac{\partial (\rho u v)}{\partial x} + \frac{\partial (\rho u w)}{\partial x} = \rho f_x - \frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left( \frac{2\mu}{\partial x} - \frac{2}{3}\mu \nabla \cdot \mathbf{u} \right) + \frac{\partial}{\partial y} \left( \mu \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right) + \frac{\partial}{\partial z} \left( \mu \left( \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \right) \tag{3.15}
\]
Chapter 3. Derivation of the 1D Navier Stokes Equations

\[
\frac{\partial (\rho v)}{\partial t} + \frac{\partial (\rho vu)}{\partial y} + \frac{\partial (\rho vv)}{\partial y} + \frac{\partial (\rho vw)}{\partial y} = \rho f_2 - \frac{\partial p}{\partial y} + \frac{\partial}{\partial x} \left( \mu \left( \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right) \right) + \frac{\partial}{\partial y} \left( 2\mu \frac{\partial v}{\partial y} - \frac{2}{3} \mu \nabla \cdot \mathbf{u} \right) + \frac{\partial}{\partial z} \left( \mu \left( \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \right) \tag{3.16}
\]

\[
\frac{\partial (\rho w)}{\partial t} + \frac{\partial (\rho wu)}{\partial z} + \frac{\partial (\rho vw)}{\partial z} + \frac{\partial (\rho ww)}{\partial z} = \rho f_3 - \frac{\partial p}{\partial z} + \frac{\partial}{\partial x} \left( \mu \left( \frac{\partial w}{\partial x} + \frac{\partial u}{\partial y} \right) \right) + \frac{\partial}{\partial y} \left( \mu \left( \frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right) \right) + \frac{\partial}{\partial z} \left( 2\mu \frac{\partial w}{\partial z} - \frac{2}{3} \mu \nabla \cdot \mathbf{u} \right) \tag{3.17}
\]

Now, as in the continuity equation, we make the 1-D flow assumption: the fluid only moves in the x-direction, i.e. \( v = 0 \) and \( w = 0 \). This allows us to reduce the momentum equation down to just the first equation and simplify it to

\[
\frac{\partial (\rho u)}{\partial t} + \frac{\partial (\rho uu)}{\partial x} = \rho f_1 - \frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left( \frac{4}{3} \mu \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left( \mu \left( \frac{\partial u}{\partial y} \right) \right) + \frac{\partial}{\partial z} \left( \mu \left( \frac{\partial u}{\partial z} \right) \right) \tag{3.18}
\]

Integrating across the cross-sectional area of the pipe results in

\[
\int_A \int_A \frac{\partial (\rho u)}{\partial t} \ dy \ dz + \int_A \int_A \frac{\partial (\rho uu)}{\partial x} \ dy \ dz = \int_A \int_A \rho f_1 \ dy \ dz - \int_A \int_A \frac{\partial p}{\partial x} \ dy \ dz + \int_A \int_A \frac{\partial}{\partial x} \left( \frac{4}{3} \mu \frac{\partial u}{\partial x} \right) \ dy \ dz + \int_A \int_A \frac{\partial}{\partial y} \left( \mu \left( \frac{\partial u}{\partial y} \right) \right) \ dy \ dz + \int_A \int_A \frac{\partial}{\partial z} \left( \mu \left( \frac{\partial u}{\partial z} \right) \right) \ dy \ dz \tag{3.19}
\]

As in the continuity equation, we assume that the variables are continuously differential of order two. This allows the derivatives and integrals to commute. We bring the integrals
3.2. Momentum Equation

inside of the derivatives.

\[
\frac{\partial}{\partial t} \iint_A (\rho u) \, dy \, dz + \frac{\partial}{\partial x} \iint_A (\rho uu) \, dy \, dz = \iint_A \rho f_1 \, dy \, dz - \frac{\partial}{\partial x} \iint_A p \, dy \, dz + \frac{4}{3} \frac{\partial}{\partial x} \left( \iint_A \mu \frac{\partial u}{\partial x} \, dy \, dz \right) + \frac{\partial}{\partial y} \left( \iint_A \mu \frac{\partial u}{\partial y} \, dy \, dz \right) + \frac{\partial}{\partial z} \left( \iint_A \mu \frac{\partial u}{\partial z} \, dy \, dz \right)
\]

(3.20)

We now define a cross-sectional area average for all variables. These are listed below:

\[
\overline{pu} = \frac{1}{A} \iint_A \rho u \, dy \, dz \Leftrightarrow A\overline{pu} = \iint_A \rho u \, dy \, dz
\]

(3.21)

\[
\overline{puu} = \frac{1}{A} \iint_A \rho uu \, dy \, dz \Leftrightarrow A\overline{puu} = \iint_A \rho uu \, dy \, dz
\]

(3.22)

\[
\overline{f_1} = \iint_A \rho f_1 \, dy \, dz \Leftrightarrow A\overline{f_1} = \iint_A \rho f_1 \, dy \, dz
\]

(3.23)

\[
\iint_A p \, dy \, dz = Ap
\]

(3.24)

\[
\overline{\mu \frac{\partial u}{\partial x}} = \frac{1}{A} \iint_A \mu \frac{\partial u}{\partial x} \, dy \, dz \Leftrightarrow A\overline{\mu \frac{\partial u}{\partial x}} = \iint_A \mu \frac{\partial u}{\partial x} \, dy \, dz
\]

(3.25)

It is assumed that the normal stresses along the walls cause frictional forces experienced by the fluid as it moves along the walls. We represent these friction forces by

\[
\frac{\partial}{\partial y} \left( \iint_A \mu \left( \frac{\partial u}{\partial y} \right) \, dy \, dz \right) + \frac{\partial}{\partial z} \left( \iint_A \mu \left( \frac{\partial u}{\partial z} \right) \, dy \, dz \right) = F_F
\]

(3.26)

The momentum equation becomes

\[
\frac{\partial (A\overline{p})}{\partial t} + \frac{\partial (A\overline{pu})}{\partial x} = A\overline{f_1} - \frac{\partial Ap}{\partial x} + \frac{4}{3} \frac{\partial}{\partial x} A \left( \overline{\mu \frac{\partial u}{\partial x}} \right) + F_F
\]

(3.27)

Next, we assume that the only body forces on the fluid is gravity. This implies that the body forces acting on each cross-section of the pipe is constant. The momentum equation simplifies to

\[
\frac{\partial (A\overline{p})}{\partial t} + \frac{\partial (A\overline{pu})}{\partial x} = A\overline{f_1} - \frac{\partial Ap}{\partial x} + \frac{4}{3} \frac{\partial}{\partial x} A \left( \overline{\mu \frac{\partial u}{\partial x}} \right) + F_F
\]

(3.28)
Chapter 3. Derivation of the 1D Navier Stokes Equations

We use the Favre-Filter on each group of variables to produce computable variables. The Favre-Filter uses mass weighted averages defined as

\[ \hat{u} = \frac{\bar{u}}{\bar{\rho}} \iff \bar{\rho} \hat{u} = \bar{\rho} \bar{u}, \]  
\tag{3.29}

and

\[ \hat{\bar{u}} \hat{u} = \frac{\bar{u} \bar{u}}{\bar{\rho}} \iff \bar{\rho} \hat{\bar{u}} \hat{u} = \bar{\rho} \bar{u} \bar{u}, \]  
\tag{3.30}

respectively. Residuals are introduced to allow us to compute the mass weighted averages. These residuals are given by

\[ \hat{\bar{u}} \hat{u} = \hat{u} \hat{u} + [G_\star, \Psi](u, u) = \hat{u} \hat{u} + \tau^r \]  
\tag{3.31}

Some terms cannot be easily approximated by a mass weighted average. To deal with this, we replace them with their computable version plus a residual.

\[ \mu \frac{\partial \hat{u}}{\partial x} = \mu \frac{\partial \hat{u}}{\partial x} + \Pi_{diff} \]  
\tag{3.32}

Using these filters, we are able to write the momentum equation as

\[ \frac{\partial (A \bar{\rho} \hat{u})}{\partial t} + \frac{\partial (A \bar{\rho} \hat{u}^2)}{\partial x} + \frac{\partial (A \bar{\rho} \tau^r)}{\partial x} = A \bar{p} \hat{f}_1 - \frac{\partial A p}{\partial x} + 4 \frac{\partial}{\partial x} A \left( \hat{\mu} \frac{\partial \hat{u}}{\partial x} \right) + \frac{4}{3} \frac{\partial}{\partial x} \left( A \Pi_{diff} \right) + F_F(t, x) \]  
\tag{3.33}

Finally, since we assume that the only body forces acting on the fluid is gravity, we can write the following relationship

\[ f_1 = -g \frac{\partial z}{\partial x} \]  
\tag{3.34}

The final 1D momentum equation is

\[ \frac{\partial (A \bar{\rho} \hat{u})}{\partial t} + \frac{\partial (A \bar{\rho} \hat{u}^2)}{\partial x} + \frac{\partial (A \bar{\rho} \tau^r)}{\partial x} = -A \bar{p} g \frac{\partial z}{\partial x} - \frac{\partial A p}{\partial x} + 4 \frac{\partial}{\partial x} A \left( \hat{\mu} \frac{\partial \hat{u}}{\partial x} \right) + \frac{4}{3} \frac{\partial}{\partial x} \left( A \Pi_{diff} \right) + F_F \]  
\tag{3.35}

In the next section, we will discuss reducing the 3D Navier Stokes Energy Equation down to 1D.
3.3 The Energy Equation

We begin with the 3d equation for conservation of energy

\[
\frac{\partial \rho e}{\partial t} + \nabla \cdot (\rho u e) = -\nabla \cdot (\rho u) + \nabla \cdot q + \nabla \cdot (\tau \cdot u) \tag{3.36}
\]

Like in the momentum equation, it is assumed that the fluid is a compressible, Newtonian fluid. Additionally, we assume the stress tensor is a linear function of strain rates, the fluid isotropic, and that when the fluid is at rest \( \nabla \cdot \tau = 0 \). Additionally, we assume that the pressure is uniform across the cross-sectional face of the pipe. These assumptions allow us to expand the initial energy equation as

\[
\frac{\partial \rho e}{\partial t} + \frac{\partial \rho u e}{\partial x} + \frac{\partial \rho v e}{\partial y} + \frac{\partial \rho w e}{\partial z} = -\frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} + \frac{\partial \rho w}{\partial z} + \frac{\partial q}{\partial x} + \frac{\partial q}{\partial y} + \frac{\partial q}{\partial z}
\]

\[
+ \frac{\partial}{\partial x} \left( \mu \left( 2u \frac{\partial u}{\partial x} + v \left( \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right) + w \left( \frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) \right) \right)
\]

\[
+ \frac{\partial}{\partial y} \left( \mu \left( 2v \frac{\partial v}{\partial y} + u \left( \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right) + w \left( \frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right) \right) \right)
\]

\[
+ \frac{\partial}{\partial z} \left( \mu \left( 2w \frac{\partial w}{\partial z} + v \left( \frac{\partial w}{\partial z} + \frac{\partial v}{\partial y} \right) + u \left( \frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) \right) \right) \tag{3.37}
\]

We now make the 1D flow assumption, that is that \( v = 0 \) and \( w = 0 \). This leads to the following equation

\[
\frac{\partial \rho e}{\partial t} + \frac{\partial \rho u e}{\partial x} = -\frac{\partial \rho u}{\partial x} + \frac{\partial q}{\partial x} + \frac{\partial q}{\partial y} + \frac{\partial q}{\partial z} + \frac{\partial}{\partial x} \left( 2\mu \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left( \mu u \frac{\partial u}{\partial y} \right) + \frac{\partial}{\partial z} \left( \mu u \frac{\partial u}{\partial z} \right) \tag{3.38}
\]

Integrating across the cross-sectional face of the pipe results in

\[
\int \int_A \frac{\partial \rho e}{\partial t} dy \, dz + \int \int_A \frac{\partial \rho u e}{\partial x} dy \, dz = -\int \int_A \frac{\partial \rho u}{\partial x} dy \, dz
\]

\[
+ \int \int_A \frac{\partial q}{\partial x} dy \, dz + \int \int_A \frac{\partial q}{\partial y} dy \, dz + \int \int_A \frac{\partial q}{\partial z} dy \, dz
\]

\[
+ \int \int_A \frac{\partial}{\partial x} \left( 2\mu u \frac{\partial u}{\partial x} \right) dy \, dz
\]

\[
+ \int \int_A \frac{\partial}{\partial y} \left( \mu u \frac{\partial u}{\partial y} \right) dy \, dz
\]

\[
+ \int \int_A \frac{\partial}{\partial z} \left( \mu u \frac{\partial u}{\partial z} \right) dy \, dz \tag{3.39}
\]
We assume that the variables are continuously differential of order two, allowing us to interchange integration and differentiation. We then move all of the integrals inside of the derivatives to produce

\[
\begin{align*}
\frac{\partial}{\partial t} \int_A (\rho e) \, dy \, dz + \frac{\partial}{\partial x} \int_A (\rho ue) \, dy \, dz &= - \frac{\partial}{\partial x} \int_A (pu) \, dy \, dz + \frac{\partial}{\partial y} \int_A (q) \, dy \, dz \\
&+ \frac{\partial}{\partial z} \int_A (q) \, dy \, dz \\
&+ \frac{\partial}{\partial x} \left( \int_A \left( 2\mu \frac{\partial u}{\partial x} \right) \, dy \, dz \right) \\
&+ \frac{\partial}{\partial y} \left( \int_A \left( \mu \frac{\partial u}{\partial y} \right) \, dy \, dz \right) \\
&+ \frac{\partial}{\partial z} \left( \int_A \left( \mu \frac{\partial u}{\partial z} \right) \, dy \, dz \right)
\end{align*}
\]

(3.40)

It is assumed that heat conduction in the y and z directions becomes heat transfer into and out of the pipe. We represent this heat transfer as

\[
\frac{\partial}{\partial y} \int_A (q) \, dy \, dz + \frac{\partial}{\partial z} \int_A (q) \, dy \, dz = \dot{Q}
\]

(3.41)

Viscous dissipation due to stress in the fluid is denoted by

\[
\Phi = \frac{\partial}{\partial x} \left( \int_A \left( 2\mu \frac{\partial u}{\partial x} \right) \, dy \, dz \right) + \frac{\partial}{\partial y} \left( \int_A \mu \left( \frac{\partial u}{\partial y} \right) \, dy \, dz \right) + \frac{\partial}{\partial z} \left( \int_A \mu \left( \frac{\partial u}{\partial z} \right) \, dy \, dz \right)
\]

(3.42)

All variables are then area averaged. We write these area averaged variables as:

\[
\overline{\rho e} = \frac{1}{A} \int_A \rho e \, dy \, dz \quad \Rightarrow \quad A\overline{\rho e} = \int_A \rho e \, dy \, dz
\]

(3.43)

\[
\overline{\rho ue} = \frac{1}{A} \int_A \rho ue \, dy \, dz \quad \Rightarrow \quad A\overline{\rho ue} = \int_A \rho ue \, dy \, dz
\]

(3.44)

\[
\overline{q} = \frac{1}{A} \int_A q \, dy \, dz \quad \Rightarrow \quad A\overline{q} = \int_A q \, dy \, dz
\]

(3.45)

\[
\overline{pu} = \frac{1}{A} \int_A pu \, dy \, dz \quad \Rightarrow \quad A\overline{pu} = \int_A pu \, dy \, dz
\]

(3.46)
3.3. The Energy Equation

The area averaged 1D energy equation is

\[
\frac{\partial A \bar{p} \bar{e}}{\partial t} + \frac{\partial A \bar{p} \bar{u} \bar{e}}{\partial x} = -\frac{\partial A \bar{p} \bar{u}}{\partial x} + \frac{\partial A \hat{q}}{\partial x} + \dot{Q} + \Phi \tag{3.47}
\]

We now apply the Favre-Filter to the enthalpy equation. Each term is shown individually for clarity:

\[
\frac{\partial A \bar{p} \bar{e}}{\partial t} = \frac{\partial A \bar{p} \hat{e}}{\partial t} \tag{3.48}
\]

\[
\frac{\partial A \bar{p} \bar{u} \bar{e}}{\partial x} = \frac{\partial A \bar{p} \bar{u} \hat{e}}{\partial x} \tag{3.49}
\]

\[
\frac{\partial A \bar{p}}{\partial x} = \frac{\partial A \bar{p}}{\partial x} + \frac{\partial A p_{close}}{\partial x} \tag{3.50}
\]

\[
\frac{\partial A \hat{q}}{\partial x} = \frac{\partial A \hat{q}}{\partial x} + \frac{\partial A q_{close}}{\partial x} \tag{3.51}
\]

We define the Favre-filtered density average heat flux to be

\[
\hat{q} = \kappa(\bar{T}) \frac{\partial \bar{T}}{\partial x} \tag{3.52}
\]

Similarly, the Favre-filtered density average viscous dissipation is

\[
\hat{\Phi} = \frac{\partial}{\partial x} \left( 2\mu(\bar{T}) \bar{u} \frac{\partial \bar{u}}{\partial x} \right) dy \, dz + \frac{\partial}{\partial y} \left( \mu(\bar{T}) \bar{u} \left( \frac{\partial \bar{u}}{\partial y} \right) \right) dy \, dz + \frac{\partial}{\partial z} \left( \mu(\bar{T}) \bar{u} \left( \frac{\partial \bar{u}}{\partial z} \right) \right) dy \, dz \tag{3.53}
\]

This is related to the area averaged viscous dissipation through

\[
\bar{\Phi} = \hat{\Phi} + \Phi_{res} \tag{3.54}
\]

The 1D Favre-filtered energy equation is:

\[
\frac{\partial A \bar{p} \hat{e}}{\partial t} + \frac{\partial A \bar{p} \bar{u} \hat{e}}{\partial x} = -\frac{\partial A \bar{p} \bar{u}}{\partial x} - \frac{\partial A p_{close}}{\partial x} + \frac{\partial A \hat{q}}{\partial x} + \frac{\partial A q_{close}}{\partial x} + \dot{Q} + \hat{\Phi} + \Phi_{res} \tag{3.55}
\]
We summarize the 1D Navier Stokes equations with and without residual terms below:

### 1D Navier Stokes Equations with Residuals

Here we list the continuity, momentum, and energy equations with all residual terms.

**Continuity:** \[
\frac{\partial \Delta \rho}{\partial t} + \frac{\partial \Delta \rho \hat{u}}{\partial x} = 0
\]

**Momentum:** \[
\frac{\partial (\Delta \rho \hat{u})}{\partial t} + \frac{\partial (\Delta \rho \hat{u}^2)}{\partial x} + \frac{\partial (\Delta \rho \tau_x)}{\partial x} = -A \Delta \rho g \frac{dz}{dx} - \frac{\partial \Delta p}{\partial x} + \frac{4}{3} \frac{\partial A}{\partial x} \left( \frac{\partial \hat{u}}{\partial x} \right)^2 + \frac{4}{3} \frac{\partial (\Delta \Pi_{diff})}{\partial x} + F_F
\]

**Energy:** \[
\frac{\partial \Delta \hat{e}}{\partial t} + \frac{\partial \Delta \hat{u} \hat{e}}{\partial x} = -\frac{\partial \Delta \hat{p}}{\partial x} - \frac{\partial A_{\Delta \hat{p} close}}{\partial x} + \frac{\partial A_{\Delta \hat{q} close}}{\partial x} + \hat{Q} + \Phi + \Phi_{res}
\]

### 1D Navier Stokes Equations without Residuals

Here we list the continuity, momentum, and energy equations with no residual terms.

**Continuity:** \[
\frac{\partial \Delta \rho}{\partial t} + \frac{\partial \Delta \rho \hat{u}}{\partial x} = 0
\]

**Momentum:** \[
\frac{\partial (\Delta \rho \hat{u})}{\partial t} + \frac{\partial (\Delta \rho \hat{u}^2)}{\partial x} = -A \Delta \rho g \frac{dz}{dx} - \frac{\partial \Delta p}{\partial x} + \frac{4}{3} \frac{\partial A}{\partial x} \left( \frac{\partial \hat{u}}{\partial x} \right)^2 + F_F
\]

**Energy Equation:** \[
\frac{\partial \Delta \hat{e}}{\partial t} + \frac{\partial \Delta \hat{u} \hat{e}}{\partial x} = -\frac{\partial \Delta \hat{p}}{\partial x} + \frac{\partial A_{\Delta \hat{q} close}}{\partial x} + \hat{Q} + \Phi
\]

In the next section we derive the enthalpy form of the energy equation.

### 3.4 The Enthalpy Equation

In this section, we derive the enthalpy form of the energy equation known as the enthalpy equation. From this point forward, we assume that all residuals are approximately zero. We begin by writing the total energy per unit mass as

\[
\hat{e} = \hat{E} + \frac{1}{2} \hat{u}^2 + X
\]  

(3.56)

\(X\) is the gravitational potential energy per unit mass and \(\hat{E}\) is the Favre-filtered internal energy per unit mass. We can now re-write the energy equation using the expanded internal unit energy. This is given by

\[
\frac{\partial \Delta \rho \hat{E}}{\partial t} + \frac{1}{2} \frac{\partial \Delta \rho \hat{u}^2}{\partial t} + \frac{\partial \Delta \rho \hat{u} \hat{E}}{\partial x} + \frac{1}{2} \frac{\partial \Delta \rho \hat{u}^3}{\partial x} + \frac{\partial \Delta \rho \hat{u}X}{\partial x} = \frac{\partial A \hat{u} \hat{p}}{\partial x} + \frac{\partial A \hat{q}}{\partial x} + \hat{Q} + \Phi
\]

(3.57)
3.4. The Enthalpy Equation

Re-arranging terms on the left hand side of the energy equation yields

\[ \frac{\partial \tilde{A} \rho \tilde{E}}{\partial t} + \frac{\partial \tilde{A} \rho \tilde{u} \tilde{E}}{\partial x} + 1 \frac{\partial \tilde{A} \rho \tilde{u}^2}{\partial t} + 1 \frac{\partial \tilde{A} \rho \tilde{u}^3}{\partial x} + \frac{\partial \tilde{A} \rho X}{\partial t} + \frac{\partial \tilde{A} \rho \tilde{u} X}{\partial x} = \]

\[ - \frac{\partial A \tilde{u} p}{\partial x} + \frac{\partial A \tilde{q}}{\partial x} + \dot{Q} + \dot{\Phi} \]  

(3.58)

We begin manipulating the left hand side of the energy equation. We use the chain rule on high order velocity and body force terms to expand the enthalpy equation. The expanded enthalpy equation is

\[ \frac{\partial \tilde{A} \rho \tilde{E}}{\partial t} + \frac{\partial \tilde{A} \rho \tilde{u} \tilde{E}}{\partial x} + 1 \frac{\partial \tilde{A} \rho \tilde{u}^2}{\partial t} + 1 \frac{\partial \tilde{A} \rho \tilde{u}^3}{\partial x} + \frac{\partial \tilde{A} \rho X}{\partial t} + \frac{\partial \tilde{A} \rho \tilde{u} X}{\partial x} = \]

\[ \frac{\partial \tilde{A} \rho \tilde{E}}{\partial t} + \frac{\partial \tilde{A} \rho \tilde{u} \tilde{E}}{\partial x} + \tilde{u} \frac{\partial \tilde{A} \rho \tilde{u}}{\partial t} + A \rho \tilde{u} \frac{\partial \tilde{u}}{\partial t} + \tilde{u} \frac{\partial A \tilde{p} \tilde{u}^2}{\partial x} + A \rho \tilde{u}^2 \frac{\partial \tilde{u}}{\partial x} 
\]

\[ + X \left( \frac{\partial \tilde{A} \rho}{\partial t} + \frac{\partial A \rho \tilde{u}}{\partial x} \right) + A \rho \frac{\partial X}{\partial t} + A \rho \frac{\partial X}{\partial x} \]  

(3.59)

Now we apply the continuity equation where applicable, assume that gravity is the only body force and that it is time independent. This simplifies the left hand side of the energy equation down to

\[ \frac{\partial \tilde{A} \rho \tilde{E}}{\partial t} + \frac{\partial \tilde{A} \rho \tilde{u} \tilde{E}}{\partial x} + \tilde{u} \left( \frac{\partial \tilde{A} \rho \tilde{u}}{\partial t} + \frac{\partial A \rho \tilde{u}^2}{\partial x} \right) + \frac{A \rho \tilde{u}^2 \tilde{u}}{\partial t} + \frac{A \rho \tilde{u}^2 \tilde{u}}{\partial x} + A \rho \frac{\partial X}{\partial x} \]  

(3.60)

The reverse chain rule allows us to expand the temporal and spatial derivatives of the velocity components as

\[ \tilde{A} \rho \frac{\partial \tilde{u}}{\partial t} = \frac{\partial A \rho \tilde{u}}{\partial t} - \tilde{u} \frac{\partial A \rho}{\partial t} \]

\[ \tilde{A} \rho \frac{\partial \tilde{u}}{\partial x} = \frac{\partial A \rho \tilde{u}^2}{\partial x} - \tilde{u} \frac{\partial A \rho \tilde{u}}{\partial x} \]

(3.61)

(3.62)

The expanded form of the 1D energy is

\[ \frac{\partial \tilde{A} \rho \tilde{E}}{\partial t} + \frac{\partial \tilde{A} \rho \tilde{u} \tilde{E}}{\partial x} + \tilde{u} \left( \frac{\partial \tilde{A} \rho \tilde{u}}{\partial t} + \frac{\partial A \rho \tilde{u}^2}{\partial x} \right) + \frac{\tilde{u} \partial A \rho \tilde{u}}{\partial t} \]

\[ - \frac{\tilde{u} \partial A \rho}{\partial t} + \frac{\tilde{u} \partial A \rho \tilde{u}^2}{\partial x} - \frac{\tilde{u} \partial A \rho \tilde{u}}{\partial x} + \frac{\partial A \rho \tilde{u} \tilde{u}}{\partial x} \]  

(3.63)
Chapter 3. Derivation of the 1D Navier Stokes Equations

Re-arranging this equation yields

\[
\frac{\partial A\rho\dot{E}}{\partial t} + \frac{\partial A\rho\dot{u}}{\partial x} + \frac{\hat{u}}{2} \left( \frac{\partial A\rho\dot{u}}{\partial t} + \frac{\partial A\rho\dot{u}^2}{\partial x} \right) \\
+ \frac{\hat{u}}{2} \left( \frac{\partial A\rho\dot{u}}{\partial t} + \frac{\partial A\rho\dot{u}^2}{\partial x} - \hat{u} \left( \frac{\partial A\rho}{\partial t} + \frac{\partial A\rho\dot{u}}{\partial x} \right) \right) + A\rho\dot{u}\frac{\partial X}{\partial x}
\]  
(3.64)

Since we assume that the only unit body force affecting the potential energy of the fluid is gravity we represent \( X \) by

\[ X = gz \]  
(3.65)

The left hand side of the energy equation becomes

\[
\frac{\partial A\rho\dot{E}}{\partial t} + \frac{\partial A\rho\dot{u}}{\partial x} + \frac{\hat{u}}{2} \left( \frac{\partial A\rho\dot{u}}{\partial t} + \frac{\partial A\rho\dot{u}^2}{\partial x} \right) \\
+ \frac{\hat{u}}{2} \left( \frac{\partial A\rho\dot{u}}{\partial t} + \frac{\partial A\rho\dot{u}^2}{\partial x} - \hat{u} \left( \frac{\partial A\rho}{\partial t} + \frac{\partial A\rho\dot{u}}{\partial x} \right) \right) + A\rho\dot{u}\frac{dz}{dx}
\]  
(3.66)

The continuity equation can be used to simplify the energy equation. The simplified energy equation is

\[
\frac{\partial A\rho\dot{E}}{\partial t} + \frac{\partial A\rho\dot{u}}{\partial x} + \frac{\hat{u}}{2} \left( \frac{\partial A\rho\dot{u}}{\partial t} + \frac{\partial A\rho\dot{u}^2}{\partial x} \right) + A\rho\dot{u}\frac{dz}{dx} = \\
- \frac{\partial A\rho\dot{p}}{\partial x} + \frac{\partial A\dot{q}}{\partial x} + \dot{Q} + \dot{\Phi}
\]  
(3.67)

Next, we multiply the momentum equation by the Favre-Filtered velocity to obtain the 1D kinetic energy equation. The 1D kinetic energy equation is given by

\[
\hat{u} \left( \frac{\partial (A\rho\dot{u})}{\partial t} + \frac{\partial (A\rho\dot{u}^2)}{\partial x} \right) = -A\hat{u}\rho\dot{g}\frac{dz}{dx} - \hat{u} \frac{\partial A\rho}{\partial x} + \frac{A\rho}{3} \frac{\partial}{\partial x} \left( \hat{\mu} \frac{\partial \hat{u}}{\partial x} \right) + \hat{u} F_F
\]  
(3.68)

Through algebraic manipulation, the 1D kinetic energy equation becomes

\[
\hat{u} \left( \frac{\partial (A\rho\dot{u})}{\partial t} + \frac{\partial (A\rho\dot{u}^2)}{\partial x} \right) = -A\hat{u}\rho\dot{g}\frac{dz}{dx} - \hat{u} \frac{\partial A\rho}{\partial x} + \frac{A\rho}{3} \frac{\partial}{\partial x} \left( \hat{\mu} \frac{\partial \hat{u}}{\partial x} \right) + \hat{u} F_F
\]  
(3.69)
3.5. Summary of Equations and Model Closure

We now subtract the kinetic energy equation from the energy equation to obtain

\[
\frac{\partial A\rho\hat{E}}{\partial t} + \frac{\partial A\rho\hat{u}\hat{E}}{\partial x} + \hat{u}\left(\frac{\partial A\rho\hat{u}}{\partial t} + \frac{\partial A\rho\hat{u}^2}{\partial x}\right) - \hat{u}\left(\frac{\partial A\rho\hat{u}}{\partial t} + \frac{\partial A\rho\hat{u}^2}{\partial x}\right) + A\rho g\frac{dz}{dx} = \frac{\partial A\rho\hat{p}}{\partial x} + A\rho g\frac{dz}{dx} + \frac{\partial A\rho\hat{q}}{\partial x} - \frac{\partial A\rho\hat{q}}{3\partial x} A\left(\rho \frac{\partial \hat{u}}{\partial x}\right) - \hat{u}F_F
\]

(3.70)

Canceling any immediate terms results in

\[
\frac{\partial A\rho\hat{E}}{\partial t} + \frac{\partial A\rho\hat{u}\hat{E}}{\partial x} = -\frac{\partial A\rho\hat{p}}{\partial x} + \frac{\partial A\rho\hat{q}}{\partial x} + \hat{Q} + \hat{\Phi} + A\rho g\frac{dz}{dx} + \hat{u}\frac{\partial A\rho\hat{p}}{\partial x} - \frac{\partial A\rho\hat{q}}{3\partial x} A\left(\rho \frac{\partial \hat{u}}{\partial x}\right) - \hat{u}F_F
\]

(3.71)

Additional algebraic manipulation yields

\[
\frac{\partial A\rho\hat{E}}{\partial t} + \frac{\partial A\rho\hat{u}(\hat{E} + \frac{\hat{p}}{\rho})}{\partial x} = \frac{\partial A\rho\hat{q}}{\partial x} + \hat{Q} + \hat{\Phi} + \hat{u}\frac{\partial A\rho\hat{p}}{\partial x} - \frac{\partial A\rho\hat{q}}{3\partial x} A\left(\rho \frac{\partial \hat{u}}{\partial x}\right) - \hat{u}F_F
\]

(3.72)

Specific enthalpy is defined as

\[
\hat{h} = \hat{E} + \frac{\hat{p}}{\rho}
\]

(3.73)

Substituting this definition into the energy equation results in the 1D enthalpy equation given by

\[
\frac{\partial A\rho\hat{E}}{\partial t} + \frac{\partial A\rho\hat{u}\hat{h}}{\partial x} = \frac{\partial A\rho\hat{q}}{\partial x} + \hat{\Phi} + \hat{u}\frac{\partial A\rho\hat{p}}{\partial x} - \frac{\partial A\rho\hat{q}}{3\partial x} A\left(\rho \frac{\partial \hat{u}}{\partial x}\right) - \hat{u}F_F
\]

(3.74)

In the next section, we summarize the continuity, momentum and enthalpy equations without residual terms.

3.5 Summary of Equations and Model Closure

In this section, we summarize the continuity, momentum, and enthalpy equations without residual terms. Information on each equation’s residual terms can be found in their respective sections. Additionally, for a generic quantity \( \phi \), we denote its cross-sectional area average
by an over-line and a density average by a hat. Notationally these are expressed as

$$\bar{\phi} = \int_A \phi \, dx$$  \hspace{1cm} (3.75)

and

$$\hat{\phi} = \frac{\bar{\phi}}{\bar{\rho}} = \frac{\int_A \phi \, dx}{\int_A \rho \, dx}$$  \hspace{1cm} (3.76)

respectively. The primitive variables of the system are pressure, enthalpy, and velocity. Enthalpy and pressure are chosen to be the state variables of the system. The continuity, momentum, and enthalpy equations are listed below:

Continuity Equation:

$$\frac{\partial A\bar{p}}{\partial t} + \frac{\partial A\bar{p}\hat{u}}{\partial x} = 0$$  \hspace{1cm} (3.77)

Momentum Equation:

$$\frac{\partial (A\bar{p}\hat{u})}{\partial t} + \frac{\partial (A\bar{p}\hat{u}^2)}{\partial x} = -A\bar{p}g \frac{dz}{dx} - \frac{\partial A\bar{p}}{\partial x} + \frac{4}{3} \frac{\partial}{\partial x} A \left( \hat{n}u \frac{\partial \hat{u}}{\partial x} \right) + F_F$$  \hspace{1cm} (3.78)

Enthalpy Equation:

$$\frac{\partial A\bar{p}\hat{E}}{\partial t} + \frac{\partial A\bar{p}\hat{h}}{\partial x} = \hat{u} \frac{\partial A\bar{p}}{\partial x} - \hat{u} \frac{4}{3} \frac{\partial}{\partial x} \left( A\bar{p}u \frac{\partial \hat{u}}{\partial x} \right) + \frac{\partial A\hat{q}}{\partial x} + \hat{Q} + \hat{\Phi} - \hat{u}F_F$$  \hspace{1cm} (3.79)

In addition to the Navier Stokes equations, we also include temperature and density correlations, $\pi(\cdot, \cdot)$ and $\omega(\cdot, \cdot)$ respectively, to close the model. These additional correlations are given by:

Temperature Correlation: $\hat{T} = \pi(p, \hat{h})$  \hspace{1cm} (3.80)

Density Correlation: $\bar{\rho} = \omega(p, \hat{h})$

To finish closing the model, we choose the following boundary conditions:

$$p(t, 0) = p_0; \quad u(t, 0) = u_0(t); \quad \hat{h}(t, 0) = \hat{h}_0(t); \quad \mu \frac{\partial \hat{h}(t, L)}{\partial x} = 0$$  \hspace{1cm} (3.81)

and the following initial conditions:

$$\hat{u}(0, x) = \hat{u}_{init}(x); \quad \hat{h}(0, x) = h_{init}(x); \quad p(0, x) = p_{init}(x)$$  \hspace{1cm} (3.82)
3.5. Summary of Equations and Model Closure

Meanings for all symbols used in this summary can be found in nomenclature section of this thesis. In the next chapter, we discuss the linear discontinuous Galerkin semi-discretization of these equations.
Chapter 4

Discontinuous Galerkin Spatial
Semi-Discretization

In this chapter, we spatially discretize the 1D Navier Stokes equations derived in Chapter 3 using a linear discontinuous Galerkin method. We leave temporal derivatives in their exact form, allowing for a wide variety of temporal discretization schemes. This is called a semi-discrete form. In section 4.1 we introduce the 1D Navier Stokes equation plus necessary closure terms. Section 4.2 defines the notation and basis functions used in the DG semi-discretization. We then perform a semi-discretization on the continuity and enthalpy equations in sections 4.3 and 4.4 respectively. We summarize the semi-discretization in section 4.5. Finally, in section 4.6 we derive the steady state version of equations from the transient equations.

4.1 Introduction

In Chapter 3 we reduced the 3D Navier Stokes equations to 1D. We list the 1D equations below. A hat over a quantity indicates that it is density averaged while a bar indicates that it is area averaged. Meanings of all terms used in this discretization can be found in the nomenclature section of this thesis. The resulting system of equations are given by

\[
\text{Continuity Equation: } \frac{\partial A\bar{\rho}}{\partial t} + \frac{\partial A\bar{\rho}\bar{u}}{\partial x} = 0
\]  

(4.1)
4.2. Notation and Background For Semi-Discretization

Momentum Equation: \[
\frac{\partial (Ap\hat{u})}{\partial t} + \frac{\partial (Ap\hat{u}^2)}{\partial x} = -Ap\rho g \frac{dz}{dx} - \frac{\partial Ap}{\partial x} + \frac{4}{3} \frac{\partial}{\partial x} \left( \hat{u} \frac{\partial \hat{u}}{\partial x} \right) + F_f \tag{4.2}
\]

Enthalpy Equation: \[
\frac{\partial Ap\hat{U}}{\partial t} + \frac{\partial Ap\hat{u}h}{\partial x} = \hat{u} \frac{\partial Ap}{\partial x} - \hat{u} \frac{4}{3} \frac{\partial}{\partial x} \left( \hat{u} \frac{\partial \hat{u}}{\partial x} \right) + \frac{\partial A\hat{q}}{\partial x} + \hat{Q} + \hat{\Phi} - \hat{u} F_f \tag{4.3}
\]

For the spatial discretization we will assume a known pressure profile. This allows us to replace the momentum equation by this known pressure profile in time and space. The resulting partial differential equations that we discretize in space are the

Continuity Equation: \[
\frac{\partial Ap}{\partial t} + \frac{\partial Ap\hat{u}}{\partial x} = 0 \tag{4.4}
\]

and the

Energy Equation: \[
\frac{\partial Ap\hat{U}}{\partial t} + \frac{\partial Ap\hat{u}h}{\partial x} = \hat{u} \frac{\partial Ap}{\partial x} - \hat{u} \frac{4}{3} \frac{\partial}{\partial x} \left( \hat{u} \frac{\partial \hat{u}}{\partial x} \right) + \frac{\partial A\hat{q}}{\partial x} + \hat{Q} + \hat{\Phi} - \hat{u} F_f \tag{4.5}
\]

We denote the known pressure profile by

Pressure Profile: \( p(t, x) = \alpha(t, x) \) \( \tag{4.6} \)

The temperature and density correlations we use to close the model are listed below:

Temperature Correlation: \( \hat{T} = \pi(p, \hat{h}) \) \( \tag{4.7} \)

Density Correlation: \( \hat{\rho} = \omega(p, \hat{h}) \)

After this introduction, we drop the bars and hats for simplicity and clarity. In the next section, we introduce the notation and background used in the linear DG semi-discretization.

### 4.2 Notation and Background For Semi-Discretization

In this section we will briefly describe the notation used in the semi-discretization. We begin by discussing the mesh. Figure 4.1 illustrates our notation for the mesh.
We break the mesh into $N$ elements denoted by $e_i$, $i = 1, 2, ..., N$. Elements are not required to be uniform. This also means that the mesh contains $N+1$ nodes denoted by $x_j$, $j = 0, 1, ..., N$. We define $x_0$ to be the left boundary of the domain and $x_N$ to be the right boundary of the domain. Each element $i$ has a length denoted $\Delta x_i = x_i - x_{i-1}$ where $i = 1, 2, ..., N$.

In the discontinuous Galerkin method, we represent the approximate solution by a piece-wise linear polynomial. This piece-wise linear approximation is the linear combination of linear basis functions $\phi_{i,0}$ and $\phi_{i,1}$ where $i = 1, 2, ...N$. We define these linear basis functions as:

$$
\phi_{i,0}(x) = \begin{cases} 
\frac{x_{i+1} - x}{x_{i+1} - x_i} & x \in [x_i, x_{i+1}] \\
0 & \text{otherwise}
\end{cases}
$$

$$
\phi_{i,1}(x) = \begin{cases} 
\frac{x - x_i}{x_{i+1} - x_i} & x \in [x_i, x_{i+1}] \\
0 & \text{otherwise}
\end{cases}
$$

Next, we assume that for an arbitrary quantity $\phi(t,x)$, its piece-wise linear approximation $\phi_h(t,x)$ can be represented as:

$$
\psi(t,x) \approx \psi_h(t,x) = \sum_{i=1}^{N} \psi_{i,0}(t)\phi_{i,0}(x) + \psi_{i,1}(t)\phi_{i,1}(x) \quad (4.8)
$$

Figure 4.2 illustrates how the approximate solution on element $i = 1, 2, ...N$ is constructed.
4.2. Notation and Background For Semi-Discretization

from linear basis functions. Each element $i = 1, 2, \ldots, N$ has exactly two non-zero basis functions $\phi_{i,0}(x)$ and $\phi_{i,1}(x)$ denoted by solid gold lines. The basis function that is one on the left, $\phi_{i,0}(x)$, is the left basis function and the basis function that is one on the right, $\phi_{i,1}(x)$, is the right basis function.

![Diagram illustrating how a global approximate solution is constructed from a linear basis functions.](image)

We scale these basis functions by the time dependent coefficients $\psi_{i,0}$ and $\psi_{i,1}$, represented by dashed blue lines. The solid orange line is the approximate solution found by scaling the basis functions by the time dependent coefficients and summing. Below, we list the linear approximations of all quantities:

\begin{align*}
\rho(t, x) &\approx \rho_h(t, x) = \bigoplus_{i=1}^{N} \rho_{i,0}(t)\phi_{i,0}(x) + \rho_{i,1}(t)\phi_{i,1}(x) \quad x \in [x_{i-1}, x_i] \quad (4.9) \\
h(t, x) &\approx h_h(t, x) = \bigoplus_{i=1}^{N} h_{i,0}(t)\phi_{i,0}(x) + h_{i,1}(t)\phi_{i,1}(x) \quad x \in [x_{i-1}, x_i] \quad (4.10) \\
u(t, x) &\approx u_h(t, x) = \bigoplus_{i=1}^{N} u_{i,0}(t)\phi_{i,0}(x) + u_{i,1}(t)\phi_{i,1}(x) \quad x \in [x_{i-1}, x_i] \quad (4.11) \\
p(t, x) &\approx p_h(t, x) = \bigoplus_{i=1}^{N} p_{i,0}(t)\phi_{i,0}(x) + p_{i,1}(t)\phi_{i,1}(x) \quad x \in [x_{i-1}, x_i] \quad (4.12) \\
T(t, x) &\approx T_h(t, x) = \bigoplus_{i=1}^{N} T_{i,0}(t)\phi_{i,0}(x) + T_{i,1}(t)\phi_{i,1}(x) \quad x \in [x_{i-1}, x_i] \quad (4.13)
\end{align*}
Chapter 4. Discontinuous Galerkin Spatial Semi-Discretization

For conciseness, we will omit function handles in this discretization. The reader should refer back to this section to determine whether a quantity is a function of space, time, or both. In the next section, we begin the spatial discretization of the continuity equation using the linear discontinuous Galerkin method described in this section.

4.3 The Continuity Equation

We implement a discontinuous Galerkin method using linear test and basis functions for the spatial discretization of the continuity equation. The first step in DGM is to derive the weak form of the continuity equation. We start by multiplying the continuity equation by the function \( \nu(x) \in H^1 \). The function handle for \( \nu \) is omitted from this point forward. The resulting equation is

\[
\frac{\partial \rho}{\partial t} \nu + \frac{\partial \rho u}{\partial x} \nu = 0 \tag{4.14}
\]

Next, we integrate across element \( e_i = [x_{i-1}, x_i] \) where \( i = 1, 2, \ldots, N \).

\[
\int_{x_{i-1}}^{x_i} \frac{\partial \rho}{\partial t} \nu \, dx + \int_{x_{i-1}}^{x_i} \frac{\partial \rho u}{\partial x} \nu \, dx \tag{4.15}
\]

We then integrate the spatial derivative by parts to move the derivative to the test function \( \nu \) to yield

\[
\int_{x_{i-1}}^{x_i} \frac{\partial \rho}{\partial t} \nu \, dx + [\rho u \nu]_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} \rho u \frac{\partial \nu}{\partial x} \, dx = 0 \tag{4.16}
\]

The arbitrary test function is replaced by one of the non-zero basis functions \( \phi_{i,j} \) that is non-zero on element \( i = 1, 2, \ldots, N \) where \( j = 0, 1 \). Thus, the continuity equation becomes

\[
\int_{x_{i-1}}^{x_i} \frac{\partial \rho}{\partial t} \phi_{i,j} \, dx + [\rho u \phi_{i,j}]_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} \rho u \frac{\partial \phi_{i,j}}{\partial x} \, dx = 0 \tag{4.17}
\]
4.3. The Continuity Equation

Next, we replace all quantities with their DG approximations and omit any basis functions that are strictly zero on element $i = 1, 2, \ldots N$ to produce

\[
\int_{x_{i-1}}^{x_i} \frac{\partial (\rho_{i,0} \phi_{i,0} + \rho_{i,1} \phi_{i,1})}{\partial t} \phi_{i,j} \, dx + [\rho u \phi_{i,j}] \bigg|_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} (\rho_{i,0} \phi_{i,0} + \rho_{i,1} \phi_{i,1})(u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1}) \frac{\partial \phi_{i,j}}{\partial x} \, dx = 0
\]

(Since we assume that $\rho_{i,j} i = 1, 2, \ldots N$ and $j = 0, 1$ is strictly a function of time and all the basis function $\phi_{i,j} i = 1, 2, \ldots N$ and $j = 0, 1$ is strictly a function of space, we can evaluate the temporal derivative as

\[
\int_{x_{i-1}}^{x_i} \frac{\partial \rho_{i,0}}{\partial t} \phi_{i,0} + \frac{\partial \rho_{i,1}}{\partial t} \phi_{i,1} \phi_{i,j} \, dx + [\rho u \phi_{i,j}] \bigg|_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} (\rho_{i,0} \phi_{i,0} + \rho_{i,1} \phi_{i,1})(u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1}) \frac{\partial \phi_{i,j}}{\partial x} \, dx = 0
\]

We will now evaluate the test functions at $j=0$ and $j=1$ to obtain a solvable system of ordinary differential equations equations.

**Left Basis Function ($j=0$)**

Assuming that the test function is the left basis function ($j=0$) yields

\[
\int_{x_{i-1}}^{x_i} \frac{\partial \rho_{i,0}}{\partial t} \phi_{i,0} + \frac{\partial \rho_{i,1}}{\partial t} \phi_{i,1} \phi_{i,0} \, dx + [\rho u \phi_{i,0}] \bigg|_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} (\rho_{i,0} \phi_{i,0} + \rho_{i,1} \phi_{i,1})(u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1}) \frac{\partial \phi_{i,0}}{\partial x} \, dx = 0
\]

Distributing and collecting terms where possible produces

\[
\int_{x_{i-1}}^{x_i} \frac{\partial \rho_{i,0}}{\partial t} \phi_{i,0}^2 + \frac{\partial \rho_{i,1}}{\partial t} \phi_{i,1} \phi_{i,0} \, dx + [\rho u \phi_{i,0}] \bigg|_{x_{i-1}}^{x_i} + \frac{1}{\Delta x_i} \int_{x_{i-1}}^{x_i} \rho_{i,0} u_{i,0} \phi_{i,0}^2 + \rho_{i,0} u_{i,1} \phi_{i,0} \phi_{i,1} + \rho_{i,1} u_{i,0} \phi_{i,0} \phi_{i,1} + \rho_{i,1} u_{i,1} \phi_{i,1}^2 \, dx = 0
\]

We evaluate all the integrals to obtain

\[
\frac{\Delta x_i}{3} \frac{\partial \rho_{i,0}}{\partial t} + \frac{\Delta x_i}{6} \frac{\partial \rho_{i,1}}{\partial t} + [\rho u \phi_{i,0}] \bigg|_{x_{i-1}}^{x_i} + \frac{\rho_{i,0} u_{i,0}}{3} + \frac{\rho_{i,0} u_{i,1}}{6} + \frac{\rho_{i,1} u_{i,0}}{6} + \frac{\rho_{i,1} u_{i,1}}{3} = 0
\]
All that is left now is to evaluate the flux between elements. Since flow will only go in one direction, we will implement an upwind scheme. This means that at the boundary of two elements, the function takes on the value on of element on the left boundary (assuming the flow is to the right). The equation for numerical flux is

\[ \left[ \rho u \phi_i \right]_{x_i=1}^{x_i} = \rho(x_i)u(x_i)\phi_i,0(x_i) - \rho(x_{i-1})u(x_{i-1})\phi_i,0(x_{i-1}) = -\rho(x_{i-1})u(x_{i-1}) \]

\[ = -\rho_{i-1,1}u_{i-1,1} \]  

(4.23)

Now combining everything, we obtain the following expression for the left basis function (j=0):

\[ \frac{\Delta x_i}{3} \frac{\partial \rho_{i,0}}{\partial t} + \frac{\Delta x_i}{6} \frac{\partial \rho_{i,1}}{\partial t} - \rho_{i-1,1}u_{i-1,1} + \frac{\rho_{i,0}u_{i,0}}{3} + \frac{\rho_{i,0}u_{i,1}}{6} + \frac{\rho_{i,1}u_{i,0}}{6} + \frac{\rho_{i,1}u_{i,1}}{3} = 0 \]  

(4.24)

**The Right Basis Function (j=1)**

We perform the same steps but assume that the test function is the right basis function (j=1) on element \( i = 1, 2, \ldots, N \). This yields the equation

\[ \int_{x_{i-1}}^{x_i} \left( \frac{\partial \rho_{i,0}}{\partial t} \phi_i,0 + \frac{\partial \rho_{i,1}}{\partial t} \phi_i,1 \right) dx + \left[ \rho u \phi_i \right]_{x_i}^{x_{i-1}} \]

\[ - \int_{x_{i-1}}^{x_i} \left( \rho_{i,0}\phi_i,0 + \rho_{i,1}\phi_i,1 \right) \left( u_{i,0}\phi_i,0 + u_{i,1}\phi_i,1 \right) \frac{\partial \phi_i,1}{\partial x} dx = 0 \]  

(4.25)

Expanding terms and simplifying where possible produces

\[ \int_{x_{i-1}}^{x_i} \frac{\partial \rho_{i,0}}{\partial t} \phi_i,0 + \frac{\partial \rho_{i,1}}{\partial t} \phi_i,1 dx + \left[ \rho u \phi_i \right]_{x_i}^{x_{i-1}} \]

\[ - \frac{1}{\Delta x_i} \int_{x_{i-1}}^{x_i} \rho_{i,0}u_{i,0}\phi_i,0^2 + \rho_{i,0}u_{i,1}\phi_i,0\phi_i,1 + \rho_{i,1}u_{i,0}\phi_i,0 + \rho_{i,1}u_{i,1}\phi_i,1^2 dx = 0 \]  

(4.26)

Evaluating the integrals yields

\[ \frac{\Delta x_i}{6} \frac{\partial \rho_{i,0}}{\partial t} + \frac{\Delta x_i}{3} \frac{\partial \rho_{i,1}}{\partial t} + \left[ \rho u \phi_i \right]_{x_i}^{x_{i-1}} - \frac{\rho_{i,0}u_{i,0}}{3} - \frac{\rho_{i,0}u_{i,1}}{6} - \frac{\rho_{i,1}u_{i,0}}{6} - \frac{\rho_{i,1}u_{i,1}}{3} = 0 \]  

(4.27)
4.4. The Enthalpy Equation

All that is left now is to evaluate the flux between elements. Since flow is only in one direction, we implement an upwind scheme. The flux is evaluated as

\[ [\rho u \phi_i]_{x_{i-1}}^{x_i} = \rho(x_i)u(x_i)\phi_{i,1}(x_i) - \rho(x_{i-1})u(x_{i-1})\phi_{i,1}(x_{i-1}) = \rho(x_i)u(x_i) \]

\[ = \rho_{i,1}u_{i,1} \] (4.28)

The resulting equation with numerical flux is

\[ \frac{\Delta x_i}{6} \frac{\partial \rho_{i,0}}{\partial t} + \frac{\Delta x_i}{3} \frac{\partial \rho_{i,1}}{\partial t} + \frac{2\rho_{i,1}u_{i,1}}{3} - \frac{\rho_{i,0}u_{i,0}}{3} - \frac{\rho_{i,0}u_{i,1}}{6} - \frac{\rho_{i,1}u_{i,0}}{6} = 0 \] (4.29)

**Summary of Continuity Equation Discretization**

We have performed a semi-discretization of the continuity equation using a discontinuous Galerkin method with linear basis and test functions. Additionally, we have chosen to use an upwind scheme to deal with flux between elements since flow should only be in one direction.

The two equations for j=0 and j=1 respectfully that discretize the continuity equation on element \( i = 1, 2, \ldots, N \) of the mesh are:

\[ \frac{\Delta x_i}{3} \frac{\partial \rho_{i,0}}{\partial t} + \frac{\Delta x_i}{6} \frac{\partial \rho_{i,1}}{\partial t} - \frac{\rho_{i-1,1}u_{i-1,1}}{3} + \frac{\rho_{i,0}u_{i,0}}{3} + \frac{\rho_{i,0}u_{i,1}}{6} + \frac{\rho_{i,1}u_{i,0}}{6} + \frac{\rho_{i,1}u_{i,1}}{3} = 0 \] (4.30)

\[ \frac{\Delta x_i}{6} \frac{\partial \rho_{i,0}}{\partial t} + \frac{\Delta x_i}{3} \frac{\partial \rho_{i,1}}{\partial t} + \frac{2\rho_{i,1}u_{i,1}}{3} - \frac{\rho_{i,0}u_{i,0}}{3} - \frac{\rho_{i,0}u_{i,1}}{6} - \frac{\rho_{i,1}u_{i,0}}{6} = 0 \] (4.31)

In the next section, we will discretize the enthalpy equation.

### 4.4 The Enthalpy Equation

We implement a discontinuous Galerkin Method using linear test and basis functions for the spatial discretization. The first step in DGM is to derive the weak form of the enthalpy equation. To begin, we multiply the entire equation by a test function \( \nu(x) \) that lives in \( H^1 = \{ \nu(x) \text{ and } \nu'(x) \text{ are square integrable} \} \). We omit the function handle from \( \nu(x) \) for
The resulting enthalpy equation is

\[ A \frac{\partial (\rho E)}{\partial t} \nu + A \frac{\partial p u h}{\partial x} \nu = u A \frac{\partial p}{\partial x} \nu - u A \frac{4}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) \nu + A \frac{\partial q}{\partial x} \nu + \dot{Q} \nu + \Phi \nu - u F_F \nu \]  

(4.32)

Integrating across an element \( e_i = [x_{i-1}, x_i] \) \( i = 1, 2, ..., N \) yields

\[
\int_{x_{i-1}}^{x_i} A \frac{\partial (\rho E)}{\partial t} \nu \, dx + \int_{x_{i-1}}^{x_i} A \frac{\partial p u h}{\partial x} \nu \, dx = \int_{x_{i-1}}^{x_i} u A \frac{\partial p}{\partial x} \nu \, dx - \int_{x_{i-1}}^{x_i} u A \frac{4}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) \nu \, dx + \int_{x_{i-1}}^{x_i} A \frac{\partial q}{\partial x} \nu \, dx + \int_{x_{i-1}}^{x_i} \dot{Q} \nu \, dx + \int_{x_{i-1}}^{x_i} \Phi \nu \, dx - \int_{x_{i-1}}^{x_i} u F_F \nu \, dx 
\]

(4.33)

We represent the heat flux in terms of temperature by

\[ \frac{\partial q}{\partial x} = \frac{\partial}{\partial x} \left( \kappa \frac{\partial T}{\partial x} \right) \]

(4.34)

The enthalpy equation becomes

\[
\int_{x_{i-1}}^{x_i} A \frac{\partial (\rho E)}{\partial t} \nu \, dx + \int_{x_{i-1}}^{x_i} A \frac{\partial p u h}{\partial x} \nu \, dx = \int_{x_{i-1}}^{x_i} u A \frac{\partial p}{\partial x} \nu \, dx - \int_{x_{i-1}}^{x_i} u A \frac{4}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) \nu \, dx + \int_{x_{i-1}}^{x_i} A \frac{\partial}{\partial x} \left( \kappa \frac{\partial T}{\partial x} \right) \nu \, dx + \int_{x_{i-1}}^{x_i} \dot{Q} \nu \, dx + \int_{x_{i-1}}^{x_i} \Phi \nu \, dx - \int_{x_{i-1}}^{x_i} u F_F \nu \, dx 
\]

(4.35)

Due to the length of this equation, we will deal with each term separately for clarity. We label each term in the enthalpy equation Table 4.1 below.
4.4. The Enthalpy Equation

<table>
<thead>
<tr>
<th>Term</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ \int_{x_{i-1}}^{x_i} A \frac{\partial E}{\partial t} \nu , dx ]</td>
<td>Temporal Term</td>
</tr>
<tr>
<td>[ \int_{x_{i-1}}^{x_i} A \frac{\partial u h}{\partial x} \nu , dx ]</td>
<td>Enthalpy Term</td>
</tr>
<tr>
<td>[ \int_{x_{i-1}}^{x_i} u A \frac{\partial p}{\partial x} \nu , dx ]</td>
<td>Pressure Term</td>
</tr>
<tr>
<td>[ - \int_{x_{i-1}}^{x_i} u A \frac{4}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) \nu , dx ]</td>
<td>Shear Term</td>
</tr>
<tr>
<td>[ \int_{x_{i-1}}^{x_i} A \frac{\partial}{\partial x} \left( \kappa \frac{\partial T}{\partial x} \right) \nu , dx ]</td>
<td>Heat Conductance Term</td>
</tr>
<tr>
<td>[ \int_{x_{i-1}}^{x_i} \dot{Q} \nu , dx ]</td>
<td>External Heat Transfer Term</td>
</tr>
<tr>
<td>[ \int_{x_{i-1}}^{x_i} \Phi \nu , dx ]</td>
<td>Viscous Dissipation Term</td>
</tr>
<tr>
<td>[ \int_{x_{i-1}}^{x_i} u F_F \nu , dx ]</td>
<td>Frictional Term</td>
</tr>
</tbody>
</table>

Table 4.1: Table labeling each term in the enthalpy equation.

The Temporal Term

We start by re-writing the temporal term as

\[ \int_{x_{i-1}}^{x_i} A \frac{\partial E}{\partial t} \nu \, dx \]  (4.36)

We can now replace the test function \( \nu \) with one of the basis functions defined on element \( i = 1, 2, \ldots, N \), \( \phi_{i,j} \), \( j = 0, 1 \) to produce

\[ \int_{x_{i-1}}^{x_i} A \frac{\partial E}{\partial t} \phi_{i,j} \, dx \]  (4.37)

Total internal energy can be represented in terms of enthalpy by

\[ E = h + \frac{p}{\rho} \]  (4.38)

Plugging this expanded form of \( E \) into the temporal term yields

\[ A \int_{x_{i-1}}^{x_i} \frac{\partial \rho(h + \frac{p}{\rho})}{\partial t} \phi_{i,j} \, dx \]  (4.39)
Chapter 4. Discontinuous Galerkin Spatial Semi-Discretization

We now distribute $\rho$ through the temporal derivative to simplify the expression

\[
A \int_{x_{i-1}}^{x_i} \frac{\partial (\rho h + p)}{\partial t} \phi_{i,j} \, dx
\]  

(4.40)

The product rule allows us to expand the temporal term as

\[
A \int_{x_{i-1}}^{x_i} h \frac{\partial \rho}{\partial t} \phi_{i,j} + \rho \frac{\partial h}{\partial t} \phi_{i,j} + \frac{\partial p}{\partial t} \phi_{i,j} \, dx
\]  

(4.41)

We now replace each quantity with its DG approximation or correlation. Additionally, we omit any uniformly zero basis functions on element $i = 1, 2, \ldots, N$. The resulting equation is

\[
A \int_{x_{i-1}}^{x_i} (h_{i,0} \phi_{i,0} + h_{i,1} \phi_{i,1}) \frac{\partial (\rho_{i,0} \phi_{i,0} + \rho_{i,1} \phi_{i,1})}{\partial t} \phi_{i,j} \, dx +
\]

\[
A \int_{x_{i-1}}^{x_i} (\rho_{i,0} \phi_{i,0} + \rho_{i,1} \phi_{i,1}) \frac{\partial (h_{i,0} \phi_{i,0} + h_{i,1} \phi_{i,1})}{\partial t} \phi_{i,j} \, dx +
\]

\[
A \int_{x_{i-1}}^{x_i} \frac{\partial \alpha(t, x)}{\partial t} \phi_{i,j} \, dx
\]  

(4.42)

Finally, we expand and combine all like terms to produce

\[
A \int_{x_{i-1}}^{x_i} h_{i,0} \frac{\partial \rho_{i,0}}{\partial t} \phi_{i,0}^2 \phi_{i,j} + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} \phi_{i,0} \phi_{i,1} \phi_{i,j} + h_{i,1} \frac{\partial \rho_{i,0}}{\partial t} \phi_{i,0} \phi_{i,1} \phi_{i,j} + h_{i,1} \frac{\partial \rho_{i,1}}{\partial t} \phi_{i,1}^2 \phi_{i,j} \, dx
\]

\[
A \int_{x_{i-1}}^{x_i} \rho_{i,0} \frac{\partial h_{i,0}}{\partial t} \phi_{i,0}^2 \phi_{i,j} + \rho_{i,0} \frac{\partial h_{i,1}}{\partial t} \phi_{i,0} \phi_{i,1} \phi_{i,j} + \rho_{i,1} \frac{\partial h_{i,0}}{\partial t} \phi_{i,0} \phi_{i,1} \phi_{i,j} + \rho_{i,1} \frac{\partial h_{i,1}}{\partial t} \phi_{i,1}^2 \phi_{i,j} \, dx +
\]

\[
A \int_{x_{i-1}}^{x_i} \frac{\partial \alpha(t, x)}{\partial t} \phi_{i,j} \, dx
\]  

(4.44)
4.4. The Enthalpy Equation

The Left Basis Function (j=0)

Now it is assumed that the test function $\nu$ is the left basis function ($j = 0$) which yields

$$
A \int_{x_{i-1}}^{x_i} \left( \frac{\partial \rho_{i,0}}{\partial t} \phi_{i,0}^3 + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} \phi_{i,0}^2 \phi_{i,1} + h_{i,1} \frac{\partial \rho_{i,0}}{\partial t} \phi_{i,0}^2 \phi_{i,1} + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} \phi_{i,0}^2 \phi_{i,1} \right) dx
$$

Evaluating integrals where possible simplifies the temporal term to

$$
\begin{align*}
A \int_{x_{i-1}}^{x_i} & \frac{\partial \rho_{i,0}}{\partial t} \phi_{i,0}^3 + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} \phi_{i,0}^2 \phi_{i,1} + h_{i,1} \frac{\partial \rho_{i,0}}{\partial t} \phi_{i,0}^2 \phi_{i,1} + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} \phi_{i,0}^2 \phi_{i,1} \, dx + \\
A \int_{x_{i-1}}^{x_i} & \frac{\partial \alpha(t,x)}{\partial t} \phi_{i,0} \, dx
\end{align*}
$$

Pulling out a $\frac{A \Delta x}{12}$ results in

$$
\begin{align*}
\frac{A \Delta x}{12} & \left( 3h_{i,0} \frac{\partial \rho_{i,0}}{\partial t} + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} + h_{i,1} \frac{\partial \rho_{i,0}}{\partial t} + \frac{h_{i,1}}{3} \frac{\partial \rho_{i,0}}{\partial t} + \frac{h_{i,0}}{\partial t} \frac{\partial \rho_{i,1}}{\partial t} + \frac{h_{i,0}}{\partial t} \frac{\partial \rho_{i,1}}{\partial t} \right) + A \int_{x_{i-1}}^{x_i} \frac{\partial \alpha(t,x)}{\partial t} \phi_{i,0} \, dx
\end{align*}
$$

(4.45)

Right Basis Function (j=1)

Now it is assumed that the test function $\nu$ is the right basis function ($j = 1$) which yields

$$
\begin{align*}
A \int_{x_{i-1}}^{x_i} & \left( \frac{\partial \rho_{i,0}}{\partial t} \phi_{i,0}^3 + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} \phi_{i,0}^2 \phi_{i,1} + h_{i,1} \frac{\partial \rho_{i,0}}{\partial t} \phi_{i,0}^2 \phi_{i,1} + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} \phi_{i,0}^2 \phi_{i,1} \right) dx \\
A \int_{x_{i-1}}^{x_i} & \frac{\partial \rho_{i,0}}{\partial t} \phi_{i,0}^3 + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} \phi_{i,0}^2 \phi_{i,1} + h_{i,1} \frac{\partial \rho_{i,0}}{\partial t} \phi_{i,0}^2 \phi_{i,1} + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} \phi_{i,0}^2 \phi_{i,1} \, dx + \\
A \int_{x_{i-1}}^{x_i} & \frac{\partial \alpha(t,x)}{\partial t} \phi_{i,1} \, dx
\end{align*}
$$

(4.48)
Chapter 4. Discontinuous Galerkin Spatial Semi-Discretization

Evaluating integrals where possible simplifies the temporal term to

\[
\begin{align*}
A & \Delta x_i \frac{\partial \rho_i,0}{\partial t} + A \Delta x_i \frac{\partial \rho_{i,1}}{\partial t} + A \Delta x_i \frac{\partial \rho_i,0}{\partial t} + A \Delta x_i \frac{\partial \rho_{i,1}}{\partial t} + A \Delta x_i \frac{\partial h_i,0}{\partial t} + A \Delta x_i \frac{\partial h_{i,1}}{\partial t} \\
& + \frac{1}{12} \rho_i,0 \frac{\partial h_i,0}{\partial t} + \frac{1}{12} \rho_i,1 \frac{\partial h_{i,1}}{\partial t} + \frac{3}{4} \rho_i,0 \frac{\partial h_i,0}{\partial t} + \frac{3}{4} \rho_i,1 \frac{\partial h_{i,1}}{\partial t} + A \int_{x_{i-1}}^{x_i} \frac{\partial \alpha(t,x)}{\partial t} \phi_{i,1} \, dx
\end{align*}
\]  

(4.49)

Pulling out a \(\frac{\Delta x_i}{12}\) results in

\[
\begin{align*}
\frac{1}{12} (h_{i,0} \frac{\partial \rho_i,0}{\partial t} + h_{i,1} \frac{\partial \rho_{i,1}}{\partial t} + h_{i,1} \frac{\partial \rho_i,0}{\partial t} + 3 h_{i,1} \frac{\partial \rho_{i,1}}{\partial t} + \rho_i,0 \frac{\partial h_i,0}{\partial t} \\
+ \rho_i,1 \frac{\partial h_{i,1}}{\partial t} + 3 \rho_i,1 \frac{\partial h_{i,1}}{\partial t}) + A \int_{x_{i-1}}^{x_i} \frac{\partial \alpha(t,x)}{\partial t} \phi_{i,1} \, dx
\end{align*}
\]  

(4.50)


Summary of Temporal Term Semi-Discretization

We have performed a semi-discretization of the temporal term of the enthalpy equation using a discontinuous Galerkin method with linear basis and test functions. Since flow is in only one direction, we use upwinding to deal with any flux terms that appear. The two equations for \(j=0\) and \(j=1\), respectively, that generate the system of ordinary differential equations that describe the temporal term on element \(i = 1, 2, ... N\) of the mesh are:

\[
\begin{align*}
\frac{1}{12} (3 h_{i,0} \frac{\partial \rho_i,0}{\partial t} + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} + h_{i,1} \frac{\partial \rho_i,0}{\partial t} + 3 h_{i,1} \frac{\partial \rho_{i,1}}{\partial t} + \rho_i,0 \frac{\partial h_i,0}{\partial t} \\
+ \rho_i,1 \frac{\partial h_{i,1}}{\partial t} + 3 \rho_i,1 \frac{\partial h_{i,1}}{\partial t}) + A \int_{x_{i-1}}^{x_i} \frac{\partial \alpha(t,x)}{\partial t} \phi_{i,0} \, dx
\end{align*}
\]  

(4.51)

\[
\begin{align*}
\frac{1}{12} (h_{i,0} \frac{\partial \rho_i,0}{\partial t} + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} + h_{i,1} \frac{\partial \rho_i,0}{\partial t} + 3 h_{i,1} \frac{\partial \rho_{i,1}}{\partial t} + \rho_i,0 \frac{\partial h_i,0}{\partial t} \\
+ \rho_i,1 \frac{\partial h_{i,1}}{\partial t} + 3 \rho_i,1 \frac{\partial h_{i,1}}{\partial t}) + A \int_{x_{i-1}}^{x_i} \frac{\partial \alpha(t,x)}{\partial t} \phi_{i,1} \, dx
\end{align*}
\]  

(4.52)

The Enthalpy Term

We begin by re-stating the enthalpy term of the enthalpy equation which is

\[
\int_{x_{i-1}}^{x_i} A \frac{\partial \rho u h}{\partial x} \nu \, dx
\]  

(4.53)
4.4. The Enthalpy Equation

The test function, $\nu$, is replaced with one of the non-zero basis functions, $\phi_{i,j}$, on element $i = 1, 2, \ldots N$ and $j = 0, 1$. The area of the pipe is assumed constant so we move it outside of the integral. The resulting term is

$$A \int_{x_{i-1}}^{x_i} \frac{\partial p u h}{\partial x} \phi_{i,j} \, dx \quad (4.54)$$

Integration by parts allows us to move the derivative to the test function and introduce flux into the system yielding

$$[A p u h \phi_{i,j}]_{x_{i-1}}^{x_i} - A \int_{x_{i-1}}^{x_i} p u h \frac{\partial \phi_{i,j}}{\partial x} \, dx \quad (4.55)$$

Next, we replace each term with its DG approximation. Additionally, we omit any terms that are uniformly zero on element $i = 1, 2, \ldots N$. The resulting equation is given by

$$[A p u h \phi_{i,j}]_{x_{i-1}}^{x_i} - A \int_{x_{i-1}}^{x_i} (\rho_{i,0} \phi_{i,0} + \rho_{i,1} \phi_{k,1})(u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1})(h_{i,0} \phi_{i,0} + h_{i,1} \phi_{i,1}) \frac{\partial \phi_{i,j}}{\partial x} \, dx \quad (4.56)$$

Distributing and collecting like terms results in

$$[A p u h \phi_{i,j}]_{x_{i-1}}^{x_i} - A \int_{x_{i-1}}^{x_i} (\rho_{i,0} h_{i,0} u_{i,0} \phi_{i,0}^3 + \rho_{i,1} h_{i,0} u_{i,1} \phi_{i,0}^2 \phi_{i,1} + \rho_{i,0} h_{i,1} u_{i,0} \phi_{i,0} \phi_{i,1}^2 + \rho_{i,0} h_{i,0} u_{i,1} \phi_{i,0} \phi_{i,1}^2 + \rho_{i,1} h_{i,0} u_{i,1} \phi_{i,0} \phi_{i,1}^2 + \rho_{i,1} h_{i,1} u_{i,0} \phi_{i,0} \phi_{i,1}^2 + \rho_{i,1} h_{i,1} u_{i,1} \phi_{i,0} \phi_{i,1}^2 + \rho_{i,1} h_{i,1} u_{i,1} \phi_{i,0} \phi_{i,1}^2) \frac{\partial \phi_{i,j}}{\partial x} \, dx \quad (4.57)$$

The Left Basis Function (j=0)

Replacing the test function, $nu$, with the left basis function $(j=0)$ produces

$$[A p u h \phi_{i,0}]_{x_{i-1}}^{x_i} - A \int_{x_{i-1}}^{x_i} (\rho_{i,0} h_{i,0} u_{i,0} \phi_{i,0}^3 + \rho_{i,1} h_{i,0} u_{i,1} \phi_{i,0}^2 \phi_{i,1} + \rho_{i,0} h_{i,1} u_{i,0} \phi_{i,0} \phi_{i,1}^2 + \rho_{i,0} h_{i,0} u_{i,1} \phi_{i,0} \phi_{i,1}^2 + \rho_{i,1} h_{i,0} u_{i,1} \phi_{i,0} \phi_{i,1}^2 + \rho_{i,1} h_{i,1} u_{i,0} \phi_{i,0} \phi_{i,1}^2 + \rho_{i,1} h_{i,1} u_{i,1} \phi_{i,0} \phi_{i,1}^2 + \rho_{i,1} h_{i,1} u_{i,1} \phi_{i,0} \phi_{i,1}^2) \frac{\partial \phi_{i,j}}{\partial x} \, dx \quad (4.58)$$
Chapter 4. Discontinuous Galerkin Spatial Semi-Discretization

Evaluating the derivatives and integrals where possible yields

\[
\left[ A \rho u h \phi_{i,0} \right]_{x_{i-1}}^{x_i} + \frac{A}{\Delta x_i} \left( \rho_{i,0} h_{i,0} u_{i,0} \frac{\Delta x_i}{4} + \rho_{i,1} h_{i,0} u_{i,0} \frac{\Delta x_i}{12} + \rho_{i,0} h_{i,1} u_{i,0} \frac{\Delta x_i}{12} + \rho_{i,1} h_{i,1} u_{i,1} \frac{\Delta x_i}{4} \right)
\]

Pulling out a \( \frac{\Delta x_i}{12} \) to obtain

\[
\left[ A \rho u h \phi_{i,0} \right]_{x_{i-1}}^{x_i} + \frac{A}{12} \left( 3 \rho_{i,0} h_{i,0} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,0} + \rho_{i,0} h_{i,1} u_{i,0} + \rho_{i,1} h_{i,1} u_{i,1} + \rho_{i,1} h_{i,1} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,1} + 3 \rho_{i,1} h_{i,1} u_{i,1} \right)
\]

We evaluate the flux with an upwind scheme since flow will only be to the right. This flux is represented below as

\[
\left[ A \rho u h \phi_{i,0} \right]_{x_{i-1}}^{x_i} = A \rho(x_i) u(x_i) h(x_i) \phi_{i,0}(x_i) - A \rho(x_{i-1}) u(x_{i-1}) h(x_{i-1}) \phi_{i,0}(x_{i-1})
\]

\[
= -A \rho(x_{i-1}) u(x_{i-1}) h(x_{i-1}) = -A \rho_{i-1,1} u_{i-1,1} h_{i-1,1}
\]

The enthalpy term with numerical flux is given by

\[
-A \rho_{i-1,1} u_{i-1,1} h_{i-1,1} + \frac{A}{12} \left( 3 \rho_{i,0} h_{i,0} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,0} + \rho_{i,0} h_{i,1} u_{i,0} + \rho_{i,1} h_{i,1} u_{i,1} + \rho_{i,1} h_{i,1} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,1} + 3 \rho_{i,1} h_{i,1} u_{i,1} \right)
\]

**The Right Basis Function (j=1)**

Replacing the the test function, \( n u \), with the right basis function (j=0) produces

\[
\left[ A \rho u h \phi_{i,1} \right]_{x_{i-1}}^{x_i} - \frac{A}{\Delta x_i} \int_{x_{i-1}}^{x_i} \left( \rho_{i,0} h_{i,0} u_{i,0} \phi_{i,0}^3 + \rho_{i,1} h_{i,0} u_{i,0} \phi_{i,0}^2 \phi_{i,1} + \rho_{i,0} h_{i,1} u_{i,0} \phi_{i,0}^2 \phi_{i,1} + \rho_{i,1} h_{i,1} u_{i,0} \phi_{i,0} \phi_{i,1}^2 + \rho_{i,0} h_{i,1} u_{i,1} \phi_{i,0}^2 \phi_{i,1} + \rho_{i,1} h_{i,1} u_{i,1} \phi_{i,1} \phi_{i,1}^2 \phi_{i,1} + \rho_{i,0} h_{i,1} u_{i,1} \phi_{i,0} \phi_{i,1}^2 \phi_{i,1} + \rho_{i,1} h_{i,1} u_{i,1} \phi_{i,0} \phi_{i,1}^3 \frac{\partial \phi_{i,1}}{\partial x} \right) dx
\]

\[
= \frac{A}{12} \left( 3 \rho_{i,0} h_{i,0} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,0} + \rho_{i,0} h_{i,1} u_{i,0} + \rho_{i,1} h_{i,1} u_{i,1} + \rho_{i,1} h_{i,1} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,1} + 3 \rho_{i,1} h_{i,1} u_{i,1} \right)
\]
4.4. The Enthalpy Equation

Integrals and derivatives are evaluated where possible yielding

\[ [A\rho h\phi_{i,1}]_{x_i}^x = \frac{A}{\Delta x_i} \left( \rho_{i,0} h_{i,0} u_{i,0} \frac{\Delta x_i}{4} + \rho_{i,1} h_{i,0} u_{i,0} \frac{\Delta x_i}{12} + \rho_{i,0} h_{i,1} u_{i,0} \frac{\Delta x_i}{12} + \rho_{i,0} h_{i,0} u_{i,1} \frac{\Delta x_i}{12} + \rho_{i,1} h_{i,1} u_{i,0} \frac{\Delta x_i}{12} + \rho_{i,1} h_{i,0} u_{i,1} \frac{\Delta x_i}{4} \right) \]  \hspace{1cm} (4.64)

We pull out a $\frac{\Delta x_i}{12}$ to obtain

\[ [A\rho h\phi_{i,1}]_{x_i}^x = \frac{A}{12} \left( 3\rho_{i,0} h_{i,0} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,0} + \rho_{i,0} h_{i,1} u_{i,0} + \rho_{i,0} h_{i,0} u_{i,1} + \rho_{i,1} h_{i,1} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,1} + 3\rho_{i,1} h_{i,1} u_{i,1} \right) \]  \hspace{1cm} (4.65)

The flux is evaluated by using an upwind scheme since flow should only be in one direction.

This is given by

\[ [A\rho h\phi_{i,1}]_{x_i}^x = A\rho(x_i) u(x_i) h(x_i) \phi_{i,1}(x_i) - A\rho(x_{i-1}) u(x_{i-1}) h(x_{i-1}) \phi_{i,1}(x_{i-1}) \]  \hspace{1cm} = A\rho(x_i) u(x_i) h(x_i) = A\rho_{i,1} u_{i,1} h_{i,1} \]  \hspace{1cm} (4.66)

Combining everything together yields:

\[ A\rho_{i,1} u_{i,1} h_{i,1} - \frac{A}{12} \left( 3\rho_{i,0} h_{i,0} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,0} + \rho_{i,0} h_{i,1} u_{i,0} + \rho_{i,0} h_{i,0} u_{i,1} + \rho_{i,1} h_{i,1} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,1} + 3\rho_{i,1} h_{i,1} u_{i,1} \right) \]  \hspace{1cm} (4.67)

**Summary of the Enthalpy Term Semi-Discretization**

We have spatially discretized the enthalpy term using a discontinuous Galerkin method with linear basis and test functions. Additionally, we have chosen to use an upwind scheme to deal with any fluxes since flow should only be in one direction. The ordinary differential equations that describe the enthalpy term for \( j=0 \) and \( j=1 \) respectfully on element \( i = 1, 2, ... N \) of the mesh are:

\[ -A\rho_{i-1,1} u_{i-1,1} h_{i-1,1} + \frac{A}{12} \left( 3\rho_{i,0} h_{i,0} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,0} + \rho_{i,0} h_{i,1} u_{i,0} + \rho_{i,0} h_{i,0} u_{i,1} + \rho_{i,1} h_{i,1} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,1} + 3\rho_{i,1} h_{i,1} u_{i,1} \right) \]  \hspace{1cm} (4.68)
Chapter 4. Discontinuous Galerkin Spatial Semi-Discretization

\[
A\rho_{i,1}u_{i,1}h_{i,1} - \frac{A}{12}(3\rho_{i,0}h_{i,0}u_{i,0} + \rho_{i,1}h_{i,0}u_{i,0} + \rho_{i,0}h_{i,1}u_{i,0} + \rho_{i,0}h_{i,0}u_{i,1} + \\
\rho_{i,1}h_{i,1}u_{i,0} + \rho_{i,1}h_{i,0}u_{i,1} + \rho_{i,0}h_{i,1}u_{i,1} + 3\rho_{i,1}h_{i,1}u_{i,1})
\]  
(4.69)

The Pressure Term

The pressure term of the enthalpy equation is given by

\[
\int_{x_{i-1}}^{x_i} Au \frac{\partial p}{\partial x} \nu \, dx
\]  
(4.70)

The cross sectional area of the pipe is assumed constant so we move it outside of the integral. Additionally, we replace \( \nu \) with a non-zero basis function, \( \phi_{i,j} \), \( j = 0, 1 \) on element \( i = 1, 2, \ldots N \) to produce

\[
A \int_{x_{i-1}}^{x_i} u \frac{\partial p}{\partial x} \phi_{i,j} \, dx
\]  
(4.71)

Next, we replace terms with their piece wise linear DG approximation. We omit any basis functions that are uniformly zero on element \( i = 1, 2, \ldots N \). It is assumed that the pressure profile is described for all time and space by \( \alpha(t, x) \). The resulting pressure term is

\[
A \int_{x_{i-1}}^{x_i} (u_i \phi_{i,0} + u_{i,1} \phi_{i,1}) \frac{\partial \alpha(t, x)}{\partial x} \phi_{i,j} \, dx
\]  
(4.72)

The Left Basis Function (j=0)

Replacing the test function \( \nu \) with the left basis function \( j = 0 \) yields

\[
A \int_{x_{i-1}}^{x_i} (u_i \phi_{i,0} + u_{i,1} \phi_{i,1}) \frac{\partial \alpha(t, x)}{\partial x} \phi_{i,0} \, dx
\]  
(4.73)

This is the final discretization for an arbitrary pressure drop. If we assume a constant pressure drop, \( \Delta p_i(t) \), along element \( i = 1, 2, \ldots N \) of the mesh we can make further simplifications. In other terms, if we assume that \( \frac{\partial \alpha(t, x)}{\partial x} \bigg|_{x \in [x_{i-1}, x_i]} = \Delta p_i(t) \) then we obtain

\[
A\Delta p_i(t) \int_{x_{i-1}}^{x_i} (u_i \phi_{i,0} + u_{i,1} \phi_{i,1}) \phi_{i,0} \, dx
\]  
(4.74)
4.4. The Enthalpy Equation

We are able to evaluate the integral as

\[ A\Delta p_i(t)(u_{i,0} \frac{\Delta x_i}{3} + u_{i,1} \frac{\Delta x_i}{6}) \]  

(4.75)

Pulling out a \( \frac{\Delta x_i}{6} \) results in

\[ A\Delta p_i(t)\frac{\Delta x_i}{6}(2u_{i,0} + u_{i+1}) \]  

(4.76)

The Right Basis Function (j=1)

Replacing the test function, \( \nu \), with the right basis function \( j = 1 \) yields

\[ A\int_{x_{i-1}}^{x_i} (u_{i,0}\phi_{i,0} + u_{i,1}\phi_{i,1}) \frac{\partial\alpha(t, x)}{\partial x}\phi_{i,1} \, dx \]  

(4.77)

This is the final discretization for an arbitrary pressure drop. If we assume a constant pressure drop, \( \Delta p_i(t) \), along element \( i = 1, 2, \ldots N \) of the mesh we can make further simplifications. In other terms, if we assume that \( \frac{\partial\alpha(t,x)}{\partial x}\bigg|_{x \in [x_{i-1}, x_i]} = \Delta p_i(t) \) then we obtain

\[ A\Delta p_i(t)\int_{x_{i-1}}^{x_i} (u_{i,0}\phi_{i,0} + u_{i,1}\phi_{i,1})\phi_{i,1} \, dx \]  

(4.78)

We are able to evaluate the integral to produce

\[ A\Delta p_i(t)(u_{i,0} \frac{\Delta x_i}{6} + u_{i,1} \frac{\Delta x_i}{3}) \]  

(4.79)

Pulling out a \( \frac{\Delta x_i}{6} \) results in

\[ A\Delta p_i(t)\frac{\Delta x_i}{6}(u_{i,0} + 2u_{i+1}) \]  

(4.80)

Summary of Pressure Term Semi-Discretization

We have spatially discretized the pressure term using a discontinuous Galerkin method with linear basis and test functions. The differential equations for \( j = 0 \) and \( j = 1 \) that describe the pressure term on element \( i = 1, 2, \ldots N \) of the mesh respectively are:

\[ A\int_{x_{i-1}}^{x_i} (u_{i,0}\phi_{i,0} + u_{i,1}\phi_{i,1}) \frac{\partial\alpha(t, x)}{\partial x}\phi_{i,0} \, dx \]  

(4.81)
Chapter 4. Discontinuous Galerkin Spatial Semi-Discretization

\[ A \int_{x_{i-1}}^{x_i} (u_{i,0}\phi_{i,0} + u_{i,1}\phi_{i,1}) \frac{\partial \alpha(t,x)}{\partial x} \phi_{i,1} \, dx \]  \hspace{1cm} (4.82)

If we assume a constant pressure drop, \( \Delta p_i(t) \), along element \( i = 1, 2, \ldots, N \) of the mesh we can make further simplifications. In other terms, if we assume that \( \frac{\partial \alpha(t,x)}{\partial x} \big|_{x \in [x_{i-1}, x_i]} = \Delta p_i(t) \) then the discretization simplifies to:

\[ A\Delta p_i(t) \frac{\Delta x_i}{6}(2u_{i,0} + u_{i+1}) \]  \hspace{1cm} (4.83)

\[ A\Delta p_i(t) \frac{\Delta x_i}{6}(u_{i,0} + 2u_{i+1}) \]  \hspace{1cm} (4.84)

**The Shear Term**

The shear term for the enthalpy equation is given by

\[ -\int_{x_{i-1}}^{x_i} \frac{4}{3} A\mu \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) \nu \, dx \]  \hspace{1cm} (4.85)

We assume that \( \mu \) and the cross sectional area of the pipe are approximately constant on the interval \([x_{i-1}, x_i]\). Moving \( \mu \) and \( A \) outside of the integral and replacing the test function \( \nu \) with a non-zero basis function \( \phi_{i,j} \) for \( j = 0, 1 \) on element \( i = 1, 2, \ldots, N \) and we obtain

\[ -\frac{4}{3} A\mu \int_{x_{i-1}}^{x_i} u \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial x} \right) \phi_{i,j} \, dx \]  \hspace{1cm} (4.86)

We now introduce a velocity flux \( \psi \) which is related to the velocity through the velocity flux correlation. The velocity flux correlation is given by

\[ \psi - \frac{\partial u}{\partial x} = 0 \]  \hspace{1cm} (4.87)

We assume that this velocity flux has as similar piece-wise linear representation as all other variables. We repeat the same steps as earlier on the velocity flux correlation. This leads to the following system of equations

\[ -\frac{4}{3} A\mu \int_{x_{i-1}}^{x_i} u \frac{\partial \psi}{\partial x} \phi_{i,j} \, dx \]  \hspace{1cm} (4.88)

\[ \int_{x_{i-1}}^{x_i} \psi \phi_{i,j} \, dx - \int_{x_{i-1}}^{x_i} \frac{\partial u}{\partial x} \phi_{i,j} \, dx = 0 \]  \hspace{1cm} (4.89)
4.4. The Enthalpy Equation

Performing integration by parts on both equations moves the derivative to the test function and introduces flux into the system. The shear term becomes

\[- \frac{4}{3} A \mu u \psi \phi_{i,j} \bigg|_{x_{i-1}}^{x_i} + \frac{4}{3} A \mu \int_{x_{i-1}}^{x_i} \psi \frac{\partial \phi_{i,j}}{\partial x} \, dx \]  \hspace{1cm} (4.90)
\[\int_{x_{i-1}}^{x_i} \psi \phi_{i,j} \, dx - [u \phi_{i,j}]_{x_{i-1}}^{x_i} + \int_{x_{i-1}}^{x_i} u \frac{\partial \phi_{i,j}}{\partial x} \, dx = 0 \]  \hspace{1cm} (4.91)

We now use the product rule to obtain

\[- \frac{4}{3} A \mu u \psi \phi_{i,j} \bigg|_{x_{i-1}}^{x_i} + \frac{4}{3} A \mu \int_{x_{i-1}}^{x_i} \psi \left( \frac{\partial \phi_{i,j}}{\partial x} + \phi_{i,j} \frac{\partial u}{\partial x} \right) \, dx \]  \hspace{1cm} (4.92)
\[\int_{x_{i-1}}^{x_i} \psi \phi_{i,j} \, dx - [u \phi_{i,j}]_{x_{i-1}}^{x_i} + \int_{x_{i-1}}^{x_i} u \frac{\partial \phi_{i,j}}{\partial x} \, dx = 0 \]  \hspace{1cm} (4.93)

Finally, we replace each of quantities by their DG approximations. Additionally, we omit any basis functions that are uniformly zero on element \(i = 1, 2, \ldots N\) of the mesh to produce the following system of equations

\[- \frac{4}{3} A \mu u \psi \phi_{i,j} \bigg|_{x_{i-1}}^{x_i} + \frac{4}{3} A \mu \int_{x_{i-1}}^{x_i} \left( \psi_{i,0} \phi_{i,0} + \psi_{i,1} \phi_{i,1} \right) \left( u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1} \right) \frac{\partial \phi_{i,j}}{\partial x} + \phi_{i,j} \frac{\partial (u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1})}{\partial x} \, dx \]  \hspace{1cm} (4.94)
\[\int_{x_{i-1}}^{x_i} \left( \psi_{i,0} \phi_{i,0} + \psi_{i,1} \phi_{i,1} \right) \phi_{i,j} \, dx - [u \phi_{i,j}]_{x_{i-1}}^{x_i} + \int_{x_{i-1}}^{x_i} \left( u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1} \right) \frac{\partial \phi_{i,j}}{\partial x} \, dx = 0 \]  \hspace{1cm} (4.95)

The Left Basis Function (\(j=0\))

Replacing the test function, \(\nu\) with the left basis function \(j = 0\) yields

\[- \frac{4}{3} A \mu u \psi \phi_{i,0} \bigg|_{x_{i-1}}^{x_i} + \frac{4}{3} A \mu \int_{x_{i-1}}^{x_i} \left( \psi_{i,0} \phi_{i,0} + \psi_{i,1} \phi_{i,1} \right) \left( u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1} \right) \frac{\partial \phi_{i,0}}{\partial x} + \phi_{i,0} \frac{\partial (u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1})}{\partial x} \, dx \]  \hspace{1cm} (4.96)
\[\int_{x_{i-1}}^{x_i} \left( \psi_{i,0} \phi_{i,0} + \psi_{i,1} \phi_{i,1} \right) \phi_{i,0} \, dx - [u \phi_{i,0}]_{x_{i-1}}^{x_i} + \int_{x_{i-1}}^{x_i} \left( u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1} \right) \frac{\partial \phi_{i,0}}{\partial x} \, dx = 0 \]  \hspace{1cm} (4.97)

We now evaluate all the fluxes using upwinding in the shear term and downwinding in velocity flux correlation. We do this for stability purposes explained by Zhang in [30] to
Chapter 4. Discontinuous Galerkin Spatial Semi-Discretization

obtain

\[ \frac{4}{3} A \mu u_{i-1,1} \psi_{i-1,1} + \frac{4}{3} A \mu \int_{x_{i-1}}^{x_i} (\psi_{i,0} \phi_{i,0} + \psi_{i,1} \phi_{i,1}) \left( \frac{u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1}}{\Delta x_i} \right) dx \]  

(4.98)

\[ \int_{x_{i-1}}^{x_i} (\psi_{i,0} \phi_{i,0} + \psi_{i,1} \phi_{i,1}) \phi_{i,0} \, dx + u_{i,0} + \int_{x_{i-1}}^{x_i} (u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1}) \frac{\partial \phi_{i,0}}{\partial x} \, dx = 0 \]  

(4.99)

Evaluating all integrals produces

\[ \frac{4}{3} A \mu u_{i-1,1} \psi_{i-1,1} + \frac{4}{3} A \mu \frac{-\Delta x_i}{6 \Delta x_i} (4 \psi_{i,0} u_{i,0} - \psi_{i,0} u_{i,1} + 2 \psi_{i,1} u_{i,0} + \psi_{i,1} u_{i,1}) \]  

(4.100)

\[ \left( \frac{\Delta x_i}{3} \psi_{i,0} + \frac{\Delta x_i}{6} \psi_{i,1} \right) + u_{i,0} - \frac{1}{\Delta x_i} \left( \frac{\Delta x_i}{2} u_{i,0} + \frac{\Delta x_i}{2} u_{i,1} \right) \right) \, dx = 0 \]  

(4.101)

Combining like terms and simplifying yields

\[ \frac{4}{3} A \mu u_{i-1,1} \psi_{i-1,1} - \frac{2}{9} A \mu (4 \psi_{i,0} u_{i,0} - \psi_{i,0} u_{i,1} + 2 \psi_{i,1} u_{i,0} + \psi_{i,1} u_{i,1}) \]  

(4.102)

\[ \frac{\Delta x_i}{6} (2 \psi_{i,0} + \psi_{i,1}) + u_{i,0} - \frac{1}{2} (u_{i,0} + u_{i,1}) = 0 \]  

(4.103)

The Right Basis Function (j=1)

Replacing the test function, \( \nu \) with the left basis function \( j = 0 \) yields

\[ - \left[ \frac{4}{3} A \mu \psi_{i,1} \right]_{x_{i-1}}^{x_i} + \frac{4}{3} A \mu \int_{x_{i-1}}^{x_i} (\psi_{i,0} \phi_{i,0} + \psi_{i,1} \phi_{i,1}) \left( \frac{u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1}}{\partial x} \right) dx \]  

(4.104)

\[ \int_{x_{i-1}}^{x_i} (\psi_{i,0} \phi_{i,0} + \psi_{i,1} \phi_{i,1}) \phi_{i,1} \, dx - [u \phi_{i,1}]_{x_{i-1}}^{x_i} + \int_{x_{i-1}}^{x_i} (u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1}) \frac{\partial \phi_{i,1}}{\partial x} \, dx = 0 \]  

(4.105)

We now evaluate fluxes using upwinding for the shear term and downwinding for the velocity flux correlations for stability purposes as explained by Zhang in [30]. The resulting shear
4.4. The Enthalpy Equation

term is given by

\[- \frac{4}{3} A \mu u_{i,1} \psi_{i,1} + \frac{4}{3} A \mu \int_{x_{i-1}}^{x_i} \left( \psi_{i,0} \phi_{i,0} + \psi_{i,1} \phi_{i,1} \right) \left( (u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1}) \frac{\partial \phi_{i,1}}{\partial x} + \phi_{i,1} \frac{\partial(u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1})}{\partial x} \right) dx \]  
(4.106)

\[ \left( \frac{\Delta x_i}{6} \psi_{i,0} + \frac{\Delta x_i}{3} \psi_{i,1} \phi_{i,1} \right) - u_{i+1,0} + \int_{x_{i-1}}^{x_i} \left( u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1} \right) \frac{\partial \phi_{i,1}}{\partial x} dx = 0 \]  
(4.107)

All integrals are evaluated to obtain

\[- \frac{4}{3} A \mu u_{i,1} \psi_{i,1} + \frac{4}{3} A \mu \frac{\Delta x_i}{6} (\psi_{i,0} u_{i,0} + 2 \psi_{i,0} u_{i,1} - \psi_{i,1} u_{i,0} + 4 \psi_{i,1} u_{i,1}) \]  
(4.108)

\[ \frac{\Delta x_i}{6} (\psi_{i,0} + 2 \psi_{i,1}) - u_{i+1,0} + \frac{1}{2} (u_{i,0} + u_{i,1}) = 0 \]  
(4.109)

Simplifying and combining terms yields

\[- \frac{4}{3} A \mu u_{i,1} \psi_{i,1} + \frac{2}{9} A \mu (\psi_{i,0} u_{i,0} + 2 \psi_{i,0} u_{i,1} - \psi_{i,1} u_{i,0} + 4 \psi_{i,1} u_{i,1}) \]  
(4.110)

\[ \frac{\Delta x_i}{6} (\psi_{i,0} + 2 \psi_{i,1}) - u_{i+1,0} + \frac{1}{2} (u_{i,0} + u_{i,1}) = 0 \]  
(4.111)

**Summary of The Shear Term Semi-Discretization**

We have spatially discretized the shear term using a discontinuous Galerkin method with linear basis and test functions. Due to the second derivative, the additional velocity flux variable \( \psi \) was introduced along with two additional velocity flux correlations relating \( \psi \) to \( u \). For stability purposes, we choose to use an upwind scheme for the shear term and downwinding for the velocity flux correlations. The system of equations for \( j = 0 \) and \( j = 1 \) respectfully that discretize the shear term on element \( i = 1, 2, ..., N \) of the mesh are:

\[ \frac{4}{3} A \mu u_{i-1,1} \psi_{i-1,1} - \frac{2}{9} A \mu (4 \psi_{i,0} u_{i,0} - \psi_{i,0} u_{i,1} + 2 \psi_{i,1} u_{i,0} + \psi_{i,1} u_{i,1}) \]  
(4.112)

\[- \frac{4}{3} A \mu u_{i,1} \psi_{i,1} + \frac{2}{9} A \mu (\psi_{i,0} u_{i,0} + 2 \psi_{i,0} u_{i,1} - \psi_{i,1} u_{i,0} + 4 \psi_{i,1} u_{i,1}) \]  
(4.113)

\[ \frac{\Delta x_i}{6} (2 \psi_{i,0} + \psi_{i,1}) + u_{i,0} - \frac{1}{2} (u_{i,0} + u_{i,1}) = 0 \]  
(4.114)

\[ \frac{\Delta x_i}{6} (\psi_{i,0} + 2 \psi_{i,1}) - u_{i+1,0} + \frac{1}{2} (u_{i,0} + u_{i,1}) = 0 \]  
(4.115)
The Heat Conduction Term

The heat conductance term for the enthalpy equation is given by

$$\int_{x_{i-1}}^{x_i} A \frac{\partial}{\partial x} \left( \kappa \frac{\partial T}{\partial x} \right) \nu \, dx \quad (4.116)$$

It is assumed that $\kappa$ and the cross-sectional area of the pipe is approximately constant on the interval $[x_{i-1}, x_i]$. Additionally, we replace $\nu$ with a non-zero basis function $\phi_{i,j} j = 0, 1$ on element $i = 1, 2, ... N$ to produce

$$A \int_{x_{i-1}}^{x_i} \left( \frac{\partial T}{\partial x} \right) \phi_{i,j} \, dx \quad (4.117)$$

Integration by parts allows us to move the derivative to the test function and introduce flux into the system. This results in

$$[A \kappa \frac{\partial T}{\partial x} \phi_{i,j}]_{x_{i-1}}^{x_i} - A \kappa \int_{x_{i-1}}^{x_i} \frac{\partial T}{\partial x} \frac{\partial \phi_{i,j}}{\partial x} \, dx \quad (4.118)$$

We replace the temperature derivative term with the temperature flux variable $\Gamma$. $\Gamma$ is related to temperature through the temperature flux correlation given by

$$\Gamma - \frac{\partial T}{\partial x} = 0 \quad (4.119)$$

We assume that $\Gamma$ has a DG approximation similar to all other variables. Repeating the same steps as earlier on the velocity flux correlation results in the following system of equations

$$[A \kappa \Gamma \phi_{i,j}]_{x_{i-1}}^{x_i} - A \kappa \int_{x_{i-1}}^{x_i} \Gamma \frac{\partial \phi_{i,j}}{\partial x} \, dx \quad (4.120)$$

$$\int_{x_{i-1}}^{x_i} \Gamma \phi_{i,j} \, dx - [T \phi_{i,j}]_{x_{i-1}}^{x_i} + \int_{x_{i-1}}^{x_i} T \frac{\partial \phi_{i,j}}{\partial x} \, dx = 0 \quad (4.121)$$

We now replace both $\Gamma$ and $T$ with their DG approximation. Additionally, we ignore any basis functions that are uniformly zero on element $i = 1, 2, ... N$ to create

$$[A \kappa \Gamma \phi_{i,j}]_{x_{i-1}}^{x_i} - A \kappa \int_{x_{i-1}}^{x_i} (\Gamma_{i,0} \phi_{i,0} + \Gamma_{i,1} \phi_{i,1}) \frac{\partial \phi_{i,j}}{\partial x} \, dx \quad (4.122)$$

$$\int_{x_{i-1}}^{x_i} (\Gamma_{i,0} \phi_{i,0} + \Gamma_{i,1} \phi_{i,1}) \phi_{i,j} \, dx - [T \phi_{i,j}]_{x_{i-1}}^{x_i} + \int_{x_{i-1}}^{x_i} (T_{i,0} \phi_{i,0} + T_{i,1} \phi_{i,1}) \frac{\partial \phi_{i,j}}{\partial x} \, dx = 0 \quad (4.123)$$
4.4. The Enthalpy Equation

The Left Basis Function \((j=0)\)

Replacing the test function, \(\nu\), with the left basis function \((j = 0)\) yields

\[
[A\kappa \Gamma \phi_{i,0}]_{x_{i-1}}^{x_i} - A\kappa \int_{x_{i-1}}^{x_i} (\Gamma_{i,0} \phi_{i,0} + \Gamma_{i,1} \phi_{i,1}) \frac{\partial \phi_{i,0}}{\partial x} \, dx
\]

\[
\int_{x_{i-1}}^{x_i} (\Gamma_{i,0} \phi_{i,0} + \Gamma_{i,1} \phi_{i,1}) \phi_{i,0} \, dx - [T \phi_{i,0}]_{x_{i-1}}^{x_i} + \int_{x_{i-1}}^{x_i} (T_{i,0} \phi_{i,0} + T_{i,1} \phi_{i,1}) \frac{\partial \phi_{i,0}}{\partial x} \, dx = 0
\]

(Equation 4.124)

(Equation 4.125)

Evaluating the derivatives where possible produces

\[
[A\kappa \Gamma \phi_{i,0}]_{x_{i-1}}^{x_i} - A\kappa \int_{x_{i-1}}^{x_i} (\Gamma_{i,0} \phi_{i,0} + \Gamma_{i,1} \phi_{i,1}) \frac{1}{\Delta x_i} \, dx
\]

\[
\int_{x_{i-1}}^{x_i} (\Gamma_{i,0} \phi_{i,0} + \Gamma_{i,1} \phi_{i,1}) \phi_{i,0} \, dx - [T \phi_{i,0}]_{x_{i-1}}^{x_i} + \int_{x_{i-1}}^{x_i} (T_{i,0} \phi_{i,0} + T_{i,1} \phi_{i,1}) \frac{1}{\Delta x_i} \, dx = 0
\]

(Equation 4.126)

(Equation 4.127)

Simplifying and evaluating integrals we obtain

\[
[A\kappa \Gamma \phi_{i,0}]_{x_{i-1}}^{x_i} + \frac{1}{2} A\kappa (\Gamma_{i,0} + \Gamma_{i,1})
\]

\[
\frac{\Delta x_i}{6} (2\Gamma_{i,0} + \Gamma_{i,1}) - [T \phi_{i,0}]_{x_{i-1}}^{x_i} - \frac{1}{2} (T_{i,0} + T_{i,1}) = 0
\]

(Equation 4.128)

(Equation 4.129)

We evaluate the flux in the heat conduction term using upwinding and downwinding for the flux in the temperature correlations. The resulting system of equations is

\[
-A\kappa \Gamma_{i-1,1} + \frac{1}{2} A\kappa (\Gamma_{i,0} + \Gamma_{i,1})
\]

\[
\frac{\Delta x_i}{6} (2\Gamma_{i,0} + \Gamma_{i,1}) + T_{i,0} - \frac{1}{2} (T_{i,0} + T_{i,1}) = 0
\]

(Equation 4.130)

(Equation 4.131)

Combining like terms results in

\[
-A\kappa \Gamma_{i-1,1} + \frac{1}{2} A\kappa (\Gamma_{i,0} + \Gamma_{i,1})
\]

\[
\frac{\Delta x_i}{6} (2\Gamma_{i,0} + \Gamma_{i,1}) + \frac{1}{2} (T_{i,0} - T_{i,1}) = 0
\]

(Equation 4.132)

(Equation 4.133)
The Right Basis Function (j=1)

Replacing the test function, $\nu$, with the right basis function (j=1) yields

$$[A\kappa \phi_{i,1}]_{x_i}^{x_i} - A\kappa \int_{x_i}^{x_i} (\Gamma_{i,0}\phi_{i,0} + \Gamma_{i,1}\phi_{i,1}) \frac{\partial \phi_{i,1}}{\partial x} dx$$

$$\int_{x_{i-1}}^{x_i} (\Gamma_{i,0}\phi_{i,0} + \Gamma_{i,1}\phi_{i,1})\phi_{i,1} dx - [T\phi_{i,1}]_{x_{i-1}}^{x_i} + \int_{x_{i-1}}^{x_i} (T_{i,0}\phi_{i,0} + T_{i,1}\phi_{i,1}) \frac{\partial \phi_{i,1}}{\partial x} dx = 0$$  \hspace{1cm} (4.135)

Evaluating the derivatives result in

$$[A\kappa \phi_{i,1}]_{x_{i-1}}^{x_i} - \frac{1}{\Delta x_i} A\kappa \int_{x_{i-1}}^{x_i} (\Gamma_{i,0}\phi_{i,0} + \Gamma_{i,1}\phi_{i,1}) dx$$

$$\int_{x_{i-1}}^{x_i} (\Gamma_{i,0}\phi_{i,0} + \Gamma_{i,1}\phi_{i,1})\phi_{i,1} dx - [T\phi_{i,1}]_{x_{i-1}}^{x_i} + \frac{1}{\Delta x_i} \int_{x_{i-1}}^{x_i} (T_{i,0}\phi_{i,0} + T_{i,1}\phi_{i,1}) dx = 0$$  \hspace{1cm} (4.137)

Simplifying and evaluating integrals produces

$$[A\kappa \phi_{i,1}]_{x_{i-1}}^{x_i} - \frac{1}{2} A\kappa (\Gamma_{i,0} + \Gamma_{i,1})$$

$$\frac{\Delta x_i}{6} (\Gamma_{i,0} + 2\Gamma_{i,1}) - [T\phi_{i,1}]_{x_{i-1}}^{x_i} + \frac{1}{2} (T_{i,0} + T_{i,1}) = 0$$  \hspace{1cm} (4.138)

We now evaluate the fluxes using upwinding in the heat conductance term and downwinding in temperature flux correlation. We do this for stability purposes explained by Zhang in [30] yields

$$A\kappa \Gamma_{i,1} - \frac{1}{2} A\kappa (\Gamma_{i,0} + \Gamma_{i,1})$$

$$\frac{\Delta x_i}{6} (\Gamma_{i,0} + 2\Gamma_{i,1}) - T_{i+1,0} + \frac{1}{2} (T_{i,0} + T_{i,1}) = 0$$  \hspace{1cm} (4.140)

Summary of Heat Conductance Semi-Discretization

We have spatially discretized the heat conductance term using a discontinuous Galerkin method with linear basis and test functions. To deal with the diffusive term, we introduce a temperature flux variable \(\Gamma\) and introduce temperature flux correlations relating \(\Gamma\) to \(T\). Additionally, we have chosen to use an upwind scheme for the heat conductance term and a downwind scheme for the temperature flux correlations for stability purposes. The system of ordinary differential equations and temperature flux correlations for \(j = 0\) and \(j = 1\)
4.4. The Enthalpy Equation

respectively that discretize the heat conductance term on element \( i = 1, 2, \ldots N \) of the mesh are:

\[
-\frac{1}{2} A \kappa (\Gamma_{i-1,1} + \Gamma_{i,1}) + A \kappa \Gamma_{i,1} - \frac{1}{2} A \kappa (\Gamma_{i,0} + \Gamma_{i,1}) \quad (4.141)
\]

\[
\Delta x_i \left( 2 \Gamma_{i,0} + \Gamma_{i,1} \right) + \frac{1}{2} (T_{i,0} - T_{i,1}) = 0 \quad (4.142)
\]

\[
\Delta x_i \left( \Gamma_{i,0} + 2 \Gamma_{i,1} \right) - T_{i+1,0} + \frac{1}{2} (T_{i,0} + T_{i,1}) = 0 \quad (4.143)
\]

\[
\frac{\Delta x_i}{6} (\Gamma_{i,0} + 2 \Gamma_{i,1}) - T_{i+1,0} + \frac{1}{2} (T_{i,0} + T_{i,1}) = 0 \quad (4.144)
\]

**The External Heat Transfer Term**

The external heat transfer term of the enthalpy equation is given by

\[
\int_{x_{i-1}}^{x_i} \hat{Q}_v \, dx \quad (4.145)
\]

Newton’s law of cooling is given as

\[
\hat{Q} = UP(T - T_{surr}) \quad (4.146)
\]

\( T_{surr} \) is the temperature of the surroundings (ambient temperature) on the other side of the pipe wall. We assume that the surrounding temperature is constant for all time and space. \( P \) is the perimeter of the pipe and is assumed to be constant (due to the pipe area being constant). \( U \) is the heat transfer coefficient and is also assumed constant. Additionally, we replace the test function with a non-zero basis function \( \phi_{i,j} \) \( j = 0, 1 \) on element \( i = 1, 2, \ldots N \) to produce

\[
\int_{x_{i-1}}^{x_i} UP(T - T_{surr}) \phi_{i,j} \, dx \quad (4.147)
\]

Moving all of the constant terms outside of the integral yields

\[
UP \int_{x_{i-1}}^{x_i} (T - T_{surr}) \phi_{i,j} \, dx \quad (4.148)
\]

We replace the temperature term with its DG approximation. Additionally, we omit any basis functions that are uniformly zero on element \( i = 1, 2, \ldots N \) of the mesh. The resulting
term is

\[ UP \int_{x_{i-1}}^{x_i} (T_{i,0} \phi_{i,0} + T_{i,1} \phi_{i,1} - T_{surr}) \phi_{i,j} \, dx \]  

(4.149)

The Left Basis Function (j=0)

Replacing the test function, \( \nu \), with the left basis function \( j = 0 \) yields

\[ UP \int_{x_{i-1}}^{x_i} (T_{i,0} \phi_{i,0} + T_{i,1} \phi_{i,1} - T_{surr}) \phi_{i,0} \, dx \]  

(4.150)

Evaluating integrals produces

\[ UP(T_{i,0} \frac{\Delta x_i}{3} + T_{i,1} \frac{\Delta x_i}{6} - T_{surr} \frac{\Delta x_i}{2}) \]  

(4.151)

Simplifying results in

\[ \frac{\Delta x_i}{6} UP(2T_{i,0} + T_{i,1} - 3T_{surr}) \]  

(4.152)

The Right Basis Function (j=1)

Replacing the test function, \( \nu \) with the right basis function \( j = 1 \) yields

\[ UP \int_{x_{i-1}}^{x_i} (T_{i,0} \phi_{i,0} + T_{i,1} \phi_{i,1} - T_{surr}) \phi_{i,1} \, dx \]  

(4.153)

Evaluating integrals produces

\[ UP(T_{i,0} \frac{\Delta x_i}{6} + T_{i,1} \frac{\Delta x_i}{3} - T_{surr} \frac{\Delta x_i}{2}) \]  

(4.154)

Simplifying results in

\[ \frac{\Delta x_i}{6} UP(T_{i,0} + 2T_{i,1} - 3T_{surr}) \]  

(4.155)

Summary of External Heat Transfer Term Semi-Discretization

We spatially discretized the external heat transfer term using a discontinuous Galerkin method with linear basis and test functions. We assume a constant surrounding temperature \( T_{surr} \) outside of the pipe, constant pipe perimeter \( P \), and a constant heat transfer coefficient \( U \). We use Newton’s Law of Cooling to characterize heat transfer from inside the
4.4. The Enthalpy Equation

pipe to outside of it. The two equations for $j = 0$ and $j = 1$ respectfully that discretize the external heat transfer term on element $i = 1, 2, ...N$ of the mesh are:

\[
UP(T_{i,0} \frac{\Delta x_i}{3} + T_{i,1} \frac{\Delta x_i}{6} - T_{surr} \frac{\Delta x_i}{2}) \quad (4.156)
\]
\[
UP(T_{i,0} \frac{\Delta x_i}{6} + T_{i,1} \frac{\Delta x_i}{3} - T_{surr} \frac{\Delta x_i}{2}) \quad (4.157)
\]

The Viscous Dissipation Term

The viscous dissipation term of the enthalpy equation is given by

\[
\Phi = \frac{4\mu}{3} \left( \frac{\partial u}{\partial x} \right)^2 \quad (4.158)
\]

Substituting this expression into the viscous dissipation term of the enthalpy equation results in

\[
\int_{x_{i-1}}^{x_i} \frac{4\mu}{3} \left( \frac{\partial u}{\partial x} \right)^2 \nu \, dx \quad (4.159)
\]

Next, we replace the test function $\nu$ with a non-zero basis function $\phi_{i,j} j = 0, 1$ on element $i = 1, 2, ...N$. We also assume that the viscosity $\mu$ is approximately constant on the interval $[x_{i-1}, x_i]$ yielding

\[
\frac{4\mu}{3} \int_{x_{i-1}}^{x_i} \left( \frac{\partial u}{\partial x} \right)^2 \phi_{i,j} \, dx \quad (4.160)
\]

We now replace the velocity term with its piece wise linear DG approximation and omit any basis functions that are uniformly zero on element $i = 1, 2, ...N$ to produce

\[
\frac{4\mu}{3} \int_{x_{i-1}}^{x_i} \left( \frac{\partial (u_{i,0}\phi_{i,0} + u_{i,1}\phi_{i,1})}{\partial x} \right)^2 \phi_{i,j} \, dx \quad (4.161)
\]

Evaluating the derivative results in

\[
\frac{4\mu}{3} \int_{x_{i-1}}^{x_i} \left( \frac{u_{i,1} - u_{i,0}}{\Delta x_i} \right)^2 \phi_{i,j} \, dx \quad (4.162)
\]
Chapter 4. Discontinuous Galerkin Spatial Semi-Discretization

**The left Basis Function** (j=0)

Replacing the test function, \( \nu \), with the left basis function (j=0) yields

\[
\frac{4\mu}{3} \int_{x_{i-1}}^{x_i} \left( \frac{u_{i,1} - u_{i,0}}{\Delta x_i} \right)^2 \phi_{i,0} \, dx
\]  

(4.163)

Evaluating the integral results in

\[
\frac{4\mu}{3} \left( \frac{u_{i,1} - u_{i,0}}{\Delta x_i} \right)^2 \frac{\Delta x_i}{2}
\]  

(4.164)

Simplifying we obtain

\[
\frac{2\mu}{3\Delta x_i} (u_{i,1} - u_{i,0})^2
\]  

(4.165)

**The Right Basis Function** (j=1)

Replacing the test function, \( \nu \), with the left basis function (j=0) yields

\[
\frac{4\mu}{3} \int_{x_{i-1}}^{x_i} \left( \frac{u_{i,1} - u_{i,0}}{\Delta x_i} \right)^2 \phi_{i,1} \, dx
\]  

(4.166)

Evaluating the integral results in

\[
\frac{4\mu}{3} \left( \frac{u_{i,1} - u_{i,0}}{\Delta x_i} \right)^2 \frac{\Delta x_i}{2}
\]  

(4.167)

Simplifying we obtain

\[
\frac{2\mu}{3\Delta x_i} (u_{i,1} - u_{i,0})^2
\]  

(4.168)

**Summary of Viscous Dissipation Term Discretization**

We have discretized the viscous dissipation term using a discontinuous Galerkin method with linear basis and test functions. The two equations for j=0 and j=1 respectively that discretize the viscous dissipation term on element \( i = 1, 2, \ldots N \) of the mesh are:

\[
\frac{2\mu}{3\Delta x_i} (u_{i,1} - u_{i,0})^2
\]  

(4.169)

\[
\frac{2\mu}{3\Delta x_i} (u_{i,1} - u_{i,0})^2
\]  

(4.170)
4.4. The Enthalpy Equation

**The Friction Term**

The friction term for the enthalpy equation is given by

\[- \int_{x_{i-1}}^{x_i} u F_F(t, x) \nu \, dx\]  \hspace{1cm} (4.171)

Replacing the test function with a non-zero basis function \( \phi_{i,j} \), \( j = 0, 1 \) on element \( i = 1, 2, \ldots N \) results in

\[- \int_{x_{i-1}}^{x_i} u F_F(t, x) \phi_{i,j} \, dx\]  \hspace{1cm} (4.172)

Next, we replace the velocity term with its piece-wise linear DG approximation and omit any basis functions that are uniformly zero on element \( i = 1, 2, \ldots N \) to produce

\[- \int_{x_{i-1}}^{x_i} (u_i,0 \phi_{i,0} + u_i,1 \phi_{i,1}) F_F(t, x) \phi_{i,j} \, dx\]  \hspace{1cm} (4.173)

**The Left Basis Function \((j=0)\)**

Replacing the test function, \( \nu \), with the left basis function \((j=0)\) yields

\[- \int_{x_{i-1}}^{x_i} (u_i,0 \phi_{i,0} + u_i,1 \phi_{i,1}) F_F(t, x) \phi_{i,0} \, dx\]  \hspace{1cm} (4.174)

This is the final discretization for an arbitrary friction factor. If we assume that the friction factor is constant on the interval \([x_{i-1}, x_i]\) then we can integrate to obtain

\[- F_F(u_{i,0} \frac{\Delta x_i}{3} + u_{i,1} \frac{\Delta x_i}{6})\]  \hspace{1cm} (4.175)

Simplifying results in

\[- F_F \frac{\Delta x_i}{6} (2u_{i,0} + u_{i,1})\]  \hspace{1cm} (4.176)

**The Right Basis Function \((j=1)\)**

Replacing the test function with the left basis function \((j=1)\) yields

\[- \int_{x_{i-1}}^{x_i} (u_i,0 \phi_{i,0} + u_i,1 \phi_{i,1}) F_F(t, x) \phi_{i,1} \, dx\]  \hspace{1cm} (4.177)
This is the final discretization for an arbitrary friction factor. If we assume that the frictional force is constant on the interval \([x_{i-1}, x_i]\) then we can integrate to obtain

\[- F_F(u_{i,0} \frac{\Delta x_i}{6} + u_{i,1} \frac{\Delta x_i}{3}) \]  

(4.178)

Simplifying results in

\[- F_F \frac{\Delta x_i}{6} (u_{i,0} + 2u_{i,1}) \]  

(4.179)

**Summary of Friction Term Discretization**

We have discretized the friction term using a discontinuous Galerkin method with linear basis and test functions. The two equations for \(j = 0\) and \(j = 1\) that discretize the friction term on element \(i = 1, 2, \ldots, N\) of the mesh respectively are:

\[- \int_{x_{i-1}}^{x_i} (u_{i,0}\phi_{i,0} + u_{i,1}\phi_{i,1}) F_F(t, x) \phi_{i,0} \, dx \]  

(4.180)

\[- \int_{x_{i-1}}^{x_i} (u_{i,0}\phi_{i,0} + u_{i,1}\phi_{i,1}) F_F(t, x) \phi_{i,1} \, dx \]  

(4.181)

However, if we assume the frictional term to be constant on the interval \([x_{i-1}, x_i]\) then we obtain:

\[- F_F \frac{\Delta x_i}{6} (2u_{i,0} + u_{i,1}) \]  

(4.182)

\[- F_F \frac{\Delta x_i}{6} (u_{i,0} + 2u_{i,1}) \]  

(4.183)

**4.5 Summary of Dynamic Semi-Discretization**

We spatially discretized the continuity and enthalpy equations of the 1-D Navier Stokes Equations derived in Chapter 3 using a discontinuous Galerkin spatial discretization with linear basis and test functions. Below we list the partial differential equations that we discretized.

Continuity Equation: \(\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} = 0\)  

(4.184)
4.5. Summary of Dynamic Semi-Discretization

Enthalpy Equation: 
\[ \frac{\partial pE}{\partial t} + A \frac{\partial \rho u h}{\partial x} = uA \frac{\partial p}{\partial x} - uA \frac{4}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) + A \frac{\partial q}{\partial x} + \dot{Q} + \Phi - uF \]  
(4.185)

We do not include the momentum equation because we assume that the pressure profile in
the pipe is known for all time and space. Thus we write:

Pressure Profile: 
\[ p(t, x) = \alpha(t, x) \]  
(4.186)

Additionally, we include the closure terms for temperature and density:

Temperature Correlation: 
\[ T = \pi(p, h) \]  
(4.187)

Density Correlation: 
\[ \rho = \omega(p, h) \]

We choose the following boundary conditions:

\[ v(t, 0) = v_0(t); \quad h(t, 0) = h_0(t); \quad \mu \frac{\partial h}{\partial x}(t, L) = 0 \]  
(4.188)

We also choose the following initial conditions:

\[ u(0, x) = u_{init}(x); \quad h(0, x) = h_{init}(x) \]  
(4.189)

Meanings for all symbols used in this summary can be found in nomenclature section of this
thesis. The spatial domain is a line with a left end point at 0 and right end point at L. We
then break this domain into N elements \((i = 1, 2, ..., N)\) and \(N+1\) points \((x_k \ k = 0, 1, ..., N)\)
where \(x_0 = 0\) and \(x_N = L\). Element \(i = 1, 2, ..., N\) of the mesh has length \(\Delta x_i = x_i - x_{i-1}\).

All spatial discretizations will be given for the \(i\)’th arbitrary element of the mesh where
\(i = 1, 2, ..., N\). The full semi-discretization comes from evaluating the resulting differential
equations for all elements \(i = 1, ..., N\). The pressure profile, temperature correlation, and
density correlations are assumed to be exact.
The Continuity Equation Semi-Discretization

\[
\frac{\Delta x_i}{3} \frac{\partial \rho_{i,0}}{\partial t} + \frac{\Delta x_i}{6} \frac{\partial \rho_{i,1}}{\partial t} - \rho_{i-1,1} u_{i-1,1,i} + \frac{\rho_{i,0} u_{i,0,i}}{3} + \frac{\rho_{i,0} u_{i,1,i}}{6} + \frac{\rho_{i,1} u_{i,0,i}}{3} = 0 \quad (4.190)
\]

\[
\frac{\Delta x_i}{6} \frac{\partial \rho_{i,0}}{\partial t} + \frac{\Delta x_i}{3} \frac{\partial \rho_{i,1}}{\partial t} + 2 \frac{\rho_{i,1} u_{i,1,i}}{3} - \frac{\rho_{i,0} u_{i,0,i}}{3} - \frac{\rho_{i,0} u_{i,1,i}}{6} - \frac{\rho_{i,1} u_{i,0,i}}{6} = 0 \quad (4.191)
\]

The Enthalpy Equation Semi-Discretization

Due to the length of the enthalpy equation, we break it up into many smaller terms. Table 4.2 below lists the label we give to each term.

<table>
<thead>
<tr>
<th>Term</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \int_{x_{i-1}}^{x_i} A \frac{\partial (\rho E)}{\partial t} \nu , dx )</td>
<td>Temporal Term</td>
</tr>
<tr>
<td>( \int_{x_{i-1}}^{x_i} A \frac{\partial \rho u h}{\partial x} \nu , dx )</td>
<td>Enthalpy Term</td>
</tr>
<tr>
<td>( \int_{x_{i-1}}^{x_i} u A \frac{\partial p}{\partial x} \nu , dx )</td>
<td>Pressure Term</td>
</tr>
<tr>
<td>( - \int_{x_{i-1}}^{x_i} u A \frac{4}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) \nu , dx )</td>
<td>Shear Term</td>
</tr>
<tr>
<td>( \int_{x_{i-1}}^{x_i} A \frac{\partial}{\partial x} \left( \kappa \frac{\partial T}{\partial x} \right) \nu , dx )</td>
<td>Heat Conductance Term</td>
</tr>
<tr>
<td>( \int_{x_{i-1}}^{x_i} \dot{Q} \nu , dx )</td>
<td>External Heat Transfer Term</td>
</tr>
<tr>
<td>( \int_{x_{i-1}}^{x_i} \Phi \nu , dx )</td>
<td>Viscous Dissipation Term</td>
</tr>
<tr>
<td>( \int_{x_{i-1}}^{x_i} u F_F \nu , dx )</td>
<td>Frictional Term</td>
</tr>
</tbody>
</table>

Table 4.2: Table labeling each term in the enthalpy equation.

We list each term’s discretization for \( j=0 \) and \( j=1 \) respectively below:
4.5. Summary of Dynamic Semi-Discretization

The Temporal Term

\[
\frac{A\Delta x_i}{12} (3h_{i,0} \frac{\partial \rho_{i,0}}{\partial t} + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} + h_{i,1} \frac{\partial \rho_{i,0}}{\partial t} + 3\rho_{i,0} \frac{\partial h_{i,0}}{\partial t} + \\
\rho_{i,0} \frac{\partial h_{i,1}}{\partial t} + \rho_{i,1} \frac{\partial h_{i,0}}{\partial t} + \rho_{i,0} \frac{\partial h_{i,1}}{\partial t} + A \int_{x_{i-1}}^{x_i} \frac{\partial \alpha(t, x)}{\partial t} \phi_{i,0} \, dx) 
\]

(4.192)

\[
\frac{A\Delta x_i}{12} (h_{i,0} \frac{\partial \rho_{i,0}}{\partial t} + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} + h_{i,1} \frac{\partial \rho_{i,0}}{\partial t} + 3h_{i,1} \frac{\partial \rho_{i,1}}{\partial t} + \rho_{i,0} \frac{\partial h_{i,0}}{\partial t} + \\
\rho_{i,0} \frac{\partial h_{i,1}}{\partial t} + \rho_{i,1} \frac{\partial h_{i,0}}{\partial t} + 3\rho_{i,1} \frac{\partial h_{i,1}}{\partial t} + A \int_{x_{i-1}}^{x_i} \frac{\partial \alpha(t, x)}{\partial t} \phi_{i,1} \, dx) 
\]

(4.193)

The Enthalpy Term

\[
-A\rho_{i-1,1} u_{i-1,1} h_{i-1,1} + \frac{A}{12} (3\rho_{i,0} h_{i,0} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,0} + \rho_{i,0} h_{i,1} u_{i,0} + \rho_{i,0} h_{i,0} u_{i,1} + \\
\rho_{i,1} h_{i,1} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,1} + \rho_{i,0} h_{i,1} u_{i,1} + 3\rho_{i,1} h_{i,1} u_{i,1}) 
\]

(4.194)

\[
A\rho_{i,1} u_{i,1} h_{i,1} - \frac{A}{12} (3\rho_{i,0} h_{i,0} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,0} + \rho_{i,0} h_{i,1} u_{i,0} + \rho_{i,0} h_{i,0} u_{i,1} + \\
\rho_{i,1} h_{i,1} u_{i,0} + \rho_{i,1} h_{i,0} u_{i,1} + \rho_{i,0} h_{i,1} u_{i,1} + 3\rho_{i,1} h_{i,1} u_{i,1}) 
\]

(4.195)

The Pressure Term

\[
A \int_{x_{i-1}}^{x_i} (u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1}) \frac{\partial \alpha(n\Delta t, x)}{\partial x} \phi_{i,0} \, dx 
\]

(4.196)

\[
A \int_{x_{i-1}}^{x_i} (u_{i,0} \phi_{i,0} + u_{i,1} \phi_{i,1}) \frac{\partial \alpha(n\Delta t, x)}{\partial x} \phi_{i,1} \, dx 
\]

(4.197)

If we assume a constant pressure drop along element \( i = 1, 2, \ldots N \) of the mesh, that is
Chapter 4. Discontinuous Galerkin Spatial Semi-Discretization

\[
\frac{\partial \alpha(t,x)}{\partial x} \bigg|_{x \in [x_{i-1},x_i]} = \Delta p_i(t)
\]

The discretization becomes:

\[
A \Delta p_i(t) \frac{\Delta x_i}{6} (2u_{i,0} + u_{i+1})
\]

(4.198)

\[
A \Delta p_i(t) \frac{\Delta x_i}{6} (u_{i,0} + 2u_{i+1})
\]

(4.199)

The Shear Term

\[
\frac{4}{3} A \mu u_{i-1,1} \psi_{i-1,1} - \frac{2}{9} A \mu (4\psi_{i,0}u_{i,0} - \psi_{i,0}u_{i,1} + 2\psi_{i,1}u_{i,0} + \psi_{i,1}u_{i,1})
\]

(4.200)

\[
-\frac{4}{3} A \mu u_{i,1} \psi_{i,1} + \frac{2}{9} A \mu (\psi_{i,0}u_{i,0} + 2\psi_{i,0}u_{i,1} - \psi_{i,1}u_{i,0} + 4\psi_{i,1}u_{i,1})
\]

(4.201)

The Heat Conductance Term

\[
-A \kappa \Gamma_{i-1,1} + \frac{1}{2} A \kappa (\Gamma_{i,0} + \Gamma_{i,1})
\]

(4.202)

\[
A \kappa \Gamma_{i,1} - \frac{\Delta x_i}{2} A \kappa (\Gamma_{i,0} + \Gamma_{i,1})
\]

(4.203)

The External Heat Transfer Term

\[
UP(T_{i,0} \frac{\Delta x_i}{3} + T_{i,1} \frac{\Delta x}{6} - T_{surr} \frac{\Delta x_i}{2})
\]

(4.204)

\[
UP(T_{i,0} \frac{\Delta x_i}{6} + T_{i,1} \frac{\Delta x}{3} - T_{surr} \frac{\Delta x_i}{2})
\]

(4.205)

The Viscous Dissipation Term

\[
\frac{2\mu}{3\Delta x_i} (u_{i,1} - u_{i,0})^2
\]

(4.206)

\[
\frac{2\mu}{3\Delta x_i} (u_{i,1} - u_{i,0})^2
\]

(4.207)
4.5. Summary of Dynamic Semi-Discretization

The Friction Term

\[- \int_{x_{i-1}}^{x_i} (u_{i,0}\phi_{i,0} + u_{i,1}\phi_{i,1}) F_F(t, x) \phi_{i,0} \, dx \quad (4.208)\]

\[- \int_{x_{i-1}}^{x_i} (u_{i,0}\phi_{i,0} + u_{i,1}\phi_{i,1}) F_F(t, x) \phi_{i,1} \, dx \quad (4.209)\]

If we assume that the friction factor is constant on the interval \([x_{i-1}, x_i]\) then the discretization becomes:

\[- F_F \frac{\Delta x_i}{6} (2u_{i,0} + u_{i,1}) \quad (4.210)\]

\[- F_F \frac{\Delta x_i}{6} (u_{i,0} + 2u_{i,1}) \quad (4.211)\]

The Velocity Flux Correlations

\[\frac{\Delta x_i}{6} (2\psi_{i,0} + \psi_{i,1}) + u_{i,0} - \frac{1}{2}(u_{i,0} + u_{i,1}) = 0 \quad (4.212)\]

\[\frac{\Delta x_i}{6} (\psi_{i,0} + 2\psi_{i,1}) - u_{i+1,0} + \frac{1}{2}(u_{i,0} + u_{i,1}) = 0 \quad (4.213)\]

The Temperature Flux Correlations

\[\frac{\Delta x_i}{6} (2\Gamma_{i,0} + \Gamma_{i,1}) + \frac{1}{2}(T_{i,0} - T_{i,1}) = 0 \quad (4.214)\]

\[\frac{\Delta x_i}{6} (\Gamma_{i,0} + 2\Gamma_{i,1}) - T_{i+1,0} + \frac{1}{2}(T_{i,0} + T_{i,1}) = 0 \quad (4.215)\]

In the next section we derive the steady state version of the 1D compressible Navier Stokes equations.
4.6 Discretization of Steady State Equations

In this section, we derive a steady state discretization of the 1D compressible Navier Stokes equations. To begin, we set all the temporal derivatives to zero in all of the partial differential equations. The resulting ordinary differential equations are given by

\[
\text{Continuity Equation: } \frac{\partial \rho u}{\partial x} = 0 \tag{4.216}
\]

\[
\text{Enthalpy Equation: } A \frac{\partial \rho u h}{\partial x} = u A \frac{\partial p}{\partial x} - u A \frac{4}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) + A \frac{\partial q}{\partial x} + \dot{Q} + \Phi - u F_F \tag{4.217}
\]

The resulting time independent pressure profile is

\[
\text{Pressure Profile: } p(x) = \alpha(x) \tag{4.218}
\]

Temperature and density correlations are used to close the model.

\[
\text{Temperature Correlation: } T = \pi(p, h) \tag{4.219}
\]

\[
\text{Density Correlation: } \rho = \omega(p, h)
\]

The Discretization of the Steady State Continuity Equation

From the steady state continuity equation, we can conclude that the mass flow rate \( \dot{m} \) through the pipe is constant. This relationship is given by

\[
A \rho(x) u(x) = \dot{m} \tag{4.220}
\]

Thus, we are able to say for \( j = 0, 1 \) on an arbitrary element \( i = 1, 2, \ldots N \):

\[
A \rho_{i,j} u_{i,j} = \dot{m} \tag{4.221}
\]

Discretization of the Steady State Enthalpy Equation

We begin with the steady state enthalpy equation

\[
A \frac{\partial \rho u h}{\partial x} = u A \frac{\partial p}{\partial x} - u A \frac{4}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) + A \frac{\partial q}{\partial x} + \dot{Q} + \Phi - u F_F \tag{4.222}
\]
4.6. Discretization of Steady State Equations

We use the product rule on the enthalpy term to obtain

\[ hA \frac{\partial \rho u}{\partial x} + \rho u A \frac{\partial h}{\partial x} = uA \frac{\partial p}{\partial x} - uA \frac{4}{3} \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial x} \right) + A \frac{\partial q}{\partial x} + \dot{Q} + \Phi - uF_F \]  \hspace{1cm} (4.223)

Using the continuity equation we can simplify this down to

\[ \dot{m}A \frac{\partial h}{\partial x} = uA \frac{\partial p}{\partial x} - uA \frac{4}{3} \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial x} \right) + A \frac{\partial q}{\partial x} + \dot{Q} + \Phi - uF_F \]  \hspace{1cm} (4.224)

The discretization of all terms on the right hand side will be exactly the same as in the transient case. As such, the discretization for these terms can be found in the previous section. The only term that we re-derive here is the enthalpy term as it is significantly different.

**Discretization of the Steady State Enthalpy Term**

We begin by multiplying the enthalpy term by a test function \( \nu \in H^1 \) and integrating across an element \( i = 1, 2, ... N \) yielding

\[ \int_{x_{i-1}}^{x_i} \dot{m}A \frac{\partial h}{\partial x} \nu \, dx \]  \hspace{1cm} (4.225)

Next, we integrate by parts to move the derivative to the test function and to introduce flux into the system:

\[ [\dot{m}Ah\nu]_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} \dot{m}Ah \frac{\partial \nu}{\partial x} \, dx \]  \hspace{1cm} (4.226)

We now replace \( \nu \) with a non-zero basis function \( \phi-i, j \) where \( j = 0, 1 \) on element \( i = 1, 2, ... N \) and \( h \) with its DG representation:

\[ [\dot{m}Ah\phi_{i,j}]_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} \dot{m}A(h_{i,0}\phi_{i,0} + h_{i,1}\phi_{i,1}) \frac{\partial \phi_{i,j}}{\partial x} \, dx \]  \hspace{1cm} (4.227)
Evaluating the steady state enthalpy term assuming that the test function is the left basis function \((j = 0)\) yields

\[
[mAh\phi_{i,0}]_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} \dot{m}A(h_{i,0}\phi_{i,0} + h_{i,1}\phi_{i,1}) \frac{\partial\phi_{i,0}}{\partial x} \, dx = \\
\dot{m}A(h(x_i)\phi_{i,0}(x_i) - h(x_{i-1})\phi_{i,0}(x_{i-1})) + \frac{\dot{m}A}{\Delta x_i} \int_{x_{i-1}}^{x_i} (h_{i,0}\phi_{i,0} + h_{i,1}\phi_{i,1}) \, dx = \\
-\dot{m}Ah_{i-1,1} + \frac{\dot{m}A}{\Delta x_i} (h_{i,0}\frac{\Delta x_i}{2} + h_{i,1}\frac{\Delta x_i}{2}) = \\
-\dot{m}Ah_{i-1,1} + \frac{\dot{m}A}{2} (h_{i,0} + h_{i,1})
\]

Similarly, we evaluate the steady state enthalpy term again but assuming that the test function is the right basis function \((j = 1)\) to obtain

\[
[mAh\phi_{i,1}]_{x_{i-1}}^{x_i} - \int_{x_{i-1}}^{x_i} \dot{m}A(h_{i,0}\phi_{i,0} + h_{i,1}\phi_{i,1}) \frac{\partial\phi_{i,1}}{\partial x} \, dx = \\
\dot{m}A(h(x_i)\phi_{i,1}(x_i) - h(x_{i-1})\phi_{i,1}(x_{i-1})) - \frac{\dot{m}A}{\Delta x_i} \int_{x_{i-1}}^{x_i} (h_{i,0}\phi_{i,0} + h_{i,1}\phi_{i,1}) \, dx = \\
\dot{m}Ah_{i,1} - \frac{\dot{m}A}{\Delta x_i} (h_{i,0}\frac{\Delta x_i}{2} + h_{i,1}\frac{\Delta x_i}{2}) = \\
\dot{m}Ah_{i,1} - \frac{\dot{m}A}{2} (h_{i,0} + h_{i,1})
\]

To summarize the steady state enthalpy term discretization for \(j = 0\) and \(j = 1\) respectively we have:

\[
-\dot{m}Ah_{i-1,1} + \frac{\dot{m}A}{2} (h_{i,0} + h_{i,1}) \\
\dot{m}Ah_{i,1} - \frac{\dot{m}A}{2} (h_{i,0} + h_{i,1})
\]

In the next chapter, we discuss the matrix from and solution to the system of semi-discrete differential algebraic equations derived in this chapter. We also discuss the solution to the steady state system of equations derived in this section.
Chapter 5

Solution of Semi-Discrete DAE

In this chapter we describe the matrix form of the semi-discretization derived in Chapter 4. In particular, we show that the semi-discretized system may be written as a Differential Algebraic Equation (DAE) of the form

\[ M(\mathbf{z}(t)) \mathbf{\dot{z}}(t) = \mathbf{G}(\mathbf{z}(t)) \]  

(5.1)

\( M(\mathbf{z}(t)) \) is the mass matrix of the system and \( \mathbf{G}(\mathbf{z}(t,x)) \) is right hand side vector of the semi-discretized equations. We define the state vector \( \mathbf{z}(t,x) \in \mathbb{R}^{14N} \) as:

\[
\mathbf{z}(t) = \begin{bmatrix}
\mathbf{h}(t) \\
\mathbf{\rho}(t) \\
\mathbf{\tau}(t) \\
\mathbf{p}(t) \\
\mathbf{u}(t) \\
\mathbf{\Psi}(t) \\
\mathbf{\Gamma}(t)
\end{bmatrix} \in \mathbb{R}^{14N}
\]
Each term inside of the state vector $z \in \mathbb{R}^{14N}$ is defined as:

$$h(t) = \begin{bmatrix} h_{1,0}(t) \\ h_{1,1}(t) \\ h_{2,0}(t) \\ h_{2,1}(t) \\ \vdots \\ h_{N,0}(t) \\ h_{N,1}(t) \end{bmatrix}, \quad \rho(t) = \begin{bmatrix} \rho_{1,0}(t) \\ \rho_{1,1}(t) \\ \rho_{2,0}(t) \\ \rho_{2,1}(t) \\ \vdots \\ \rho_{N,0}(t) \\ \rho_{N,1}(t) \end{bmatrix}, \quad T(t) = \begin{bmatrix} T_{1,0}(t) \\ T_{1,1}(t) \\ T_{2,0}(t) \\ T_{2,1}(t) \\ \vdots \\ T_{N,0}(t) \\ T_{N,1}(t) \end{bmatrix}, \quad \mathbf{p}(t) = \begin{bmatrix} p_{1,0}(t) \\ p_{1,1}(t) \\ p_{2,0}(t) \\ p_{2,1}(t) \\ \vdots \\ p_{N,0}(t) \\ p_{N,1}(t) \end{bmatrix}$$

$$u(t) = \begin{bmatrix} u_{1,0}(t) \\ u_{1,1}(t) \\ u_{2,0}(t) \\ u_{2,1}(t) \\ \vdots \\ u_{N,0}(t) \\ u_{N,1}(t) \end{bmatrix}, \quad \mathbf{\Psi}(t) = \begin{bmatrix} \psi_{1,0}(t) \\ \psi_{1,1}(t) \\ \psi_{2,0}(t) \\ \psi_{2,1}(t) \\ \vdots \\ \psi_{N,0}(t) \\ \psi_{N,1}(t) \end{bmatrix}, \quad \mathbf{\Gamma}(t) = \begin{bmatrix} \Gamma_{1,0}(t) \\ \Gamma_{1,1}(t) \\ \Gamma_{2,0}(t) \\ \Gamma_{2,1}(t) \\ \vdots \\ \Gamma_{N,0}(t) \\ \Gamma_{N,1}(t) \end{bmatrix}$$

Where $h(t), \rho(t), T(t), p(t), u(t), \mathbf{\Psi}(t), \mathbf{\Gamma}(t) \in \mathbb{R}^{2N}$.

Some of these equations are algebraic; this means that the mass matrix $M(z(t))$ is not invertible. This implies that instead of a system of ordinary differential equations, we are dealing with a system of differential algebraic equations (DAEs). When we include the Dirichlet boundary conditions, outlined in Chapter 4, the system consists of $10N+2$ algebraic equations and $4N-2$ differential equations. To account for the Dirichlet boundary conditions, we eliminate the first continuity and enthalpy equation ($i = 1, j = 0$). Thus, the mass matrix $M(z(t))$ will have the following form:

$$M(z(t)) = \begin{bmatrix} M_{1,1}(z(t)) & M_{1,2}(z(t)) \\ 0 & 0 \end{bmatrix}$$
Where $M_{1,1}, M_{1,2} \in \mathbb{R}^{4N-2} \times \mathbb{R}^{7N}$ and each block of zeros is in $\mathbb{R}^{10N+2} \times \mathbb{R}^{7N}$. Specifically, $M(z(t))$ will be structured as follows:

$$M(z(t)) = \begin{bmatrix} \text{Interior Enthalpy Equations LHS} \\
\text{Interior Continuity Equations LHS} \\
0 \end{bmatrix}$$

Similarly, the right hand side $G$ will be structured as follows:

$$G(z(t)) = \begin{bmatrix} \text{Interior Enthalpy Equations RHS} \\
\text{Interior Continuity Equations RHS} \\
\text{Dirichlet Boundary Conditions} \\
\text{Temperature Correlations} \\
\text{Pressure Profile} \\
\text{Density Correlations} \\
\Psi \text{ Correlations} \\
\Gamma \text{ Correlations} \end{bmatrix}$$

We use $LHS$ to stand for Left Hand Side and corresponds to all the coefficients in front of any temporal derivatives. We use $RHS$ to stand for all other terms in an expression. “Interior” refers to all equations besides the equation describing the left node of the first element. In the next section we break each equation into its left and right hand sides. Terms on the left hand side of the equals sign belong in the mass matrix $M(z(t))$ while terms on the right hand side belong in the right hand side vector $G(t))$. Additionally, each of the block matrices or vectors must be placed into the global matrix or vector accordingly.

For our implementation we assume a constant friction factor, a constant linear pressure drop and that the fluid is an ideal gas. In this matrix form, we will first give the general form of the correlations, then the correlations we use in our implementation.
Matrix Form of the Continuity Equations

\[
\frac{\Delta x_i}{6} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} \frac{\partial \rho_{i,0}}{\partial t} \\ \frac{\partial \rho_{i,1}}{\partial t} \end{bmatrix} = \begin{bmatrix} \rho_{i-1,1} u_{i-1,1} - \frac{\rho_{i,0} u_{i,0}}{3} - \frac{\rho_{i,0} u_{i,1}}{6} - \frac{\rho_{i,1} u_{i,0}}{6} - \frac{\rho_{i,1} u_{i,1}}{3} \\ - \frac{2\rho_{i,1} u_{i,1}}{3} + \frac{\rho_{i,0} u_{i,0}}{3} + \frac{\rho_{i,0} u_{i,1}}{6} + \frac{\rho_{i,1} u_{i,0}}{6} \end{bmatrix}
\]

Matrix Form of the Enthalpy Equations

\[
A \frac{\partial U}{\partial t} = -uA \frac{\partial p}{\partial x} - A \frac{\partial (\rho u h)}{\partial x} + uA \frac{4}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) + A \frac{\partial q}{\partial x} + \dot{Q} + \Phi - u F_F
\]

Due to the length of the enthalpy equation it does not neatly fit into a matrix form. We rearranged the terms in the equation above so that they correspond to the appropriate matrix. The semi-discretization of each of these terms can be found in the summary of Chapter 4. We do, however, list the mass matrix for a pressure drop invariant system below:

\[
\frac{A \Delta x_i}{12} \begin{bmatrix} (3h_{i,0} + h_{i,1}) & (h_{i,0} + h_{i,1}) & (3\rho_{i,0} + \rho_{i,1}) & (\rho_{i,0} + \rho_{i,1}) \\ (h_{i,0} + h_{i,1}) & (h_{i,0} + 3h_{i,1}) & (\rho_{i,0} + \rho_{i,1}) & (\rho_{i,0} + 3\rho_{i,1}) \end{bmatrix} \begin{bmatrix} \frac{\partial \rho_{i,0}}{\partial t} \\ \frac{\partial \rho_{i,1}}{\partial t} \\ \frac{\partial h_{i,0}}{\partial t} \\ \frac{\partial h_{i,1}}{\partial t} \end{bmatrix}
\]

Matrix Form of Dirichlet Boundary Conditions

\[
\begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} h_{i,0} - h_{\text{init}}(x) \\ u_{i,0} - u_{\text{init}}(x) \end{bmatrix}
\]
Matrix Form of Pressure Correlations

\[
\begin{bmatrix}
0 \\
0
\end{bmatrix} = \begin{bmatrix}
p_{i,0} - \alpha(t, x) \\
p_{i,1} - \alpha(t, x + \Delta x_i)
\end{bmatrix}
\]

In this correlation, \(x\) corresponds to the position of the left boundary of element \(i = 1, 2, \ldots N\). In our implementation we assume a uniform linear pressure drop of size \(\Delta p\) with an initial pressure of \(p_0\). Thus, \(\alpha(x, t) = p_0 - x\Delta p\). The pressure correlations in our implementation are:

\[
\begin{bmatrix}
0 \\
0
\end{bmatrix} = \begin{bmatrix}
p_{i,0} - p_0 + x\Delta p \\
p_{i,1} - p_0 + (x + \Delta x_i)\Delta p
\end{bmatrix}
\]

Matrix Form of Temperature Correlations

\[
\begin{bmatrix}
0 \\
0
\end{bmatrix} = \begin{bmatrix}
T_{i,0} - \pi(p_{i,0}, h_{i,0}) \\
T_{i,1} - \pi(p_{i,1}, h_{i,1})
\end{bmatrix}
\]

For an ideal gas the temperature is purely a function of enthalpy. Specifically, \(\pi(p, h) = h/C_p\) where \(C_p\) is the heat capacity of the ideal gas. Thus we have for our MATLAB\textsuperscript{TM} implementation:

\[
\begin{bmatrix}
0 \\
0
\end{bmatrix} = \begin{bmatrix}
T_{i,0} - h_{i,0}/C_p \\
T_{i,1} - h_{i,1}/C_p
\end{bmatrix}
\]

Matrix Form of Density Correlations

\[
\begin{bmatrix}
0 \\
0
\end{bmatrix} = \begin{bmatrix}
p_{i,0} - \omega(p_{i,0}, h_{i,0}) \\
p_{i,1} - \omega(p_{i,1}, h_{i,1})
\end{bmatrix}
\]
Since we assume an ideal gas in our implementation, we have: \( \omega(p, h) = \frac{pM_w}{RT} \) where \( M_w \) is the molecular weight of the gas, and \( R \) is the ideal gas constant. Thus, for our implementation we have:

\[
\begin{bmatrix}
0 \\
0 \\
0
\end{bmatrix} = 
\begin{bmatrix}
\rho_{i,0} - \frac{p_{i,0}M_w}{RT_{i,0}} \\
\rho_{i,1} - \frac{p_{i,1}M_w}{RT_{i,0}}
\end{bmatrix}
\]

### Matrix Form of Velocity Flux Correlations

\[
\begin{bmatrix}
0 \\
0
\end{bmatrix} = 
\begin{bmatrix}
\frac{\Delta x_i}{6} (2\psi_{i,0} + \psi_{i,1}) + u_{i,0} - \frac{1}{2}(u_{i,0} + u_{i,1}) \\
\frac{\Delta x_i}{6} (\psi_{i,0} + 2\psi_{i,1}) - u_{i+1,0} + \frac{1}{2}(u_{i,0} + u_{i,1})
\end{bmatrix}
\]

### Matrix Form of Temperature Flux Correlations

\[
\begin{bmatrix}
0 \\
0
\end{bmatrix} = 
\begin{bmatrix}
\frac{\Delta x_i}{6} (2\Gamma_{i,0} + \Gamma_{i,1}) + \frac{1}{2}(T_{i,0} - T_{i,1}) \\
\frac{\Delta x_i}{6} (\Gamma_{i,0} + 2\Gamma_{i,1}) - T_{i+1,0} + \frac{1}{2}(T_{i,0} + T_{i,1})
\end{bmatrix}
\]

Each of these matrix forms must be combined in such a way to fit the overall structure of a DAE:

\[
M(z(t))\dot{z}(t) = G(z(t))
\]

### 5.1 Numerical Solutions to DAEs

This section will briefly go over some numerical theory behind DAEs and their numerical solutions. Differential equations can be thought of as ordinary differential equations constrained to a manifold [22]. To illustrate this, consider the following differential equation in \( \mathbb{R}^2 \):

\[
y'(t) = y(t) + x(t) \quad IC: y(0) = y_0
\]
5.1. Numerical Solutions to DAEs

where \( x(t) \) is a forcing function. Assuming that \( x(t) \) is a "nice" function, \( y(t) \) exists for all time \( t > 0 \) and if the initial condition choice is arbitrary, then \( y(t) \) can span all of \( \mathbb{R} \). The pair \( (x(t), y(t)) \) lives and spans the entirety of \( \mathbb{R}^2 \). Now consider the same differential equation but with an algebraic constraint:

\[
\begin{align*}
y'(t) &= y(t) \\
1 &= y^2(t) - x^2(t)
\end{align*}
\]

This algebraic constraint drastically changes the possible solution space. We are no longer able to arbitrarily pick \( y_0 \) or \( x(t) \). Instead, the pair \( (y(t), x(t)) \) must live on the boundary of the unit circle. Additionally, \( y(t) \) may not exist for all \( t > 0 \). Some initial conditions yield curves that have a maximum integration time. For example, if we let \( y_0 = \sqrt{2} \) and \( x_0 = \sqrt{2} \), then at \( \hat{t} \approx 0.2196 \) we have \( y(\hat{t}) = 1 \) and \( x(\hat{t}) = 0 \). No further time steps, no matter how small, can be taken. Since \( y'(\hat{t}) = 1 \) letting additional time pass will result in \( y(t) > 1 \). This would break the algebraic constraint set on the system. Even in this simple example the manifold that constrains the differential equation has a large impact on the solution properties of the DAE.

Constraining differential equations to manifolds can make them significantly more difficult to solve than unconstrained ordinary differential equations. The past three decades have seen a significant amount of progress in understanding the nature of DAEs and finding their numerical solutions. Although DAEs have a similar matrix structure to ODEs, standard numerical integration techniques for ODEs may not work on all DAEs. Petzold outlines many problems that may occur when ODE techniques are directly applied to an arbitrary DAE [20]. Consider the following system:

\[
Ey' = Ay + g(t)
\]

Petzold illustrates that if \( g(t) \) does not have a continuous derivative, then backward difference time integrators will not only fail due to the discontinuity, but fail at the incorrect time
step [20]. Additionally, she explains that error estimates for arbitrary DAEs are massively different from ODEs which means that special techniques may need to be developed to solve any particular DAE.

Despite these potential pitfalls, some standard ODE techniques still work for special DAEs. Gear and Petzold determined which ODE methods are suitable for different classes of DAEs [9]. This is important because if a DAE falls into one of these classes, then the entire wealth of ODE solution algorithms are at ones disposal to solve it. On the other hand, if the DAE is outside of one of these classes, then a completely different technique is required to solve the DAE. An important finding of Gear and Petzold is that all DAEs of order 1 can be solved with standard ODE integration techniques. [9]. This is important in this thesis because with the ideal gas assumption, the semi-discretized 1D Navier Stokes equations is order 1.

To solve the the semi-discretized DAE described in this chapter, we use the built in MATLAB\textsuperscript{TM} function \textit{ode23t}. This solver is a Runge-Kutta single step time integrator that is third order accurate. It is the only explicit, built in MATLAB\textsuperscript{TM} time integrator that is capable of handling non-constant mass matrices. If initial conditions that are inconsistent with the algebraic constraints are given to \textit{ode23t}, then it will automatically perform a Newton solve to find initial conditions that are consistent with the algebraic constraints. Additionally, \textit{ode23t} is an adaptive solver, meaning that if a time step produces error estimates outside of the tolerance, it will continue reducing the step size until the error estimate is within the acceptable bound.

The discretized steady state equations are simply a set of non-linear equations. To solve these equations, we use the built in MATLAB\textsuperscript{TM} function \textit{fsolve}. \textit{Fsolve} is a Newton solver that can use three different algorithms: trust-region, trust-region dogleg, and Levenberg-Marquadt. The default algorithm is trust-region dogleg. To use \textit{fsolve} we must put the discretized equations in the form $F(z) = 0$ and provide an initial solution profile guess $z_{init}$. 
5.1. Numerical Solutions to DAEs

`Fsolve` will iterate until it has found a sufficiently close solution, the maximum number of iterations has passed, or it has performed the maximum number of right hand side function evaluations. In the next chapter, we will report the numerical results from our MATLAB\textsuperscript{TM} implementation of the semi-discretized equations.
Chapter 6

Steady State Newton Solver Results

In this section we will present the results of numerical tests of the steady state Newton solver from our MATLAB™ implementation described in Chapter 5. The purpose of these numerical tests is to illustrate three key points:

- The steady state solvers produce physically realistic results
- Illustrate the importance of non-uniform meshing for computational efficiency
- The steady state solver gives physically realistic solutions for a variety of flow regimes

All MATLAB™ scripts can be obtained from the following github: https://github.com/Zgrigorian/1D-Compressible-Navier-Stokes.

6.1 Implementation Assumptions and Introduction

We use MATLAB™ to solve for the steady state profile of the 1D compressible Navier Stokes equations. We will first list additional simplifications for our implementation:

- Linear time invariant pressure profile: \( \alpha(t, x) = p_0 + x\Delta p \)
- Fluid is air and behaves like an ideal gas:
  
  \[- T(p, h) = \pi(p, h) = \frac{h}{c_p} \]
  
  \[- \rho(p, h) = \omega(p, h) = \frac{nMW}{RT} \]
6.1. Implementation Assumptions and Introduction

\[ R = 8.314 \frac{kJ}{kmol \cdot K} \]

\[ M_w = 0.029 \frac{kg}{kmol} \]

- Dynamic viscosity \( \mu \) and thermal conductivity \( \kappa \) are constant

- The pipe perimeter \( P \) can be combined with the heat transfer coefficient \( U \) to form an overall heat transfer coefficient \( \tilde{U} \)

- The temperature of the surroundings is constant \( \frac{\partial T_{surr}}{\partial x} = \frac{\partial T_{surr}}{\partial t} = 0 \)

- The overall heat transfer coefficient of the pipe is constant \( \frac{\partial \tilde{U}}{\partial x} = \frac{\partial \tilde{U}}{\partial t} = 0 \)

The base parameters used in all of these tests are listed in the table below. Each test will only change one or two of these parameters. Our steady state solver uses the stable formulation described in Chapter 4. We use the built-in MATLAB\textsuperscript{TM} function \texttt{fsolve} to solve the system of non-linear equations. \texttt{Fsolve} is a Newton solver that uses three different algorithms, trust-region dogleg, trust-region, and Levenberg-Marquard. The default algorithm is trust-region

<table>
<thead>
<tr>
<th>Name</th>
<th>Symbol</th>
<th>Value</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pipe Area</td>
<td>( A )</td>
<td>1 ( m^2 )</td>
<td></td>
</tr>
<tr>
<td>Heat Capacity</td>
<td>( C_p )</td>
<td>1.0049 ( \frac{kJ}{kg \cdot K} )</td>
<td></td>
</tr>
<tr>
<td>Mass Flow Rate</td>
<td>( \dot{m} )</td>
<td>1 ( \frac{kg}{s} )</td>
<td></td>
</tr>
<tr>
<td>Molecular Weight</td>
<td>( M_w )</td>
<td>0.029 ( \frac{kg}{kmol} )</td>
<td></td>
</tr>
<tr>
<td>Friction Factor</td>
<td>( F_F )</td>
<td>1550 ( \frac{Pa}{m} )</td>
<td></td>
</tr>
<tr>
<td>Pipe Length</td>
<td>( L )</td>
<td>1 ( m )</td>
<td></td>
</tr>
<tr>
<td>Initial Pressure</td>
<td>( P_0 )</td>
<td>( 10^5 ) ( Pa )</td>
<td></td>
</tr>
<tr>
<td>Pressure Drop</td>
<td>( \Delta P )</td>
<td>-1500 ( \frac{Pa}{m} )</td>
<td></td>
</tr>
<tr>
<td>Ideal gas constant</td>
<td>( R )</td>
<td>8.314 ( \frac{kJ}{kmol \cdot K} )</td>
<td></td>
</tr>
<tr>
<td>Surrounding Temperature</td>
<td>( T_{surr} )</td>
<td>290 ( K )</td>
<td></td>
</tr>
<tr>
<td>Overall Heat Transfer Coefficient</td>
<td>( \tilde{U} )</td>
<td>45 ( \frac{kW}{m \cdot K} )</td>
<td></td>
</tr>
<tr>
<td>Dynamic Viscosity</td>
<td>( \mu )</td>
<td>( 2 \times 10^{-6} ) ( \frac{m^2}{s} )</td>
<td></td>
</tr>
<tr>
<td>Thermal Conductivity</td>
<td>( \kappa )</td>
<td>( 2.6 \times 10^{-5} ) ( \frac{kW}{m \cdot K} )</td>
<td></td>
</tr>
</tbody>
</table>

Table 6.1: Table of test parameters used for determining effect of non-uniform meshing on solution accuracy.

---

change one or two of these parameters. Our steady state solver uses the stable formulation described in Chapter 4. We use the built-in MATLAB\textsuperscript{TM} function \texttt{fsolve} to solve the system of non-linear equations. \texttt{Fsolve} is a Newton solver that uses three different algorithms, trust-region dogleg, trust-region, and Levenberg-Marquard. The default algorithm is trust-region
dogleg. We first describe preliminary results using a uniform mesh. These results motivate us to implement non-uniform meshing to reduce computational costs. Finally, we illustrate that our solver produces physically realistic results even as the mass flow rate approaches zero.

6.2 Uniform Meshing

We initially implemented the solver using a uniform mesh. Figures 6.1 and 6.2 demonstrate the specific enthalpy and velocity steady state solution profiles using the parameters found in Table 6.1. We run the steady state solver at varying numbers of elements until a smooth solution profile is achieved. Unrealistic oscillations in the solution profile appear when the mesh is not fine enough. For example, in Figure 6.1 one can see the specific enthalpy dips well below its equilibrium of approximately $290 \frac{kJ}{kg}$ for 4, 8, 16, and 32 elements. It isn’t until 64 elements that a physically realistic solution is attained. In Figure 6.1 it can be

![Steady State Specific Enthalpy Profile for Uniform Mesh](image_url)

Figure 6.1: The steady state enthalpy profile using a uniform mesh.
6.3. Non-Uniform Meshing

Figure 6.2: The steady state velocity profile using a uniform mesh.

seen that equilibrium is reached after about 0.15m of pipe. We need many elements in the first 0.15m of pipe but only a few afterwards. This is because after the first 0.15m of pipe the solution profile is either constant or linear. To improve efficiency, we allow the solver to handle non-uniform meshes. We include a mesh generator that will cluster elements in the beginning of the mesh. The reason for this is because the system should reach equilibrium relatively quickly assuming a reasonably sized overall heat transfer coefficient $\bar{U}$. The next section demonstrates the potential computational savings resulting from using a non-uniform mesh.

6.3 Non-Uniform Meshing

Intelligently meshing a system can significantly reduce the computational costs of a simulation while still producing physically realistic results. By clustering points where large changes in the profile occur, the physics of the system can be accurately captured. This is
especially important when the system reaches equilibrium quickly. To illustrate these potential savings we re-run the uniform mesh case but allow for non-uniform meshes. These parameters are listed in Table 6.1.

We cluster 70% of the elements in the first 10% of the pipe. The results of these tests can be seen in Figures 6.3 and 6.4. 10 elements were sufficient to generate a physically realistic solution. We do not see the large oscillations like in the uniform mesh case for even 32 elements. Additionally, the plots using both elements match relatively well and have the same equilibrium specific enthalpy. We then check the computational costs for each run.

![Figure 6.3: Plot of approximate steady state enthalpy profile using 10 non-uniformly clustered elements and 64 uniform elements.](image)

In both cases, the initial guess for the enthalpy profile a constant $300 \frac{kJ}{kg}$ for the entire pipe length. We then use the continuity equation, density correlation, and temperature correlations to solve for density, velocity, and temperature initial guesses. The initial guess for all flux correlation equations is 0. We use a dog-leg trust region to solve the resulting
6.3. Non-Uniform Meshing

Figure 6.4: Plot of approximate steady state velocity profile using 10 non-uniformly clustered elements and 64 uniform elements.

non-linear equations.

The system of 64 uniform elements requires 10766 right hand side function evaluations and 13 iterations. The system of 10 non-uniform elements requires only 1574 right hand side function evaluations and 13 iterations. This is approximately an 85% reduction in right hand side function evaluations. These savings become more dramatic if the system reaches equilibrium earlier in the pipe. One such way would be to further increase the overall heat transfer coefficient \( \tilde{U} \) increased.

An extreme example would be to let \( \tilde{U} = 1000 \frac{KW}{m^2\cdot K} \). In this case it requires 750 uniformly spaced elements to generate a physically realistic solution. Solving this system with 750 uniformly spaced elements required 144017 right hand side function evaluations and 16 iterations. We generated almost the exact same solution profiles with only 10 elements by clustering 70% of the elements in the first 1% of the pipe. This required only 1815 right
hand side function evaluations and 14 iterations to converge. Non-uniform meshing saved almost 99% of the right hand side function evaluations!

6.4 Steady State Low Flow Regimes
We wish for the steady state solver to work for a wide range of flow regimes. Specifically, it is important that the solver produces physically meaningful results as the mass flow rate approaches zero. We use all of the test case parameters outlined in Table 6.1 except for mass flow rate $\dot{m}$. We run the steady state solver with a mass flow rate of $1 \frac{kg}{s}$, $0.1 \frac{kg}{s}$, and $0.01 \frac{kg}{s}$. We use 64 non-uniform elements in all three cases. For each case, we cluster elements appropriately to achieve physically realistic solutions for all three mass flow rates. Figure 6.5 focuses on just the beginning 15% of the pipe. All of the flow regimes in Figure 6.5 are physically realistic. They do not exhibit any oscillations while the enthalpy approaches equilibrium. Additionally, we run the same system but in nearly adiabatic conditions. We

![Steady State Specific Enthalpy Profile](image)

Figure 6.5: Steady state enthalpy profile at various mass flow rates.
lower $U$ from 45 to 0.5 in this second test case. Figure 6.6 shows the steady state enthalpy solution at a mass flow rate of $\frac{1.5}{s}$, $0.1\frac{kg}{s}$, and $0.01\frac{kg}{s}$. We used 64 uniform elements for the discretization. As one can see, at low mass flow rates the air moves through the pipe slowly enough to the point where it is in equilibrium with the outside air. In the next chapter, we present the results of our dynamic solver and pseudo-transient initialization scheme.

![Figure 6.6: Nearly adiabatic (U=0.5) steady state enthalpy profile at varying mass flow rates.](image)

---

95
Chapter 7

Dynamic Solver Results

In this section we present the results of numerical tests of the dynamic solver from our MATLAB\textsuperscript{TM} implementation described in Chapter 5. The purpose of these numerical tests is to illustrate two key points:

- The dynamic solver produces physically realistic results
- Proof of concept for the potential computational savings of a pseudo-transient initialization scheme

All MATLAB\textsuperscript{TM} scripts can be obtained from the following github account:

https://github.com/Zgrigorian/1DCompressible-Navier-Stokes.

We make the same modeling assumptions as stated in Chapter 6.

7.1 Introduction

Our dynamic solver is designed to only integrate the dynamic solution for at most a few seconds. Additionally, the mass matrix is singular and dependent upon the state vector \( \mathbf{z}(t) \). We use MATLAB's\textsuperscript{TM} built in time integrator \textit{ode23t}. This integrator is capable of handling singular state dependent mass matrices. One problem with \textit{ode23t} is that it does not handle very stiff problems well. This means that it has trouble finding small enough step sizes when the solution profile changes quickly. To reduce the stiffness of the problem,
7.2. Dynamic System Solution

we omit the velocity and temperature flux correlations for the shear and heat conductance terms. Instead, we use the weakly unstable method described in the beginning of Chapter 2 section 2.2 to handle the flux terms. In this weakly unstable formulation, we allow ourselves to take the spatial derivatives of $T_h(t, x)$ and $u_h(t, x)$ to evaluate $\frac{\partial T_h(t, x)}{\partial x}$ and $\frac{\partial u_h(t, x)}{\partial x}$. When performing a pseudo-transient initialization we also use this weakly unstable formulation in the steady state solver so that the number of unknowns matches.

7.2 Dynamic System Solution

We run a test case to see if the dynamic solver produces physically realistic results. To reduce the stiffness of the system, we use a low overall heat transfer coefficient. Table 7.1 lists the test parameters used in the trial. We integrate the test case for one second using ode23t. Figures 7.1-7.3 show the enthalpy, velocity, and density solution profiles for an integration period of one second. Figures 7.4-7.5 show the enthalpy and velocity profiles of the dynamic solver at one second (the end of the time integration). For this trial run, we discretized the mesh with 32 uniform elements. The initial enthalpy and velocity profiles fed into ode23t.

<table>
<thead>
<tr>
<th>Name</th>
<th>Symbol</th>
<th>Value</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pipe Area</td>
<td>$A$</td>
<td>1</td>
<td>$m^2$</td>
</tr>
<tr>
<td>Heat Capacity</td>
<td>$C_p$</td>
<td>1.0049</td>
<td>$kJ/Kg/K$</td>
</tr>
<tr>
<td>Molecular Weight</td>
<td>$M_w$</td>
<td>0.029</td>
<td>$kg/kmol$</td>
</tr>
<tr>
<td>Friction Factor</td>
<td>$F_F$</td>
<td>1550</td>
<td>$m/Kg/s$</td>
</tr>
<tr>
<td>Length Perimeter</td>
<td>$L$</td>
<td>1</td>
<td>$m$</td>
</tr>
<tr>
<td>Initial Pressure</td>
<td>$P_0$</td>
<td>$10^5$</td>
<td>$Pa$</td>
</tr>
<tr>
<td>Pressure Drop</td>
<td>$\Delta P$</td>
<td>-1500</td>
<td>$Pa$</td>
</tr>
<tr>
<td>Initial Velocity</td>
<td>$u_0$</td>
<td>1</td>
<td>$m/s$</td>
</tr>
<tr>
<td>Ideal gas constant</td>
<td>$R$</td>
<td>8.314</td>
<td>$kJ/kmol/K$</td>
</tr>
<tr>
<td>Surrounding Temperature</td>
<td>$T_{surr}$</td>
<td>290</td>
<td>$K$</td>
</tr>
<tr>
<td>Overall Heat Transfer Coefficient</td>
<td>$\bar{U}$</td>
<td>10</td>
<td>$kW/m^2.K$</td>
</tr>
<tr>
<td>Dynamic Viscosity</td>
<td>$\mu$</td>
<td>$2 \times 10^{-6}$</td>
<td>$kg/m.s$</td>
</tr>
<tr>
<td>Thermal Conductivity</td>
<td>$\kappa$</td>
<td>$2.6 \times 10^{-5}$</td>
<td>$kW/m.K$</td>
</tr>
</tbody>
</table>

Table 7.1: Table of test parameters used for testing dynamic compressible flow solver
is a constant $300 \frac{kJ}{kg}$ and $1 \frac{m}{s}$. All other initial conditions result from solving the rest of the algebraic equations. Note that *ode23t* automatically solved the DAE to alter the velocity initial conditions to a line despite us feeding it a constant initial condition. All solution profiles are physically realistic and meaningful. Solution profiles have been rotated so that the majority of the surface plot can be seen.

Figure 7.1: The dynamic enthalpy profile over an integration period of one second. Note: the pipe coordinate axis is flipped to see surface plot more clearly.
7.2. Dynamic System Solution

Figure 7.2: The dynamic velocity profile over an integration period of one second. Note: the pipe coordinate axis is flipped to see surface plot more clearly.

Figure 7.3: The dynamic density profile over an integration period of one second. Note: the pipe coordinate axis is flipped to see surface plot more clearly.
Chapter 7. Dynamic Solver Results

Figure 7.4: Final enthalpy profile of the dynamic solver compared to steady state solution.

Figure 7.5: Final velocity profile of the dynamic solver compared to steady state solution.
7.3. Pseudo-Transient Initialization

Dynamic Solver Summary
We implement the discontinuous Galerkin semi-discretization described in Chapter 4 using the built in MATLAB™ time integrator ode23t. This built in function is capable of handling non-constant mass matrices. Due to the stiffness of the compressible Navier Stokes equations we need to use a weakly unstable implementation of diffusive terms. We tested our dynamic solver using the test case parameters in Table 7.1. The initial conditions fed to the solver is a constant enthalpy/velocity profile. The algebraic equations are solved to generate all other initial conditions. Our dynamic solver was able to march the dynamics forward by one second and produced physically realistic solution profiles.

7.3 Pseudo-Transient Initialization
In this section, we test our pseudo-transient initialization scheme against a baseline Newton solver initialization for a constant initial guess. We compare the number of right hand side function evaluations required to obtain the steady-state solution profile after simulating the dynamics for a short amount of time against plugging the initial guess straight into the steady state Newton solver. The left hand enthalpy boundary condition is always a constant; that is \( h(t, 0) = h_0 \). We use this as the initial condition for the dynamic solver \( h(x, 0) = h_0 \) and as the initial guess for the steady state solution profile for the Newton solver \( h_{init}(x) = h_0 \). We use the parameters listed in Table 7.1 to test the pseudo-transient initialization scheme except for the mass flow rate \( \dot{m} \). In the dynamic solver, we set the velocity boundary condition \( u(t, 0) = 1 \) and in the steady state Newton solver, we solve the continuity equation to determine the mass flow rate corresponding to \( u(0) = 1 \). We solve \( \pi(p, h) \) and \( \omega(p, h) \) to determine the appropriate initial conditions for the dynamic solver and corresponding initial guesses for the steady state Newton solver. Table 7.2 shows the right hand side function evaluations required for increasing integration times and different
Chapter 7. Dynamic Solver Results

Newton solver algorithms.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0s</td>
<td>0</td>
<td>18468</td>
<td>32832</td>
<td>55455</td>
</tr>
<tr>
<td>0.1s</td>
<td>548</td>
<td>15904</td>
<td>14364</td>
<td>44158</td>
</tr>
<tr>
<td>0.2s</td>
<td>556</td>
<td>12825</td>
<td>18981</td>
<td>33888</td>
</tr>
<tr>
<td>0.3s</td>
<td>563</td>
<td>7695</td>
<td>7695</td>
<td>23618</td>
</tr>
<tr>
<td>0.4s</td>
<td>571</td>
<td>5130</td>
<td>5643</td>
<td>13348</td>
</tr>
<tr>
<td>0.5s</td>
<td>578</td>
<td>4618</td>
<td>4104</td>
<td>6159</td>
</tr>
<tr>
<td>0.6s</td>
<td>586</td>
<td>3591</td>
<td>5130</td>
<td>5132</td>
</tr>
<tr>
<td>0.7s</td>
<td>593</td>
<td>2565</td>
<td>3591</td>
<td>3592</td>
</tr>
<tr>
<td>0.8s</td>
<td>601</td>
<td>3079</td>
<td>2052</td>
<td>2565</td>
</tr>
<tr>
<td>0.9s</td>
<td>608</td>
<td>2052</td>
<td>2052</td>
<td>2565</td>
</tr>
<tr>
<td>1.0s</td>
<td>616</td>
<td>2052</td>
<td>1539</td>
<td>1026</td>
</tr>
<tr>
<td>5.0s</td>
<td>917</td>
<td>2052</td>
<td>1539</td>
<td>1026</td>
</tr>
</tbody>
</table>

Table 7.2: Table of number of right hand side function evaluations needed to solve for steady state using the pseudo-transient initialization scheme employing different Newton algorithms.

As the integration time increases, the number of steady state solver evaluations generally decreases for all Newton solver methods. By one second of integration time, all three algorithms are only performing a few iterations to find the steady state solution profile. There is a point of diminishing returns depending on the algorithm. At five seconds the Levenberg Marquardt algorithm is taking a single step; the minimum required for the solver. Additionally, the trust-region dogleg and trust region methods do not save on any right hand side function evaluations after about one second of iteration time. Overall, we have shown that time marching an initialization for the steady state solver can produce significant computational savings; upwards of 85% or 90%.

We now recreate this experiment using different enthalpy boundary conditions. We vary the inlet enthalpy boundary condition from $h_{bc} = 270$ to $h_{bc} = 310$. The initial condition for the dynamic solver and initial guess for the Newton solver is both a constant equal to the enthalpy boundary condition. That is, if $h_{bc} = 300$ then $h(0, x) = 300$ for the dynamic solver and $h_{init} = 300$. All other initial conditions/guesses are solved for by using the
7.3. Pseudo-Transient Initialization

algebraic constraints of the DAE. Since trust-region dogleg performed the best overall from the previous test, it is used for these trials. We omit the number of dynamic evaluations since they are relatively inexpensive; the bulk of the function evaluations come from the steady state Newton solver. We compile the results of our runs in Table 7.3. In the \( h_{bc} = 310 \) case,

<table>
<thead>
<tr>
<th>Integration time</th>
<th>( h_{bc} = 270 ) RHS Evals.</th>
<th>( h_{bc} = 280 ) RHS Evals.</th>
<th>( h_{bc} = 290 ) RHS Evals.</th>
<th>( h_{bc} = 300 ) RHS Evals.</th>
<th>( h_{bc} = 310 ) RHS Evals.</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0s</td>
<td>21546</td>
<td>14365</td>
<td>12825</td>
<td>18468</td>
<td>30267</td>
</tr>
<tr>
<td>0.1s</td>
<td>18982</td>
<td>11799</td>
<td>7695</td>
<td>15904</td>
<td>18984</td>
</tr>
<tr>
<td>0.2s</td>
<td>15390</td>
<td>7182</td>
<td>5130</td>
<td>12825</td>
<td>14365</td>
</tr>
<tr>
<td>0.3s</td>
<td>9234</td>
<td>4617</td>
<td>4618</td>
<td>7695</td>
<td>11799</td>
</tr>
<tr>
<td>0.4s</td>
<td>5643</td>
<td>4618</td>
<td>3591</td>
<td>5130</td>
<td>7182</td>
</tr>
<tr>
<td>0.5s</td>
<td>5131</td>
<td>3591</td>
<td>2565</td>
<td>4618</td>
<td>6157</td>
</tr>
<tr>
<td>0.6s</td>
<td>4619</td>
<td>2052</td>
<td>2052</td>
<td>2565</td>
<td>NaN</td>
</tr>
<tr>
<td>0.8s</td>
<td>2565</td>
<td>2052</td>
<td>2052</td>
<td>3079</td>
<td>NaN</td>
</tr>
<tr>
<td>0.9s</td>
<td>2052</td>
<td>2052</td>
<td>2052</td>
<td>2052</td>
<td>NaN</td>
</tr>
<tr>
<td>1.0s</td>
<td>2052</td>
<td>2052</td>
<td>2052</td>
<td>2052</td>
<td>NaN</td>
</tr>
<tr>
<td>5.0s</td>
<td>2052</td>
<td>2052</td>
<td>2052</td>
<td>2052</td>
<td>NaN</td>
</tr>
</tbody>
</table>

Table 7.3: Table of right hand side function evaluations to attain steady state varying the enthalpy boundary condition

the dynamic solver was unable to integrate the system forward further than approximately 0.5s. Only data up to 0.5s was able to be collected for this test case. However, as one can see, the pseudo transient initialization reduces the number of right hand side function evaluations for a variety of enthalpy boundary conditions and initializations. In all but one case, where the dynamics could not integrate the system forward far enough, all boundary conditions were able to reach a point where Newton’s method only required a few steps. Additionally, the further the boundary condition is from the ambient surrounding temperature the longer it takes to reach the region of quadratic convergence. This is because the equilibrium of the system is close to the surrounding temperature. The further one starts from the equilibrium, the longer it should take to reach. In the next chapter, we present a summary and conclusions of this thesis.
Chapter 8

Summary and Conclusions

This thesis had four main goals:

- Derive a model for 1D compressible fluid pipe flow from the 3D compressible Navier Stokes equations
- Semi-discretize the model using a linear Discontinuous Galerkin method
- Implement a dynamic and steady state solver for the discretized equations
- Obtain preliminary information on the efficiency of a pseudo-transient initialization scheme for the steady state Newton solver

The 1D compressible Navier Stokes equations derived in Chapter 3 are:

Continuity Equation: \[ \frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} = 0 \]

Momentum Equation: \[ \frac{\partial (\rho u)}{\partial t} + \frac{\partial (\rho u^2)}{\partial x} = -\rho g \frac{dz}{dx} - \frac{\partial \rho}{\partial x} + \frac{4}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) + F_F \]

Enthalpy Equation: \[ \frac{\partial \rho E}{\partial t} + \frac{\partial \rho u h}{\partial x} = u \frac{\partial \rho}{\partial x} - u \frac{4}{3} \frac{\partial}{\partial x} \left( \frac{\mu}{\partial x} \right) \]
\[ + \frac{\partial A q}{\partial x} + \dot{Q} + \Phi - u F_F \]

(8.1)

We then simplified these equations by assuming a known pressure profile \( p(t, x) = \alpha(t, x) \) to replace the momentum equation. We also close the model by including a temperature correlation \( T(t, x) = \pi(p, h) \) and \( \rho(t, x) = \omega(p, h) \).
8.1 Final Differential Algebraic Equations

The final system of differential algebraic equations consist of the following differential equations:

Continuity Equation:
\[
\frac{\partial A}{\partial t} + \frac{\partial A u}{\partial x} = 0
\]

Enthalpy Equation:
\[
\frac{\partial A E}{\partial t} + \frac{\partial A u h}{\partial x} = u \frac{\partial A p}{\partial x} - u \frac{4}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) + \frac{\partial A q}{\partial x} + \dot{Q} + \Phi - u F_F \tag{8.2}
\]

With known algebraic pressure profile and closure equations:

Pressure Profile: \( p(t, x) = \alpha(p, h) \)

Temperature Correlation: \( T(t, x) = \pi(p, h) \) (8.3)

Density Correlation: \( \rho(t, x) = \omega(p, h) \)

To finish closing the model, we choose the following boundary conditions:

\[
p(t, 0) = p_0; \quad u(t, 0) = u_0(t); \quad h(t, 0) = h_0(t); \quad \mu \frac{\partial h}{\partial x}(t, L) = 0
\] (8.4)

We also choose the following initial conditions:

\[
u(0, x) = u_{init}(x); \quad h(0, x) = h_{init}(x)
\] (8.5)

8.2 Dynamic System Semi-Discretization

We spatially discretize the system with a linear discontinuous Galerkin method to produce the following semi-discrete model:

The Continuity Equation Semi-Discretization

\[
\frac{\Delta x_i}{3} \frac{\partial \rho_{i,0}}{\partial t} + \frac{\Delta x_i}{6} \frac{\partial \rho_{i,1}}{\partial t} - \rho_{i-1,1} u_{i-1,1} + \frac{\rho_{i,0} u_{i,0}}{3} + \frac{\rho_{i,0} u_{i,1}}{6} + \frac{\rho_{i,1} u_{i,0}}{6} + \frac{\rho_{i,1} u_{i,1}}{3} = 0
\] (8.6)

\[
\frac{\Delta x_i}{6} \frac{\partial \rho_{i,0}}{\partial t} + \frac{\Delta x_i}{3} \frac{\partial \rho_{i,1}}{\partial t} + \frac{2\rho_{i,1} u_{i,1}}{3} - \frac{\rho_{i,0} u_{i,0}}{3} - \frac{\rho_{i,0} u_{i,1}}{6} - \frac{\rho_{i,1} u_{i,0}}{6} = 0
\] (8.7)
The Enthalpy Equation Semi-Discretization

Due to the length of the enthalpy equation, we break it up into many smaller terms. Table 8.1 below lists the label we give to each term.

<table>
<thead>
<tr>
<th>Term</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\int_{x_{i-1}}^{x_i} A \frac{\partial (\rho E)}{\partial t} \nu , dx$</td>
<td>Temporal Term</td>
</tr>
<tr>
<td>$\int_{x_{i-1}}^{x_i} A \frac{\partial \rho u h}{\partial x} \nu , dx$</td>
<td>Enthalpy Term</td>
</tr>
<tr>
<td>$\int_{x_{i-1}}^{x_i} u A \frac{\partial p}{\partial x} \nu , dx$</td>
<td>Pressure Term</td>
</tr>
<tr>
<td>$- \int_{x_{i-1}}^{x_i} u A \frac{4}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) \nu , dx$</td>
<td>Shear Term</td>
</tr>
<tr>
<td>$\int_{x_{i-1}}^{x_i} A \frac{\partial}{\partial x} \left( \kappa \frac{\partial T}{\partial x} \right) \nu , dx$</td>
<td>Heat Conductance Term</td>
</tr>
<tr>
<td>$\int_{x_{i-1}}^{x_i} \dot{Q} \nu , dx$</td>
<td>External Heat Transfer Term</td>
</tr>
<tr>
<td>$\int_{x_{i-1}}^{x_i} \Phi \nu , dx$</td>
<td>Viscous Dissipation Term</td>
</tr>
<tr>
<td>$\int_{x_{i-1}}^{x_i} u F_F \nu , dx$</td>
<td>Frictional Term</td>
</tr>
</tbody>
</table>

Table 8.1: Table labeling each term in the enthalpy equation.

The Temporal Term

\[
\frac{A \Delta x_i}{12} \left( 3 h_{i,0} \frac{\partial \rho_{i,0}}{\partial t} + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} + h_{i,1} \frac{\partial \rho_{i,0}}{\partial t} + h_{i,1} \frac{\partial \rho_{i,1}}{\partial t} + 3 \rho_{i,0} \frac{\partial h_{i,0}}{\partial t} + \rho_{i,0} \frac{\partial h_{i,1}}{\partial t} + \rho_{i,1} \frac{\partial h_{i,0}}{\partial t} + \rho_{i,1} \frac{\partial h_{i,1}}{\partial t} \right) + A \int_{x_{i-1}}^{x_i} \frac{\partial \alpha(t,x)}{\partial t} \phi_{i,0} \, dx
\]

(8.8)

\[
\frac{A \Delta x_i}{12} \left( h_{i,0} \frac{\partial \rho_{i,0}}{\partial t} + h_{i,0} \frac{\partial \rho_{i,1}}{\partial t} + h_{i,1} \frac{\partial \rho_{i,0}}{\partial t} + h_{i,1} \frac{\partial \rho_{i,1}}{\partial t} + 3 \rho_{i,0} \frac{\partial h_{i,0}}{\partial t} + \rho_{i,0} \frac{\partial h_{i,1}}{\partial t} + \rho_{i,1} \frac{\partial h_{i,0}}{\partial t} + \rho_{i,1} \frac{\partial h_{i,1}}{\partial t} \right) + A \int_{x_{i-1}}^{x_i} \frac{\partial \alpha(t,x)}{\partial t} \phi_{i,1} \, dx
\]

(8.9)
8.2. Dynamic System Semi-Discretization

The Enthalpy Term

\[-A\rho_{i-1}u_{i-1}h_{i-1} + \frac{A}{12}(3\rho_{i,0}h_{i,0}u_{i,0} + \rho_{i,1}h_{i,0}u_{i,0} + \rho_{i,0}h_{i,1}u_{i,0} + \rho_{i,0}h_{i,0}u_{i,1}+ \rho_{i,1}h_{i,1}u_{i,0} + \rho_{i,1}h_{i,0}u_{i,1} + \rho_{i,0}h_{i,1}u_{i,0} + 3\rho_{i,1}h_{i,1}u_{i,1}) \tag{8.10}\]

\[A\rho_{i,1}u_{i,1}h_{i,1} - \frac{A}{12}(3\rho_{i,0}h_{i,0}u_{i,0} + \rho_{i,1}h_{i,0}u_{i,0} + \rho_{i,0}h_{i,1}u_{i,0} + \rho_{i,0}h_{i,0}u_{i,1}+ \rho_{i,1}h_{i,1}u_{i,0} + \rho_{i,1}h_{i,0}u_{i,1} + \rho_{i,0}h_{i,1}u_{i,0} + 3\rho_{i,1}h_{i,1}u_{i,1}) \tag{8.11}\]

The Pressure Term

\[A \int_{x_{i-1}}^{x_i} (u_{i,0}\phi_{i,0} + u_{i,1}\phi_{i,1}) \frac{\partial\alpha(n\Delta t, x)}{\partial x} \phi_{i,0} \, dx \tag{8.12}\]

\[A \int_{x_{i-1}}^{x_i} (u_{i,0}\phi_{i,0} + u_{i,1}\phi_{i,1}) \frac{\partial\alpha(n\Delta t, x)}{\partial x} \phi_{i,1} \, dx \tag{8.13}\]

If we assume a constant pressure drop along element i of the mesh, that is \(\frac{\partial \alpha(t, x)}{\partial x}\big|_{x \in [x_{i-1}, x_i]} = \Delta p_i(t)\) equation becomes:

\[A\Delta p_i(t) \frac{\Delta x_i}{6}(2u_{i,0} + u_{i+1}) \tag{8.14}\]

\[A\Delta p_i(t) \frac{\Delta x_i}{6}(u_{i,0} + 2u_{i+1}) \tag{8.15}\]

The Shear Term

\[\frac{4}{3}A\mu u_{i-1,1}\psi_{i-1,1} - \frac{2}{9}A\mu(4\psi_{i,0}u_{i,0} - \psi_{i,0}u_{i,1} + 2\psi_{i,1}u_{i,0} + \psi_{i,1}u_{i,1}) \tag{8.16}\]

\[-\frac{4}{3}A\mu u_{i,1}\psi_{i,1} + \frac{2}{9}A\mu(\psi_{i,0}u_{i,0} + 2\psi_{i,0}u_{i,1} - \psi_{i,1}u_{i,0} + 4\psi_{i,1}u_{i,1}) \tag{8.17}\]
The Heat Conductance Term

\[-A\kappa \Gamma_{i-1,1} + \frac{1}{2}A\kappa(\Gamma_{i,0} + \Gamma_{i,1})\]  
\[A\kappa \Gamma_{i,1} - \frac{\Delta x_i}{2}A\kappa(\Gamma_{i,0} + \Gamma_{i,1})\]  

(8.18)  

(8.19)

The External Heat Transfer Term

\[UP(T_{i,0}\frac{\Delta x_i}{3} + T_{i,1}\frac{\Delta x_i}{6} - T_{surr}\frac{\Delta x_i}{2})\]  
\[UP(T_{i,0}\frac{\Delta x_i}{6} + T_{i,1}\frac{\Delta x_i}{3} - T_{surr}\frac{\Delta x_i}{2})\]  

(8.20)  

(8.21)

The Viscous Dissipation Term

\[2\mu \frac{2}{3\Delta x_i}(u_{i,1} - u_{i,0})^2\]  
\[2\mu \frac{2}{3\Delta x_i}(u_{i,1} - u_{i,0})^2\]  

(8.22)  

(8.23)

The Friction Term

\[-\int_{x_{i-1}}^{x_i} (u_{i,0}\phi_{i,0} + u_{i,1}\phi_{i,1})F_F(t, x)\phi_{i,0} \, dx\]  
\[-\int_{x_{i-1}}^{x_i} (u_{i,0}\phi_{i,0} + u_{i,1}\phi_{i,1})F_F(t, x)\phi_{i,1} \, dx\]  

(8.24)  

(8.25)
8.3. Steady State Discretization

However, if we assume that the friction factor is constant on the interval \([x_{i-1}, x_i]\) then we obtain:

\[
-F_F \frac{\Delta x_i}{6} (2u_{i,0} + u_{i,1})
\]

(8.26)

\[
-F_F \frac{\Delta x_i}{6} (u_{i,0} + 2u_{i,1})
\]

(8.27)

The Velocity Flux Correlations

\[
\frac{\Delta x_i}{6} (2\psi_{i,0} + \psi_{i,1}) + u_{i,0} - \frac{1}{2} (u_{i,0} + u_{i,1}) = 0
\]

(8.28)

\[
\frac{\Delta x_i}{6} (\psi_{i,0} + 2\psi_{i,1}) - u_{i+1,0} + \frac{1}{2} (u_{i,0} + u_{i,1}) = 0
\]

(8.29)

The Temperature Flux Correlations

\[
\frac{\Delta x_i}{6} (2\Gamma_{i,0} + \Gamma_{i,1}) + \frac{1}{2} (T_{i,0} - T_{i,1}) = 0
\]

(8.30)

\[
\frac{\Delta x_i}{6} (\Gamma_{i,0} + 2\Gamma_{i,1}) - T_{i+1,0} + \frac{1}{2} (T_{i,0} + T_{i,1}) = 0
\]

(8.31)

8.3 Steady State Discretization

We generate the steady state version of the dynamic equations by assuming the temporal derivative is zero. The pressure profile, temperature correlation, and density correlations all remain the same as in the dynamic case. The resulting steady state version of the continuity and enthalpy equations are shown below:

Continuity Equation:

\[
\frac{\partial \rho u}{\partial x} = 0
\]

(8.32)

Enthalpy Equation:

\[
A \frac{\partial \rho u h}{\partial x} = u A \frac{\partial p}{\partial x} - u A \frac{4}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) + A \frac{\partial q}{\partial x} + \dot{Q} + \Phi - u F_F
\]

(8.33)
Chapter 8. Summary and Conclusions

Discretized Steady State Continuity Equation

\[ A \rho_{i,j} u_{i,j} = \dot{m} \quad (8.34) \]

where \( \dot{m} \) is the mass flow rate through the pipe for \( j = 0 \) or \( 1 \).

Discretized Steady State Enthalpy Equation

All terms in the enthalpy equation are the same as in the dynamical case except for the enthalpy term. The steady state enthalpy term is:

\[ -\dot{m} A h_{i-1,1} + \frac{\dot{m} A}{2} (h_{i,0} + h_{i,1}) \quad (8.35) \]
\[ \dot{m} A h_{i,1} - \frac{\dot{m} A}{2} (h_{i,0} + h_{i,1}) \quad (8.36) \]

8.4 Numerical Results

We implemented the steady state Newton solver using MATLAB’s\textsuperscript{TM} built-in solver \texttt{fsolve} and the dynamical solver using MATLAB’s\textsuperscript{TM} built-in DAE time integrator \texttt{ode23t}. We made the following modeling assumptions:

- Linear time invariant pressure profile: \( \alpha(t,x) = p_0 + x \Delta p \)
- Fluid is air and behaves like an ideal gas:
  - \( T(p,h) = \pi(p,h) = \frac{h}{C_p} \)
  - \( \rho(p,h) = \omega(p,h) = \frac{\rho M_w}{RT} \)
  - \( R = 8.314 \frac{kJ}{kmol \cdot T} \)
  - \( M_w = 0.029 \frac{kg}{kmol} \)
- Dynamic viscosity and thermal conductivity are constant
8.5. Future Work

- The pipe perimeter $P$ can be combined with the heat transfer coefficient $U$ to form an overall heat transfer coefficient $\bar{U}$

- The temperature of the surroundings is constant $\frac{\partial T_{\text{surr}}}{\partial x} = \frac{\partial T_{\text{surr}}}{\partial t} = 0$

- The overall heat transfer coefficient of the pipe is constant $\frac{\partial \bar{U}}{\partial x} = \frac{\partial \bar{U}}{\partial t} = 0$

Additionally, due to the stiffness of the problem, our dynamical solver uses a weak formulation of all second derivatives present. The Newton solver uses the strong formulation presented in this thesis except while testing our pseudo-transient initialization scheme. We use the weak implementation in the Newton solver to keep the number of unknowns identical between the two solvers. Non-uniform meshing in the Newton solver has been shown to increase computational efficiency in cases where the system reaches equilibrium quickly. In extreme cases, we were able to achieve a 99% reduction in right hand side function evaluations to achieve physically realistic steady state solutions. Additionally, the Newton solver is capable of handling a wide variety of flow regimes. Using non-uniform meshing, we were able to achieve physically realistic solutions mass flow rates for $0.01 \frac{\text{kg}}{\text{s}}$, $0.1 \frac{\text{kg}}{\text{s}}$ and $1.0 \frac{\text{kg}}{\text{s}}$. We tested the dynamic solver and it produces physically realistic results. However, it has trouble integrating systems that change too rapidly due to stiffness problems. Finally, we obtained preliminary results for our pseudo-transient initialization scheme. The scheme was able to significantly reduce computational costs using a variety of initial conditions/guesses and Newton solver algorithms.

8.5 Future Work

Many areas surrounding the 1D compressible Navier Stokes equations and our pseudo-transient initialization scheme remain unexplored in this thesis. The first area of exploration would be to build a better time integrator for the dynamic solver. Specifically, it would be
interesting to implement an implicit solver instead of using an explicit solver. Even though the explicit time stepper is computationally less expensive and easier to implement, it is ultimately unstable. An implicit time stepper is unconditionally stable, allowing for larger time steps and avoiding many instability issues. A first step in this investigation would be using MATLAB’s \textsuperscript{TM} built-in implicit DAE time integrator \textit{ode15i}.

Another area of potential research is allowing for parameters to vary with time and space; including pipe area, perimeter, surrounding temperature, dynamic viscosity, thermal conductivity, and the overall heat transfer coefficient. All of these parameters can vary greatly in real-world applications. The assumption that they are all constant is only realistic when the fluid state has very little variation. Additionally, this allows us to begin exploring two-phase pipe flow which is when there is both liquid and gas simultaneously flowing through a pipe. This application is important for heat exchangers since heat exchangers commonly condense hot steam into water or heat water into steam.

In the future, we would like to eliminate the ideal gas assumption and instead use empirical correlations for density and temperature. Additionally, we would like to allow for the pressure profile to be a function of enthalpy and mass flow rate. It is rare to know the pressure profile everywhere in a pipe. In practice, empirical pressure drop correlations based on velocity and enthalpy are used to estimate the change in pressure across an element. To accompany this, we would also like to replace the constant friction factor with a correlation based on fluid velocities. Eliminating these assumptions allows the model to be used in more physically realistic situations.

Finally, we would like to investigate large eddy approximation techniques to approximate the residual terms in Chapter 3. The assumption that these residual terms are negligible is largely based off of practical experience and intuition. It is valuable to know under what flow regimes this assumption is valid. Additionally, if there is a flow regime where this
8.5. Future Work

assumption is not valid, large eddy approximation techniques can be used to quantify and handle these residual terms.
Bibliography


[8] Paul Deuring, Robert Eymard, and Marcus Mildner. $L^2$-Stability Independent of Diffu-
sion for a Finite Element–Finite Volume Discretization of a Linear Convection-Diffusion


BIBLIOGRAPHY


